

**DESIGN OPTIMISATION OF MULTI-STORY STEEL
STRUCTURES WITH SINGLE AND MULTI-OBJECTIVE
FUNCTIONS USING THREE OPTIMISATION TECHNIQUES**

Abdihakem Emhemmed Melad Alkhadashi

A thesis submitted in partial fulfilment of the requirements of the Nottingham
Trent University for the degree of Doctor of Philosophy

August 2022

Dedication

I dedicate this work to the soul of my late father

COPYRIGHT DECLARATION

I certify that: The copyright of this dissertation rests with Nottingham Trent University. No information derived from this dissertation shall be published without prior consent of the University.

The submission of this work for assessment confirms that the work is mine alone and that all other sources have been acknowledge consistent with regulation of the University.

Name: Abdalhakem Alkhadashi

List of Publications

A number of papers and posters were presented at different conferences across the world, generated from some parts of the current research study. These are cited as follows:

Alkhadashi, A., Mohammad, F., Zubayr, R.O., Aoun Klalib, H., Balik, P., 2022. Multi-objective design optimisation of steel framed structures using three different methods. *International Journal of Structural Integrity* 13, 92–111. <https://doi.org/10.1108/IJSI-07-2021-0080>

Alkhadashi, A., (2022). Estimation and Minimization of Embodied Carbon in Steel Structures Using Two Optimisation Approaches. *Athens Institution for Education and Research (ATINER conference)*. Athens, Greece.

Alkhadashi, A., (2021). Multi-objective Design Optimisation of 3D Steel Structures using GA, PSO and HAS Methods. Nottingham Trent University, *School of Architecture, Design and Built Environment*, (Annual conference). Nottingham, United Kingdom.

A poster entitled as ‘The Effect of Optimisation Techniques on The Design Optimisation of Steel Structures’ was presented at ‘Nottingham Trent University’ ADBE Conference 2019’ in June 2019. The conference was organised by ‘School of Architecture, Design and Built Environment’ in Nottingham, UK.

Abstract

Undoubtedly, the best design of any structure aims at the most economical and environmentally balanced solution without impairing its function and structural integrity. To achieve this, structural designers often engaged in design optimisation. In this study, single and multi-objective stochastic search methods are proposed for optimum design of two and three-dimensional multi-story structures with three, six, and nine stories. The optimality objectives are the structure weight and embodied energy as well as calculating the cost and embodied carbon of the resulting optimum options. Three optimality algorithms developed in MATLAB, namely, Genetic Algorithms (GA), Particle Swarm Optimisation (PSO) and Harmony Search Algorithm (HSA), were used for structural optimisation to compare the effectiveness of the algorithms. Two life cycle stages were considered, production and construction stages which include three boundaries: materials, transportation, and erection. In the formulation of the optimum design problem, 107 universal beams “UKB” and 64 columns “UKC” sections were considered for the discrete design variables. The imposed behavioural constraints in the optimum design process were set according to the provision of EC3.

This research developed optimisation models for evaluation of embodied energy, embodied carbon and cost for steel structures to assist designers, during the initial stages, to evaluate design decisions against their energy consumption and carbon impacts. The study shows that the integration of the analysis, design and optimisation methods employed in MATLAB can be effective in obtaining prompt optimum results during the decision-making stage.

Overall, this research demonstrates that the three methods (i.e GA, PSO and HAS) are very useful tools in improving the structural performance of buildings and are effective in reducing the structural weight and embodied energy. Although, a critical observation of the optimised designs shows that the results obtained via HSA are generally better solutions in comparison with those derived from GA and PSO. The total weight and embodied energy for HSA are, on average, 3% and 5% less than that of GA and PSO respectively, when applied to single objective problems. Whereas, when the weight and embodied energy functions are taken as a multi-objective, HSA showed an average difference of 16% and 7% less than that of GA and PSO respectively, and this difference increases in larger structures. Overall, the resulting embodied energy was directly linear to weight. In addition to this, extensive optimum design charts were produced to enable designers in obtaining prompt optimum results during the

decision-making stage. The research suggests areas for further investigation and provides recommendations based on the study findings and conclusions.

The results show that combining analysis, design, and optimization methods in MATLAB can be effective in obtaining prompt optimal results during the decision-making stage, with solutions obtained in less than 12 minutes for up to nine-story three-dimensional design problems.

Keywords: optimisation model; embodied energy; genetic algorithm; particle swarm optimisation; harmony search algorithm; MATLAB; EC3.

ACKNOWLEDGEMENT

بِسْمِ اللَّهِ الرَّحْمَنِ الرَّحِيمِ

All praise and adoration to the Almighty Allah for his assistance and for letting me have the courage to overcome all the challenges I encountered throughout the period of my PhD journey.

I am also indebted to the Libyan government and, in particular, the Ministry of Higher Education for sponsoring me throughout the period of my studies.

I would like to express my deepest gratitude to my research supervisor, Dr. Fouad Mohammad, who has provided me with unlimited support throughout this project. His wisdom, knowledge, and commitment inspired and motivated me to push myself further and be the best I could be. I would also like to extend a huge thanks to my second supervisor, Dr. Hynda Kalalib, for her advice, support, feedback, and knowledge throughout the development of my thesis.

Especially heartfelt thanks and high respect are offered to my mother, the origin of my success, and my wife and children, who have participated with me in both my happy and sad moments. They shared the highs and lows of this seemingly epic journey with me and hardly ever complained during my absence.

My real thanks and appreciation must go to my friend Dr. Bubaker Al Shakmak for his endless support.

May Allah reward you immensely and bless you with good health and long life.

Contents

COPYRIGHT DECLARATION	I
List of Publications	II
Abstract	III
ACKNOWLEDGEMENT	V
Contents	VI
List of Figures	X
List of Tables	XIV
List of Symbols	XVII
List of Abbreviations	XXII
CHAPTER ONE. INTRODUCTION	1
1.1 Overview	1
1.2 Background and Motivation.....	2
1.3 Code for Structural steel Design	6
1.4 Aim and Objectives	6
1.5 Thesis Outlines	7
CHAPTER TWO. LITERATURE REVIEW	9
2.1 Introduction	9
2.2 Steel Material	9
2.3 Steel Frames	10
2.4 Analysis of Rigid Frames	13
2.5 Conventional Design of Steel Structures	14
2.6 Structural Optimisation	16
2.6.1 Applications of Optimisation	19
2.6.2 Structure of Optimisation Problems	20

2.7	Optimisation Techniques.....	25
2.7.1	Genetic Algorithm (GA) Technique	26
2.7.2	Particle Swarm Optimisation Technique.....	29
2.7.3	Harmony Search Optimisation Technique	32
2.8	Life Cycle Assessment of Steel Structures in Construction Stage.....	36
2.8.1	Cost Evaluation	36
2.8.1	Cost Model	38
2.8.2	Evaluation of Embodied Energy and Embodied Carbon	43
2.8.3	Embodied energy and carbon models	45
2.9	MATLAB Software.....	50
2.10	Review of Related Past Works.....	51
2.11	Summary	54
CHAPTER THREE. ANALYSIS AND DESIGN PROCEDURE.....		55
3.1	Introduction	55
3.2	Analysis.....	55
3.2.1	Analysis of 2D Structures	56
3.2.2	Analysis of three dimensional (3D) Structures	60
3.2.3	Programming the stiffness method.....	63
3.3	Design Procedure	64
3.3.1	Design Strength	65
3.3.2	Ultimate Limit State	70
3.3.3	Serviceability Limit State.....	90
3.4	Summary	93
CHAPTER FOUR. OPTIMISATION PROCEDURE		95
4.1	Introduction	95
4.2	General	96
4.3	Optimisation Model.....	97
4.4	Optimisation Techniques and Tools.....	99
4.4.1	Genetic Algorithm (GA)	100

4.4.2	Particle Swarm Optimisation (PSO)	103
4.4.3	Harmony Search A lgorithm (HAS).....	106
4.5	Optimum Design Formulation	109
4.5.1	Objective Function File	111
4.5.2	Design Constraints	118
4.6	Summery	121
CHAPTER FIVE. Validation of the Program and the Methods Used		122
5.1	Introduction:	122
5.2	Structural Analysis validation	122
5.3	Validation of the Design Optimisation.....	123
5.3.1	Two-dimensional, six-story, two bay rigid steel frame.....	123
5.3.2	Two-dimensional, ten-story, one bay rigid steel frame.....	128
5.3.3	Three-dimensional, five-story, two bays steel structure	131
5.3.4	Three-dimensional, ten-story, two X-bay and three Z-bay steel structure.....	134
5.4	Summary	137
CHAPTER SIX. NUMERICAL EXAMPLES.....		138
6.1	Introduction	138
6.2	Two-Dimensional Steel Framed Structures	138
6.2.1	Numerical examples	139
6.3	Three-Dimensional Steel Framed Structures	146
6.3.1	Three-bay three-story steel structure	147
6.3.2	Three-bay six-storey steel structure	152
6.3.3	Three-bay nine-storey steel structure	158
6.4	Analysis of life cycle assessment	167
6.5	Summary:	169
CHAPTER SEVEN. PARAMETRIC STUDY.....		170
7.1	Introduction	170
7.2	Parametric results	176

7.2.1	Group 1: Steel structures with grade (S355) subjected to variable action of 2 kN/m ²	176
7.2.2	Group 2: Steel structures with grade (S355) subjected to variable action of 4 kN/m ²	178
7.2.3	Group 3: Steel structures with grade (S355) subjected to variable action of 6 kN/m ²	181
7.2.4	Group 4: Steel structures with grade (S275) subjected to variable action of 2 kN/m ²	184
7.2.5	Group 5: Steel structures with grade (S275) subjected to variable action of 4 kN/m ²	186
7.2.6	Group 6: Steel structures with grade (S275) subjected to variable action of 6 kN/m ²	189
7.3	Discussion:	191
7.3.1	Effect of Steel Grade	191
7.3.2	Effect of the variable actions.....	192
7.3.3	Optimum design graphs for the structure for different variable actions.	194
7.3.4	Running time for optimisation	195
7.4	Summary	196
CHAPTER EIGHT. Conclusion and Suggestions for Future Work.....		197
8.1	Introduction	197
8.2	Contribution to knowledge.....	197
8.3	Conclusions	198
8.4	Recommendations for future work.....	200
References		202
APPENDIX-A		211
APPENDIX-B		223
APPENDIX-C		233

List of Figures

Figure 2-1: Hot rolled steel sections in practice (Dogan, 2010).....	10
Figure 2-2: Traditional design process (Coello et al., 2007) ... Error! Bookmark not defined.	
Figure 2-3: Optimal design process (Coello et al., 2007).....	17
Figure 2-4: Minimum of $f(x)$ is the same as maximum of $-f(x)$, (Rao, 2009).....	18
Figure 2-5: Optimum solution of $c f(x)$ or $c + f(x)$ same as that of $f(x)$ (Rao, 2009).....	19
Figure 2-6: Cantilever beam under concentrated load (Rao, 2020).....	21
Figure 2-7: Constraints in surfaces hypothetical two-dimensional design spaces (Rao, 2009)	24
Figure 2-8: Illustration of local and global minima (Rao, 2009).....	26
Figure 2-9: Steps of genetic algorithm process (Kameshki, 2003; Torregosa and Kanok- Nukulchai, 2002).....	28
Figure 2-10: Fish schooling (Dogan, 2010).....	30
Figure 2-11: Particle swarm process representation for discrete problems (Dogan, 2010).....	32
Figure 2-12: Life cycle energy and emissions of a building (UKGBC, 2014).....	45
Figure 3-1: (a) Representation of nodes and numbering of members (b) Displacements and member end forces in local coordinates (c) Displacements and member end forces in global coordinates.	57
Figure 3-2: Part of Stiffness Matrix assembly procedure in MATLAB R2020a.....	59
Figure 3-3: Segment of space frame element showing forces and displacements at the nodal coordinates	60
Figure 3-4: Force component in the local coordinates along with its components in the global coordinates	62
Figure 3-5: Analysis workflow chart	64
Figure 3-6: Moment rotation behaviour of cross-sections of different classes (Trahair et al., 2008)	68

Figure 3-7: Gross cross-section of a beam.....	69
Figure 3-8: Shear area of steel cross-section according to EC3	75
Figure 3-9: Design flowchart according to Eurocode-3	93
Figure 4-1: Automated workflow for the process of formulating the written program developed in this study	98
Figure 4-2: Schematic description for GA procedure.....	102
Figure 4-3: Schematic description for PSO procedure	105
Figure 4-4: Schematic description for HSA procedure.....	108
Figure 5-1: Six-story, two bay rigid steel frame	125
Figure 5-2: Search history for six-storey structure	126
Figure 5-3: Grouping rearrangement of six-storey, two bay rigid steel frame	127
Figure 5-4: Search history for six-storey structure after rearrangement of grouping	128
Figure 5-5: Representation of the layout of ten-story, one bay rigid steel frame.	129
Figure 5-6: Search history for 10-storey structure	130
Figure 5-7: Five-story, two bay rigid steel structure.....	132
Figure 5-8: Search history for five-storey structure.....	133
Figure 5-9: Ten-story rigid steel structure	135
Figure 5-10: Search history for Ten-storey structure.....	136
Figure 6-1: representation of the layout and loading of the three-storey frame	139
Figure 6-2: Comparison of Search history between GA, HSA and PSO for the three-storey frame (W-objective function)	141
Figure 6-3: Comparison of Search history between GA, HSA and PSO for the three-storey frame (EE-objective function)	142
Figure 6-4: Comparison of Search history between GA, HSA and PSO for the three-storey frame (Multi-objective function)	143
Figure 6-5: Schematic representation of the difference in weight among the three objective functions.....	145

Figure 6-6: Running time average for the three algorithms for the three objective functions.	146
Figure 6-7: representation of the layout and loading of the three-storey structure.....	147
Figure 6-8: Comparison of Search history between GA, PSO and HSA for the three-storey structure (EE-objective function).....	148
Figure 6-9: Comparison of Search history between GA, PSO and HSA for the three-storey structure (Multi-objective function).....	150
Figure 6-10: Comparison of minimum weight between GA, PSO, HSA and manual for 3 story structure among the two objective functions	151
Figure 6-11: Running time average for the three algorithms for the three optimisation methods.	152
Figure 6-12: representation of the layout ad loading of the six-storey structure	153
Figure 6-13: Comparison of Search history between GA, PSO and HSA for the six-storey structure (EE-objective function).....	154
Figure 6-14: Comparison of Search history between GA, PSO and HSA for the six-storey structure (Multi-objective function).....	156
Figure 6-15: Schematic representation of the difference in weight among the three objective functions.....	157
Figure 6-16: Running time average for the three algorithms for the three objective functions.	158
Figure 6-17: representation of the layout and loading of the six-storey structure	159
Figure 6-18: Comparison of Search history between GA, PSO and HSA for the nine-storey structure (EE-objective function).....	161
Figure 6-19: Comparison of Search history between GA, PSO and HSA for the nine-storey structure (Multi-objective function).....	162
Figure 6-20: Schematic representation of the difference in weight among the three objective functions for nine storey frame	163
Figure 6-21: Running time average for the three algorithms for the three objective functions for nine storey frames.	164

Figure 6-22: Comparison of weight obtained by three algorithms across the examples	165
Figure 6-23: Comparison of weight obtained from multiobjective optimisation by the algorithms for the three examples.....	166
Figure 6-24: Comparison in weight between single and multi-objective optimisation for the three algorithms across the three examples.	167
Figure 7-1: Layout and loading of the six-storey structure with (3X3) bay	Error! Bookmark not defined.
Figure 7-2: Optimum embodied energy for group 1.....	177
Figure 7-3: Search history for the optimal case in group 1.....	177
Figure 7-4: Optimum embodied energy for group 2.....	179
Figure 7-5: Search history for the optimal case in group 2.....	180
Figure 7-6: Optimum embodied energy for group 3.....	182
Figure 7-7: Search history for the optimal case in group 3.....	183
Figure 7-8: Optimum embodied energy for group 4.....	185
Figure 7-9: Search history for the optimal case in group 4.....	185
Figure 7-10: Optimum embodied energy for group 5.....	187
Figure 7-11: Search history for the optimal case in group 5.....	188
Figure 7-12: Optimum embodied energy for group 6.....	190
Figure 7-13: Search history for the optimal case in group 6.....	190
Figure 7-14: Comparison of optimum embodied energy between group 1 and 4	191
Figure 7-15: comparison of optimum embodied energy for all cases in group 1, 2 and 3 when steel grade S355.....	193
Figure 7-16: Embodied energy and Embodied carbon per sqm for the optimal cases 6X6 of group 1, 2 and 3 when steel grade is S355.....	194
Figure 7-17: Weight and Cost per sqm for the optimal cases 6X6 of group 1, 2 and 3 when steel grade is S355.....	195
Figure 7-18: Computational time for the optimisation process of all the cases in group 1...	196

List of Tables

Table 3.1: Input for the excel file.....	56
Table 3.2: Partial factor and load factor of safety (Ref: Concise Eurocodes, 2009)	66
Table 3.3: Nominal values of yield strength f_y and ultimate tensile strength f_u (EC3-EN 1993-1-1).....	67
Table 3.4: Determination of buckling curve for rolled steel cross-sections (EC3-EN 1993-1-1)	81
Table 3.5: Imperfection factors used for different buckling curves (EC3-EN 1993-1-1)	81
Table 3.6: Recommended imperfection factors for lateral torsional buckling curves (EC3-EN 1993-1-1).....	83
Table 3.7: Lateral torsional buckling curves recommended for different cross-sections (EC3-EN 1993-1-1)	83
Table 3.8: Recommended values of the coefficient C_1 for various moment conditions (EC3-EN 1993-1-1)	84
Table 3.9: Recommended values of k_c for various moment conditions (EC3-EN 1993-1-1).....	86
Table 3.10: Interaction factors recommended for members that are not susceptible to torsional-deformation (EC3-EN 1993-1-1).....	88
Table 3.11: Interaction factors recommended for members susceptible to torsional-deformation (EC3-EN 1993-1-1)	89
Table 3.12: Equivalent uniform moment factors C_m for members not susceptible to torsional (EC3-EN 1993-1-1)	90
Table 3.13: Recommended vertical deflection limits due to characteristic combination (EC3-EN 1993-1-1)	91
Table 3.14: Suggested limits for horizontal deflection (EC3-EN 1993-1-1).....	91
Table 4.1: Parameters used for the objective function.....	110
Table 5.1: The optimum solutions obtained by current study, (Isaa, 2010) and (Saka, 2007)	125

Table 5.2: The optimum solution after grouping rearrangement obtained by current study .	127
Table 5.3: The optimum solutions obtained by current study, Issa, (2010) and Camp, et al (2005).....	130
Table 5.4: The optimum solutions obtained by current study and Aydoğdu (2010)	133
Table 5.5: The optimum solutions obtained by current study, (Kaveh and Talatahari, 2012) and (Talatahari et al, 2015).....	136
Table 6.1: Optimisation output for the three-storey frame (W-objective function)	141
Table 6.2: Optimisation output for the three-storey frame (EE-objective function)	142
Table 6.3: Optimisation output for the three-storey frame (Multi-objective function)	144
Table 6.4: Optimisation output for the three-storey structure (EE-objective function).....	149
Table 6.5: Optimisation output for the three-storey structure (Multi-objective function).....	150
Table 6.6: Optimisation output for the six-storey structure (EE-objective function).....	155
Table 6.7: Optimisation output for the six-storey structure (Multi-objective function)	156
Table 6.8: Optimisation output for the nine-storey structure (EE-objective function).....	161
Table 6.9: Optimisation output for the nine-storey structure (Multi-objective function).....	163
Table 7.1: Fixed parameters.....	172
Table 7.2: Variable parameters	173
Table 7.3: Details for the groups classification	175
Table 7.4: Case details and optimum results for group 1	176
Table 7.5: Optimisation output for the optimal case in group 1	178
Table 7.6: Case details and optimum results for group 2	179
Table 7.7: Optimisation output for the optimal case in group 2	181
Table 7.8: Case details and optimum results for group 3	182
Table 7.9: Optimisation output for the optimal case in group 3	184
Table 7.10: Case details and optimum results for group 4	184
Table 7.11: Optimisation output for the optimal case in group 4	186

Table 7.12: Case details and optimum results for group 5	187
Table 7.13: Optimisation output for the optimal case in group 5	188
Table 7.14: Case details and optimum results for group 6	189
Table 7.15: Optimisation output for the optimal case in group 6	191

List of Symbols

A	Gross cross-sectional area
$A_{c,eff}$	Effective section area
A_i	Area of beam section of the i^{th} group
A_j	Area of column section of the j^{th} group
A_{pl}	Total steel surface area
A_v	Shear area of steel sections
\bar{b}	Appropriate width of the cross-section
C	Width of outstand flange
c_{ere}	Erection cost factors
c_f / t_f	Ration for local buckling in flange
c_w / t_w	Ration for local buckling in web
C_{ws}	On-site welding cost factor
Cc	Corrosion cost factor
D	Distance travelled in km (100km)
$d_{T,F}$	Total distance travelled from fabrication site to construction site
$d_{T,N}$	Total distance travelled from factory to fabrication site
d_p	Total distance travelled from factory to fabrication site
d_s	Total distance travelled from fabrication site to construction site
E	Modulus of elasticity of the section
E_2	Transportation energy
E_{LP}	Consumption factor during loading and unloading
E_t	Transportation energy
E_T	Transportation energy

EC^A	Energy consumption factor for hoisting crane
EC^P	Factor for the embodied energy
ECF	Energy consumption factor
$EE1$	Embodied energy of steel elements during production
$EE1$	Embodied energy of paint coat
EC^T	Consumption factor for transportation
EE_e	Embodied energy during erection
EE_i	Factor for the embodied energy
EET_i	Consumption factor for transportation
EE_m	Embodied energy resulting from materials
EE_T	Total embodied energy of the structure
EE_t	Embodied energy during transportation
f_k	Consumption factor for transportation
f_n	Emission factor
f_y	Steel yield strength
f_{yf}	Flange's yield strength
G	Shear modulus
GHP	Gross horsepower of equipment
h_w	Depth of the web
i	Radius of gyration
I	Second moment of area
I_p	Factor for the embodied energy
I_T	Consumption factor for transportation
K	Amount of fuel consumed per brake
$k_{m.paint}$	Paint coat's cost

k_c	Correction factor
k_{erect}	Erection cost factors
k_{paint}	Cost of labour and equipment for paint
k_{transp}	Transportation cost factor
k_τ	Buckling coefficient
$k_{zy}k_{zy}k_{zy}k_{zy}$	Interaction factors
k_σ	Factor for the buckling
KPL	Fuel density
L	Span length
l_i	Sections' length of the section
l_{ib}	Length of beam element b in group i
l_{jc}	Length of column element c in group j
l_{ws}	Weld length
LF	Load factor in percentage
$LMPH$	Litre used per hour of the equipment
m	Sections' mass
$M_{b,Rd}$	Design buckling resistance moment
$M_{c,Rd}$	Design resistance for bending of cross section
$M_{cb,z,Rd}$	Design bending resistance in the minor axis
$M_{el,Rd}$	Design elastic resistance moment
$M_{f,Rd}$	Flange's bending moment resistance
$M_{pl,Rd}$	Design plastic resistance moment
$M_{y,Ed}$	Design maximum moment about (y-y)
$M_{z,Ed}$	Design maximum moment about (z-z)
M_{cr}	Elastic critical buckling moment

M_{Ed}	Design bending moment
M_{paint}	Paint quantity used
M_{str}	Structure's mass
mh	Man-hour of fire protection
n	Number of distribution centres
$N_{b,y,Rd}$	Design buckling resistance about y-y axis
$N_{b,z,Rd}$	Design buckling resistance about z-z axis
$N_{c,Rd}$	Design compressive resistance of the cross-section
N_b	Total number of beams in group i
N_{Ed}	Design value of compression force
N_{sb}	Total number of beam groups
N_c	Total number of columns in group j
r	Root radius
R_r	Total amount of resources or energy usage
T	Amount of time for lifting and installation process
t	thickness of the flange or web
T_c	Transportation cost factor
T_{erect}	Total erection time
t_f	Flange thickness
T_{paint}	Painting time required for painting
t_w	Web thickness
ν	Poisson ratio in elastic stage
$V_{bf,Rd}$	Flange's buckling shear resistance
$V_{bw,Rd}$	Buckling shear resistance of the web
$V_{c,Rd}$	Design shear resistance

$V_{pl,Rd}$	Plastic design shear resistance
V_{Ed}	Design shear force
V_{el}	Elements' volume
V_{struct}	Structural elements' volume
W	Weight of the structure
$W_{el,min}$	Minimum elastic section modulus
W_{el}	Elastic section modulus
W_{pl}	Design plastic section modulus
W_y	Section modulus
Wf	Percentage of waste (5%)
α_{LT}	Imperfection factor due to torsional-flexural buckling
δ_h	Horizontal deflection
δ_{tip}	Tip deflection of the beam
δ_v	Vertical deflection
ε	Factor depending on steel strength
ε_i	Waste factor
η	Coefficient value
$\bar{\lambda}_p$	Factor of stress ratio
$\bar{\lambda}_w$	Modified web plate slenderness
ρ	Density of the sections
τ_{cr}	Critical elastic local buckling stress
χ_{LT}	Reduction factor due to torsional-flexural buckling
χ_w	Web contribution to shear buckling factor
\tilde{y}_{pi}	Total steel surface area

List of Abbreviations

ASD	Allowable Stress Design
ACO	Ant Colony
pc	Crossover probability
DE	Demolition Energy
DSM	Direct Strength Method
EC	Embodied Carbon
EE	Embodied Energy
EC3	EuroCode-3
FEA	Finite Element Analysis
FER	Fixed End Reaction
GRG	Generalised Reduced Gradient
GA	Genetic Algorithm
GHG	Greenhouse gases
HMCR	Harmony memory considering rate
HMS	Harmony memory size
HSA	Harmony Search Algorithm
IEE	Initial Embodied Energy
ICE	Institution of Civil Engineers
LCA	Life Cycle Assessment
LRFD	Load and resistance factor design
MI	Mixed-Integer
pm	Mutation probability
OC	Operational Carbon
OE	Operational Energy

PSO	Particle Swarm Optimisation
PCR	Pitch adjusting rate
REE	Recurring Embodied Energy
SQP	Sequential Quadratic Programming
SUMT	Sequential Unconstrained Minimisation Technique
SLS	Serviceability limit state
SA	Simulated Annealing
SCI	Steel Construction Institute
ULS	Ultimate limit state
UKB	Universal steel beams
UKC	Universal steel columns

CHAPTER ONE. INTRODUCTION

1.1 Overview

The design process ensures that a given structure satisfies the architectural requirement, on one hand, and maintains a safe, serviceable, and durable life span, on the other hand (McKenzie, 1998). For the design of a simple structure, the usual practice is to follow the intuition and experience of the structural engineer. However, this may not be enough to obtain an optimal solution for a large structure due to the complexities involved, especially when there are different load case scenarios and design variables experienced by the structure. The design of a large structure usually requires that many criteria are considered, because their influences on the response of the structure tend to change if there is a slight alteration in the members' properties. As such, structural design optimisations are often engaged to find the best satisfactory performance with minimum effort when designing large and complex structures. Meanwhile, the increasing advancement in optimisation techniques does not correspond to its practical applications in structural engineering (Cohn and Dinovitzer, 1994).

Structural optimisation has been a topic of interest to researchers for more than 100 years. It originated from the early works of Maxwell (1890) and Mitchell (1904) cited in (Akin and Arjona-baez, 2001). The 1940s and 1950s for the researchers involved considerable analytical works on component optimisation. By the 1970s, some of the researchers produced a comprehensive statement on the application of mathematical programming methods in solving the non-linear and inequality constrained problem of designing structures under a varied loading condition. Notable among them were Schmit and Farshi (1974) cited in (Akin and Arjona-baez, 2001). Advancement in the mathematical programming method was achieved in the 1980s when some algorithms that can use the finite element analysis method for optimisation were developed.

In recent years, a significant number of novel and innovative optimisation techniques that have been successful to various degrees have been developed. The basis for most of them being stochastic search which uses the idea of simulation of natural phenomena. They are functionally and structurally simple to use in practice. However, they are slow-process methods and thus require some changes to speed up their performance (Karlaftis, 2015). Additionally, rapid growth of the domestic personal computer over the past years has further strengthened

the motivation to formulate design problems using one of the stochastic optimisation techniques and to ensure their practical application in structural engineering. It is needless to say that reducing the computation time and ensuring that the optimisation technique is robust enough to obtain global optimum solutions is paramount. Consequently, it is necessary to embark on more studies to modify the available optimisation techniques so that they will be very effective in optimising real size engineering problems in the offices of structural engineers.

1.2 Background and Motivation

The use of structural steel for residential and commercial buildings is gaining more popularity worldwide due to its speedy construction. The high strength to weight ratio and durability of the steel, makes steel buildings the selected option when open space having long spans without intermediate columns is required. It is even more preferred in areas with high earthquake occurrences (Barraza et al., 2017). However, the rise in the demand for steel in construction has led to a corresponding increase in energy consumption from the environment as well as carbon emissions into the atmosphere.

Structural optimisation serves as a means of exploiting the full functionality of structural systems while minimizing the inputs such as embodied energy, material weight, cost etc. Optimisation can ensure that the best option adopted, sufficiently satisfies the required criteria (boundary conditions). Common optimisation targets (objective function) are cost, aesthetic outlook of structure, weight etc. However, the concern to reduce the negative environmental footprints of buildings over the last decades has intensified. Such footprint includes the reduction of the amount of non-renewable embodied energy used in construction.

The embodied energy and embodied carbon of building materials are respectively used to quantify the energy consumed and carbon dioxide emissions generated, through sourcing, manufacturing, and using construction materials and component (IStruct, 2000). This energy is estimated by conducting a complete Life Cycle Analysis (LCA). LCA for buildings encompasses the energy consumption and greenhouse gas (GHG) emissions involved in material extraction, production, transportation, assembly, operation and demolition, in order to assess their effects across the different life cycle phases (Scheuer et al., 2003).

Statistics has shown that half of the UK carbon emissions originate from energy used in constructing and operating buildings (BREEAM, 2012; cited by (Msabawy, 2017). Ordinarily, energy is most consumed in the operation phase of the life cycle of a structure (Domingos et al., 2015). However, recent findings show that the operating energy accrued over 15 years for a normal energy use building is equivalent to the embodied energy consumed during the building's life cycle. It can also be higher for buildings with low energy usage (Milne, 2013). Consequently, embodied energy phase may become the lead contributor in subsequent years due to the increase in the construction of low energy buildings enforced by strict legislations, borne out of the concern for the increase in the negative environmental impacts from the operational phase.

In addition, steel buildings have been found to be higher in embodied energy when compared with other building materials such as concrete, although they may be lower in embodied carbon (Zhou and Azar, 2018). Therefore, the embodied energy and carbon of steel structures must be taken into account during decision-making and design stages in order to minimize the environmental impact on human health and the environment.

It is expected that the embodied carbon from human activities will continue to grow if it is not adequately checked. This was demonstrated from the UN panel report on climate change whereby an 80% increase dating back to the pre-industrial times in the 1970s was recorded (Camp and Huq, 2013). Presently, the built environment contributes 30% of total GHG emissions in the UK. Thus, to achieve the target of a 50% reduction by 2025, a reduction of 21% must be achieved in the built environment sector (Victoria, 2016). Although, using sustainable materials can help to curb the negative environmental impact, however, optimisation of materials and processes can bring about efficient use of the desired material in an environmentally friendly manner. In the same vein, optimisation of the structural design of steel seems to be the best possible way of reducing the environmental impact of steel as not much can be done in its manufacturing process unlike concrete (Cullen et al., 2012).

Economic utilization of resources can be ensured when consideration is given to the reduction in the cost of structures. Whilst studies on the simultaneous optimisation of structures (from other building materials) for cost and embodied energy is gaining traction (Camp and Assadollahi, 2013; Camp and Huq, 2013; Eleftheriadis et al., 2018; Oh et al., 2019), less attention is given to steel structures. Therefore, this study seeks to achieve weight and

embodied energy optimisation of two- and three-dimensional steel frames. Weight minimisation is being used to replace cost minimisation here due to the findings from previous researches that there is a linear relationship between them (Marwaha, 2017; Pavlovčič et al., 2004). Minimum embodied carbon was also obtained from the optimisation of embodied energy of the frames. Consequently, the minimum weight of the steel frame is observed from the point of view of embodied energy and that of embodied energy is observed from the point of view of weight. Single and multi-objective functions were also investigated to obtain a possible balance between the two items.

Conversely, the trade-offs possible between weight and embodied energy is of interest during decision making to prevent a situation whereby the reduction in one leads to the increase in the other. Thus, it is required that the relationship between weight and embodied energy is established. Optimisation can be used to either find the best option for a single objective or for more than one objective (multi-objective function). Thus, this research employs both the single-objective and multi-objective optimisation for embodied energy and weight to address both functions individually and simultaneously so that a balanced solution is obtained.

Optimisation is achieved through different techniques that are usually classified under numerical, analytical, and heuristic methods (Kaveh and Ghazaan, 2018). Although the numerical and analytical methods are exact, however, they are only used for simple engineering problems as they may become too complex for real-size engineering problems. Hence, the technical development of the adopted purpose was to derive Mixed-Integer (MI) linear (or quadratic) model, interfaced through MATLAB toolbox called YALMIP. This would ensure the use of state-of-the-art commercial numerical solvers such as Gurobi and Mosek. However, during the testing phase, the resulting model was Mixed-Integer with bilinear and nonlinear constraints. In such circumstances, the only way to solve such problems is through black-box approaches. Since it is against core idea of YALMIP, the meta-heuristic-based methods also known as stochastic methods were chosen. The meta-heuristic approaches also known as stochastic methods are based on bio-inspired and artificial intelligence techniques. The solutions are approximate, but the approaches are suitable for real-size engineering problems. Their use can increase the chance of quickly obtaining global optimum through a random selection of variables, which are usually near the optimum (Aydoğdu et al., 2012; Dogan, 2010; Messac, 2015).

In the recent time, the widely adopted optimisation methods are Genetic Algorithm (GA), Simulated Annealing (SA) Particle Swarm Optimisation (PSO), Ant Colony (AC), Harmony Search Algorithm (HSA), and many more (Charalampakis and Tsiatas, 2019; Dokeroglu et al., 2019). The two oldest and most used among them are the GA and PSO methods (Luthra et al., 2017). This study thus focused on these two as they represent two different phenomena: evolutionary strategy and swarm intelligence. The existence of many algorithms can make it difficult in choosing the “proper one”. In (Ravat et al., 2021) opinion, newer “novel” algorithms lack scientific rigor and maturity with their development seeming doubtful. As such, many studies decided to only adopt GA and PSO. Additionally, HSA was considered as an augmentation to evolutionary algorithm GA. Typically, the generation of offspring in the gene pool involves the union of two parents' genes. HSA in its approach permits the whole population at once, to contribute to gene pool. The simplicity of implementing HSA as well as the presence of other desirable features made HSA to be considered as a third option. This study will therefore further develop and validate the performance of these methods for single and multi-objective optimisation of steel structure.

Moreso, the study will use built-in function in MATLAB for the implementation of the optimisation methods. The popularity of MATLAB for computational modelling as well as the several algorithms embedded in it makes it a good option to be adopted. It also allows for the integration of the analysis procedure (direct stiffness method) for the frames. The integration of the analysis is to investigate the viability of the simple approach in obtaining optimum solutions as against developing a software tool based on a more complex procedure (e.g. FEA) or using already developed software like open sees which is the usual practice in previous researches of; (Cazacu and Grama, 201; Fathizadeh et al., 2021; Plevris et al., 2011; Soori and Salajegheh, 2019; Yassami and Ashtari, 2015)

In this research, Genetic Algorithm (GA), Particle Swarm Optimisation (PSO) and Harmony Search Algorithm (HSA) methods developed in MATLAB are employed to examine their accuracy of the results and computational time required in single and multi-objective problems, for two and three-dimensional steel structures. Moreover, optimisation models have been developed for the evaluation of cost, embodied energy and embodied carbon for steel structures, to aid designers in assessing their energy consumption and carbon impacts due to their designs, during the initial stages.

1.3 Code for Structural steel Design

Structural design codes are design guides established to ensure consistency in design for each type of structure in a particular region. Fundamentally, it serves as a check tool for authorities to ensure that structural designs are done appropriately to prevent casualty (Galambos, 2006). The European Standard Code of practice (EC3) is the most commonly used design code for steel structures in the UK and all other European countries. Furthermore, some other countries outside Europe have shown interest in adopting this code. For this main reason, Eurocode 3 (EC3) is adopted for design in this study.

In addition, the adoption of EC3 was also due to its broad utilisation in the region where the research been carried out as well as its predominance as the code for structural design of steel throughout the EU. According to (Brown and Elms, 2007), EC3 can give economical designs and still maintain the structural stability of steel structures. Therefore, many regions in the world are deliberating on its adoption in their area. Thus, the use of EC3 in this study is worthwhile considering the foregoing.

1.4 Aim and Objectives

This study aims to investigate the single and multi-objective functions of weight and embodied energy optimisation for two-dimensional (2D) and three-dimensional (3D) steel structures using three different optimisation techniques which are genetic algorithm (GA), particle swarm (PSO) and Harmony search (HSA). In order to achieve the aim of this research, the following objectives were established:

1. Carrying out a desk study on steel frame analysis and design, optimisation techniques, and current state of optimisation on weight, cost, embodied energy, and embodied carbon.
2. Developing Life Cycle Analysis (LCA) assessment models for evaluation of cost, embodied energy and embodied carbon for steel structures.
3. Develop a program designed to optimise any type of plane and space steel structures. The program is to involve all aspects of the developed algorithms. It should be compatible with available personal computer so that it may be used by structural engineers.

4. Comparison of results with literature in order to validate the already developed MATLAB code, which is an integration of analysis and optimisation for 2D and 3D structures.
5. Investigate and evaluate the influence of different optimisation methods on the design optimisation required.
6. Investigate and evaluate the influence of different objective functions on the performance of the design optimisation methods.
7. Conducting a parametric study to investigate the optimum 2D and 3D steel structures considering weight, cost, embodied energy, and embodied carbon.
8. Establishing optimum design charts to assist designers, during initial stages, in finding the most sustainable solution.

1.5 Thesis Outlines

The research includes eight chapters. A brief description of the chapters' contents is presented below:

- Chapter 1: Introduction: A general overview on structural design optimisation was presented. Then, justification of research, aims and objectives are stated in the chapter.
- Chapter 2: Literature Review: this chapter presents the revised literature related to steel structures, introduction to optimisation, optimisation techniques as well as optimisation of steel structures.
- Chapter 3: Structural Analysis and design procedure: this chapter will describe the structural analysis method and design code of practice used in this study and their integration in MATLAB.
- Chapter 4: Optimisation Process: this chapter presents the developed optimisation methods, tools utilised in this research as well as the models developed for evaluation of embodied energy, embodied carbon, and cost.

- Chapter 5: Validation: this chapter presents the evaluation and validation of the self-coded optimisation program as well as benchmarking of the functions.
- Chapter 6: Numerical Examples: this chapter widely investigates and discussed the performance of the used methods in single and multi-objective functions for two and three-dimensional structures.
- Chapter 7: Parametric study: this chapter involves an extensive analysis and discussion of the results obtained from a parametric study undertaken.
- Chapter 8: Conclusions and Recommendations: this chapter summarises the findings and states the main points extracted from the results. In addition, recommendations for future work are also provided.

CHAPTER TWO. LITERATURE REVIEW

2.1 Introduction

The purpose of this chapter is to develop an understanding of the existing knowledge of the design of steel-framed structures using optimisation techniques. It entails a comprehensive review of steel structures, conventional design of steel structures, structural optimisation, and its techniques. This chapter will also undertake a wide review of the existing objective function formulas that represent the life cycle assessment of steel structures. The objective functions considered are embodied energy, embodied carbon, and the cost of two and three-dimensional steel structures. Finally, a critical review of some previous studies relevant to the study was presented.

2.2 Steel Material

The construction business before the eighteenth century was familiar with the utilization of wood (timber) and stonework (masonry) as building materials. After the production of cast iron, it was recognized to be used a high load bearing structural element such as beams and columns. With the passage of time and more research studies showed more confidence on use of wrought iron for structural element due to its high compressive, flexural and tensile strength as compared to cast iron. Later on, use of cast iron and wrought iron was reduced due to their brittle behaviour and they were replaced by steel because of more ductility, strength and economical.(Mckhenzie, 2004).

Steel is now successfully using as an effective and effective building material combined with state-of -the art design and analysis techniques for modern construction over the time. Numerous structures like the Eiffel tower – Paris, guaranty building- Buffalo New York etc. are an example of steel structures. The prefabricated units also gained importance in light weight and speedy construction to save the overall cost of entire project like the 1000-bed hospital that was made in Brunel and shipped to Crimea (Martin and Purkiss, 2008; Mckhenzie, 2004).

Steel has been used in various forms like cold form and hot rolled sections as shown in Figure 2-1. The hot rolled section was found to be more affordable and economical which used in connection with bolts and rivets. (Mckhenzie, 2004).

The quantities of silicon, carbon, manganese, and other elements added after impurities are removed from molten pig iron determine the type of steel produced and its qualities. Mild steel, for example, is low in carbon but high in ductility. High carbon steel has more strength but is less ductile, whereas high yield steel has equal stiffness as mild steel but has more strength. Mild steel is the most common, cheapest to make, and easiest to weld of all of them.(Megson, 2019).

Due to its high ductility, steel with reduced carbon content is used to manufacture cold-formed sections, which can be bent cold as opposed to hot rolled steel, which is formed at high temperatures up to 1400C. This form of steel is light, flexible, and simple to install. Their application as structural members, such as in portal frames, is gaining popularity (Msabawy, 2017).



Figure 2-1: Hot rolled steel sections in practice (Dogan, 2010).

2.3 Steel Frames

The structure of a building is frequently thought of as an assembly of planar frames when it is being designed. Beams and columns are linked at their joints to form plane frames. Multi-story building frames support the floors, roofs, external walls, or claddings while also resisting wind

loads. Each of these elements transmits load to the frame in the form of wind, snow, or people, floor coverings, and furniture (live load), which is subsequently communicated to the building's foundations, which support the entire structure. (Megson, 2019).

Moment frames have strong joints, which contribute to their stability. The principal internal member forces due to the loads applied throughout their length include shear forces, bending moment, and axial force. They are quite simple to install because they simply require welding or bolting to join beams and columns. (Williams, 2009).

Steel-framed structures are well-known, and their application in complex designs is continuously increasing. They are even more popular because of their simplicity and ease to builder they are to build and how quickly they can be completed. Long spans without columns are possible with steel buildings, allowing for a huge open space. This might make future service integration and building use changes easier. Steel's excellent strength-to-weight ratio allows for minimal material consumption while maintaining a sturdy construction. As a result, resource efficiency is improved, and poor soil can be used as a foundation (SCI, 2008).

Steel-framed structures also have the advantage of being able to use prefabricated elements on site. This can result in safer working circumstances in terms of sufficient safety measures during manufacture as opposed to on-site, as well as weather protection. It will also speed up the construction process, decreasing interruption to nearby structures. It might lose its rigidity and strength during a fire if certain precautions are not taken.(Robb, 1972).

Framing of steel structures determines the type of stability they possess. Three classes are permitted in Euro code for steel structures (EC3). They are:

- **Simple framing:** All joints are pinned in this kind of framing system, which makes it statically determinate. Stability will be essential because they are vulnerable to lateral loads. Bracing, shear walls, or core walls are used to provide stability for this frame class. Only shear force is transmitted between the beam ends and the column since the beam to column connection is believed to be nominally pinned. This form of frame is cost-effective and relatively simple to build and analysis (IstructE, 2000) and (SCI, 2008).
- **Semi-continuous framing:** This type of framing system presupposes that the joints are rigid and that the moment rotation behaviour can be reliably predicted. Because they

also transfer moment, they are sometimes known as semi-rigid frames, and their analysis is comparable to that of fully rigid frames. The analysis differs in that it takes into account rotation at the joints utilising moment rotation curves to avoid stiffness loss and increased deflection (IstructE, 2000).

- Fully continuous framing: The term "rigid connections" between beams and columns is used here, and it is considered that the fully rigid joints give continuity. A whole end depth plate can give the required stiffness. To properly resist the applied moments, this form of frame necessitates a somewhat deep beam and an appropriately large column. Because no rotation is predicted, the column is designed to take the full bending force from the beam when full end connection constraint is assumed (Dogan, 2010). The critical forces required in design of members are determined for the continuous beams and frames with negligible sway effects under the applied loading (IstructE, 2000).

Numerous studies have been conducted in comparison between these steel frames and other types of frames.

Yassami and Ashtari (2015) conducted a study and compared rigid and semi rigid steel frames. It was concluded that Semi rigid frames can produce lighter frames with higher displacements than rigid frames. The weight savings in the tall frame examined were similarly found to be minimal when compared to the higher displacements.

The research by Lim and Nethercot (2002) nearly parallel that of Yassami and Ashtari (2015) in their comparison of rigid and semi-rigid portal frames. Semi rigid portal frame having longer span (14m) but lower height (3m) was found to be 14% more efficient than rigid portal frame. While that having shorter span (8m) and more height (6m) was 36% less efficient. The findings therefore show that when using semi-rigid frames, the height of buildings is a significant issue. Because of the higher bending forces present, rigid frames are prone to produce heavier constructions.

Furthermore, Fathizadeh et al (2021) investigated the effect of soil interaction on moment frames cross sections. When it came to lowering soil stiffness, it was discovered that heavier sections would be required. This conclusion supports the idea that a stiffer construction is necessary to withstand any failure caused by the soil.

2.4 Analysis of Rigid Frames

EC3 considers two methods of analysis for a structure; Elastic global analysis and plastic analysis.

- 1) Elastic global analysis: In this method, elastic analysis is used to calculate the moments and forces in the members of the frame considering both ultimate and serviceability limit states.
- 2) Plastic analysis: Plastic behaviour of the sections governs this analysis and only ultimate limit state is considered. It can be rigid plastic method or elastic plastic methods (IstructE, 2000).

The direct stiffness method is a kind of elastic analysis that has been proven to be useful for analysing indeterminate structures. To calculate the member end displacements, it employs the concept of section properties in the form of a stiffness matrix (Nagarajan, 2018). It forms the basis for most finite element analysis (FEA) based software. However, it does not need discretizing of the members as they can be divided directly following the structural layout.

Chen and Chen (1999) conducted a research study and concluded that structural engineers are using the stiffness approach to incorporate connection behaviour in the analysis and design of frames because of the efficiency and ease with which the requisite matrices can be constructed inside computer processes.

However, the use of FEA method and modern software has led to the reduction in the use of direct stiffness method. Most researches on optimisation would rather use a finite element software incorporated into the optimisation model or directly include FEA in the algorithm even when the problem is simple. For example, Plevris et al (2011) developed a FEA based software for analysis to achieve the weight optimisation of steel frames. Cazacu and Grama (2014) integrated FEA method into MATLAB for truss optimisation. He used Opensees software, based on FEA method, in conjunction with MATLAB for optimisation of rigid and semi-rigid steel frames. Soori and Salajegheh (2019) also integrated Opensees software with MATLAB for weight minimization of steel frames.

As a result, this study would test the suitability of using direct stiffness matrix developed as part of an optimisation algorithm for rigid frames.

2.5 Conventional Design of Steel Structures

Traditional design procedure as shown in **Error! Reference source not found.** illustrates that the designer has to verify problem requirements by mathematical analysis. If such requirements are not satisfied, then dimensions and/or reinforcement of the elements are changed, and a new solution is performed based on engineering perception. This repeated process consumes considerable time, until a suitable section is found (Coello et al., 2007).

Structural design involves the determination of the weight, size and profile of the various elements that describe the structures of the built environment. It is assumed that the resultant structure will properly support the loads for which it was designed, be cost-effective, and appropriately perform the intended function for the duration of its life. As a result, it is the structural engineer's obligation to guarantee that the design code's critical specifications are followed so that the structure meets the above standards.

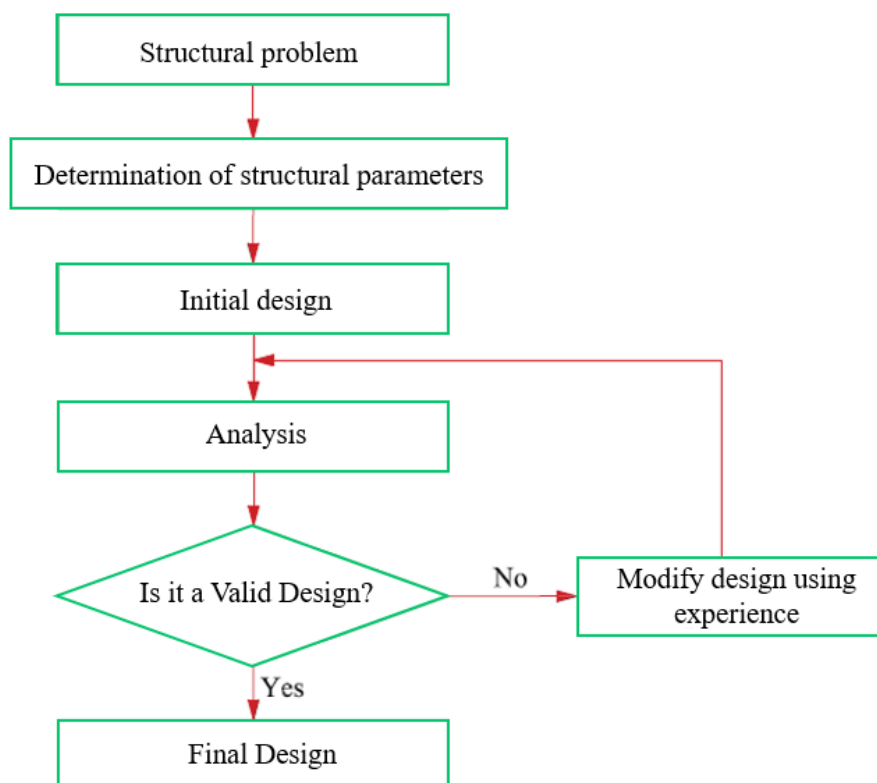


Figure 2-2: Traditional design process (Coello et al., 2007)

The three major design approach are the limit state design, load and resistance factor design (LRFD) and allowable stress (elastic) design (ASD). The Euro code (EC3) adopted here in this

study is based on the limit state design where ultimate and serviceability states are considered. This means the conditions that can lead to failure or unserviceability of the structure is taken into account by applying different factors of safety to loads and materials.

Aghoury et al (2020) investigated at the suitability of three different design methodologies for calculating the compressive strength of cold formed steel columns. To determine the compressive strength of the columns, the researchers used an experimental and numerical approach. The results were compared to the ASD approach, which used the AISI-2007 code, the limit state design, which used the Euro code, and finally, the direct strength method (DSM). The codes produced values that were equivalent to what was found; however they were slightly higher than the numerical analysis results. Both Euro code and AISI yielded nearly identical results, with DSM's being somewhat higher.

Issa (2010) on the other hand compared BS5950 and EC3 which are based on limit state design. He found that EC3 gave lighter frame when the frame is subjected to gravity and lateral loads. This was stated to be due to the lower load factor for load combination specified in EC3. However, no difference was observed in the case of concentrated lateral load because of the control of the optimum solution by lateral displacement which depends on working loads.

According to Torregosa and Kanok-Nukulchai (2002), three major steps are followed in the design of steel structures in actual practice. They are:

- 1) Modelling of the structure and analysis of its members
- 2) Designing of the members using the structural analysis results
- 3) Final analysis of the structure based on the designed members.

However, the processes outlined above do not guarantee optimal design. In order to derive the stresses, the first stage necessitates the assumption of member sizes. The members of the frame are then designed using these stresses. The final assessment of the study using the planned members will disclose a new set of stresses for which some members will be insufficient. However, just the sections that break the criteria are replaced, and new sections are chosen to replace them. As any segment that satisfies the constraints can be picked, this will result in conservative numbers, as the overall combinations of these sections may over-satisfy the criteria. As a result, before an optimal design can be achieved, members assumed during the

design process must be considered. i.e. stressed to the nearest maximum allowable (Torregosa and Kanok-Nukulchai, 2002).

2.6 Structural Optimisation

Optimisation is a mathematical method for determining the best use of a resource. As a result, optimisation techniques can assist a designer in arriving at the optimal solution while ensuring that the structure meets its functional criteria (Kaveh and Ghazaan, 2018). All engineering challenges, such as the design of civil engineering infrastructures, aircraft and aerospace structures, and so on, can benefit from optimisation (Phan et al., 2012). The structural engineering goal of design optimisation can be to achieve a minimum weight, most economic structure, most sustainable option, or their combination (Kaveh and Ghazaan, 2018).

According to Kaveh and Ghazaan (2018), optimisation challenges can include determining the best size (sizing optimisation) or shape (shape optimisation) of a structural element, as well as determining the optimal size and connection between structural members (topology optimisation). In order to solve an optimisation problem, three main steps should be followed:

1. Identification of the parameters of the problem
2. Setting of the objective function and constraints
3. Choosing a suitable optimiser to solve the problem

The optimisation process, as defined by Rao (2009) and Cicconi et al (2016), is the act of finding the best feasible results for a given problem under certain conditions. This could be for the purpose of increasing or decreasing the objective function (which might be weight, cost or merit function). Furthermore, Prakash et al (1988) defined optimisation theory as a process in which mathematics and numerical approaches are utilised to discover the best solution among a set of alternatives for a given problem. Schematic idea of the optimal design process is presented in Figure 2-3.

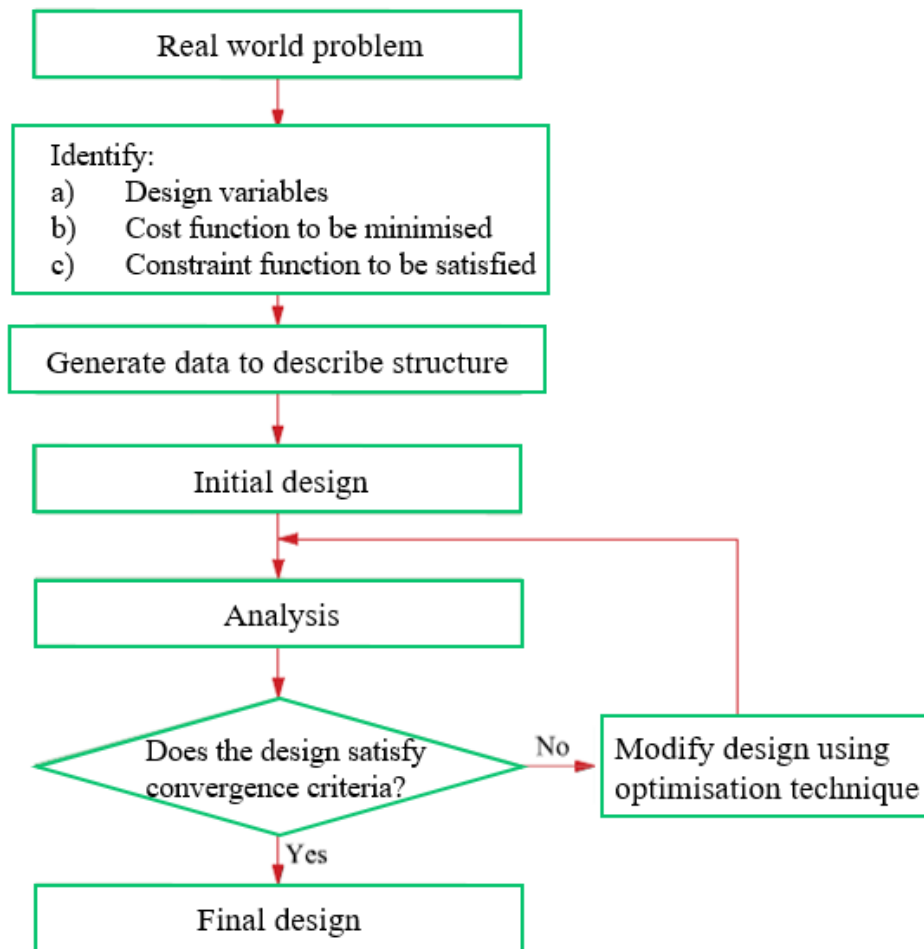


Figure 2-3: Optimal design process (Coello et al., 2007)

Any structural optimisation problem, according to Belegundu and Chandrupatla (2011), should be formulated to meet the requirement of sustainability while also considering the structure's life cycle, which includes ease of implementation, maintenance requirements, and disassembly at the end of the cycle.

Adeli and Sarma (2006) added to Belegundu and Chandrupatla's study by emphasising that, independent of profits derived from the employment of optimisation techniques, the environment has a significant portion of this. Because the goal of this strategy is to reduce the cost or weight of a structure, less material will be used in the construction. This will help to reduce the amount of energy used to create these materials.

However, the focus of this research is on determining the best design for 2D and 3D steel buildings in terms of weight and embodied energy. The optimisation algorithm will take into account three major factors: materials, transportation, and erection.

Engineering is built on the principle of optimisation in construction and maintenance. Civil engineers are responsible for making technological and administrative judgments at various levels in order to improve design processes and produce less expensive systems. The ultimate purpose of engineers' decisions is either to maximise the desired benefit (e.g. maximum capacity, profit) and to minimise the effort required (e.g. minimum weight, volume and/or time) (Karlaftis, 2015).

Since, the effort required and the benefit desired are expressed as a function of certain decision variables in any practical situation, the optimisation was defined by Rao (2009) as a process of finding the conditions that give the minimum or maximum value of function. Figure 2-4 demonstrates that if a point x^* corresponds to the minimum value of function $f(x)$, the same point also corresponds to the maximum value of the negative of the function, $-f(x)$. Thus without loss of generality, optimisation can be taken to mean minimisation since the maximum of a function can be found by seeking the minimum of the negative of the same function (Rao, 2009).

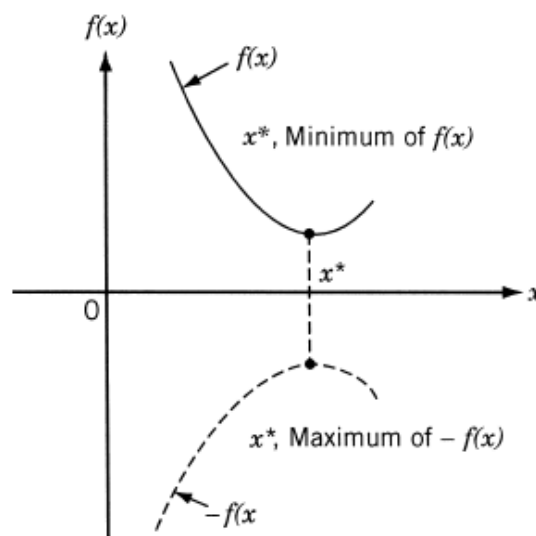


Figure 2-4: Minimum of $f(x)$ is the same as maximum of $-f(x)$, (Rao, 2009).

In addition, Figure 2-5 clearly shows some operations on the objective function, which will not affect the optimum solution (x^*). This can lead to conclusion that the optimum solution results from the multiplication of $f(x)$ by a positive constant (c) will lead to the same optimum solution results from $f(x)$. Also, the same optimum solution will be obtained when a positive constant (c) added to $f(x)$ as shown below.

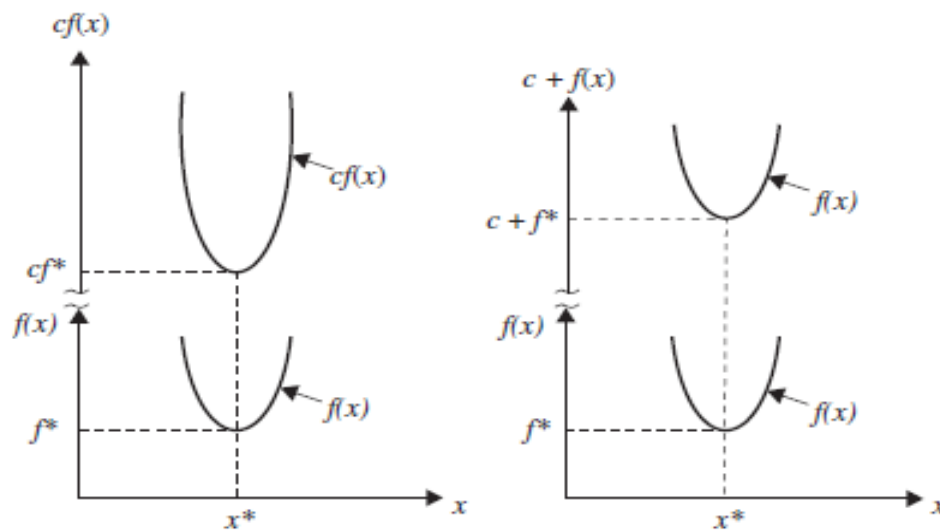


Figure 2-5: Optimum solution of $c f(x)$ or $c + f(x)$ same as that of $f(x)$ (Rao, 2009)

2.6.1 Applications of Optimisation

Optimisation as a term has many dimensions and countless applications in the field of modern engineering. Ravindran et al (2006) stated that the optimisation could be applied to solve varieties of problem in different engineering disciplines as follows:

1. Design of aerospace, vehicle and aircraft structures for minimum weight leading to minimum cost.
2. Optimum design of civil engineering structures, for example, bridges, piles, frames, slabs, towers and dams for minimum cost.
3. In earthquake regions optimisation can applied to achieve the minimum weight design of structures
4. Optimisation can be used in the design of water resources systems, to achieve the maximum benefit.
5. Design of pumps and turbines for maximising the efficiency
6. Selection of a site for an industry
7. Optimum design for electrical networks
8. Optimal production planning, controlling and scheduling

Furthermore, regardless of its engineering applications, optimisation is applied in business and public services to reduce costs and environmental pollution. Consequently, this application can lead to higher profit and success in the competitive fight, which will benefit the customers (Ravindran et al., 2006).

2.6.2 Structure of Optimisation Problems

Arora (2016) defines optimisation as a mathematical process to find the best possible solution amongst several choices for a certain problem. Arora summarised the general scheme of any optimisation problem as:

- 1- Given – Design Parameters
- 2- Find – Design variables
- 3- Minimise – Objective functions
- 4- Satisfy – Design constraints

2.6.2.1 Statement of Structural Optimisation Problem

Rao (2020) expounded that; a nonlinear mathematical or optimisation programming problem for single optimisation problems can be formulated as:

$$\text{Minimise } Z = F\{x\} = F(x_1, \dots, x_n) \quad (2.1)$$

$$\text{Subjected to } g_i\{x\} = g_i(x_1, \dots, x_n) \leq 0 \quad i = 1, 2, 3, \dots, p \quad (2.2)$$

$$h_i\{x\} = h_i(x_1, \dots, x_n) = 0 \quad i = 1, 2, 3, \dots, m \quad (2.3)$$

$$\{x^l\} \leq \{x\} \leq \{x^u\} \quad (2.4)$$

Where:

$\{x^l\} = \{x^{il}\} \quad i = 1, 2, \dots, n$: is the lower limit vector of the design variables $\{x\}$,

$\{x^u\} = \{x^{iu}\} \quad i = 1, 2, \dots, n$: is the upper limit of the design variables $\{x\}$.

Whilst the rest of the symbols of optimisation problem will be comprehensively defined as follows:

Design variables $\{x\}$

In construction, the design variables $\{x\}$ could be either, continuous variables such as the dimensions of the member and the joint coordinates or might be discrete for example, area of standard steel section and number of reinforcing bars (Arora, 2016). Furthermore, based on the nature of the design variables, Rao (2020) classified the optimisation problems into two categories as follows:

- 1- Static optimisation problems: In these problems, values of set of design parameters that make some prescribed function of these parameters minimum subject to certain constraints are estimated. For example, the problem of minimum-weight design of a prismatic beam shown in Figure 2-6 (a) subject to limitation on the maximum deflection can be stated as follows:

$$\text{Find } X = \begin{Bmatrix} b \\ d \end{Bmatrix} \quad \text{which minimises} \quad (2.5)$$

$$f(X) = \rho \times l \times b \times d \quad (2.6)$$

Subjected to the constraints

$$\delta_{tip}(X) = \leq \delta_{max} \quad (2.7)$$

$$b \geq 0$$

$$d \geq 0$$

Where:

ρ : is the density of the member

δ_{tip} : is the tip deflection of the beam

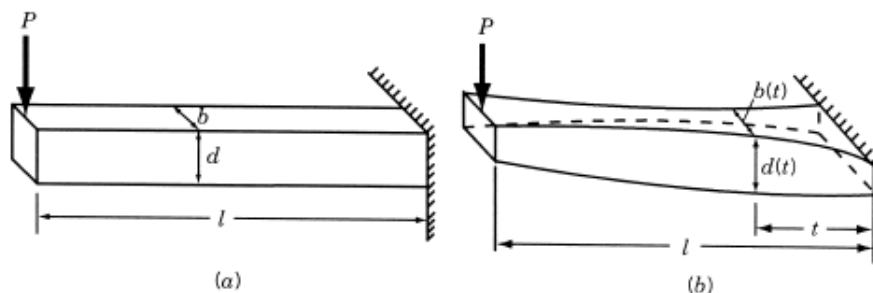


Figure 2-6: Cantilever beam under concentrated load (Rao, 2020)

- 2- Dynamic optimisation problems: These problems are to find a set of design parameters which are all continuous functions of some other parameter that minimise an objective function subject to a set of constraints (Rao, 2020). If the cross-sectional dimensions of the rectangular beam are allowed to vary along its length as shown in Figure 2-6 (b), the optimisation problem can be stated as:

$$\text{Find } X(t) = \begin{Bmatrix} b(t) \\ d(t) \end{Bmatrix} \quad \text{which minimises} \quad (2.8)$$

$$f[X(t)] = \rho \int_0^l b(t) d(t) dt \quad (2.9)$$

Subjected to the constraints

$$\delta_{tip}[X(t)] \leq \delta_{max}, \quad 0 \leq t \leq l \quad (2.10)$$

$$b(t) \geq 0, \quad 0 \leq t \leq l \quad (2.11)$$

$$d(t) \geq 0, \quad 0 \leq t \leq l \quad (2.12)$$

Here the design variables are function of the length parameter (t). This type of problem, where each design variable is a function of one or more parameters, is also known as trajectory optimisation problem.

Objective Function F{x} or Z

The numerical value for the statement of the design problem with which each solution is expressed as problem's objective function. It is a function of the vector of the design variable. It might be the reduction or increase of a quantity, such as cost or profit. For the objective function to be valid, the design variables must directly or indirectly influence it (Arora, 2016). Furthermore, according to Rao (2020), an objective function might have only one goal, known as a single objective function, or numerous goals, known as a multi-objective function. The single objective function is used to optimise a single quantity, such as price or weight. When numerous factors are optimised at the same time, the multi-objective method is used. It may be necessary, for example, to minimise the weight of a building while also minimising the deflection at a specific spot. In this instance, an overall objective function must be created to include each factor's separate objective function (Arora, 2016; Rao, 2020).

Design Constraints $g_i(\mathbf{x})$

The constraints on the other hand are the limitations that the variables have to satisfy before a solution can be accepted. For civil engineering problems, they are usually specified in the code and as such cannot be randomly chosen. The constraints can be to ensure stiffness, stability, strength etc. of the structure (Alkhadashi, 2016; Torregosa and Kanok-Nukulchai, 2002a).

There might be both geometric (side) and functional (behavioural) limitations, as suggested by Rao (2020). Geometric constraints are physical limitations on design variables, such as constructability, whereas behavioural constraints are limitations on the structure's performance, such as a deflection limit. Restrictions of equality and inequality can be used to classify the aforementioned constraints.

Rao (2020) stated that the behaviour restrictions $g_i(\mathbf{x})$ may also be linear or nonlinear. This is mostly determined by the analytical approach used and the assumptions made in order to solve the optimization problem. Additionally, he said that the majority of optimisation problems based on plastic theory have constraints that are generally linear. In contrast, optimisation problems that based on elastic theory often include non-linear restrictions.

The side (explicit) constraints $g_i(\mathbf{x})$ are generally specified constraints applied on the design variables (lower L and upper U values). These restrictions are often imposed by the use of a mathematical formula based on the design requirements, such the minimum thickness of a concrete slab and maximum area of reinforcement steel in any structural element. (Mohammad, 1998).

As shown in Figure 2-7, Rao (2009) illustrates the restrictions surfaces in hypothetical two-dimensional design spaces. The hatched lines serve as a limitations or border for the feasible region, where the black dots are design points. It is shown that some of them lie on the constraint surfaces and they called bound points, where all the constraints attached to these points are called active constraint. Whereas the other design points which lie within constraint surfaces (feasible region) are called free points. Design points can be classified as one of the following categories depending on their location:

- 1- Points lie on feasible region, are called free and acceptable points
- 2- Points lie on infeasible region, are called free and unacceptable points
- 3- Points lie on the hatched lines of feasible region, are called bound and acceptable point

- 4- Points lie on the hatched lines of infeasible region, are called bound and unacceptable point

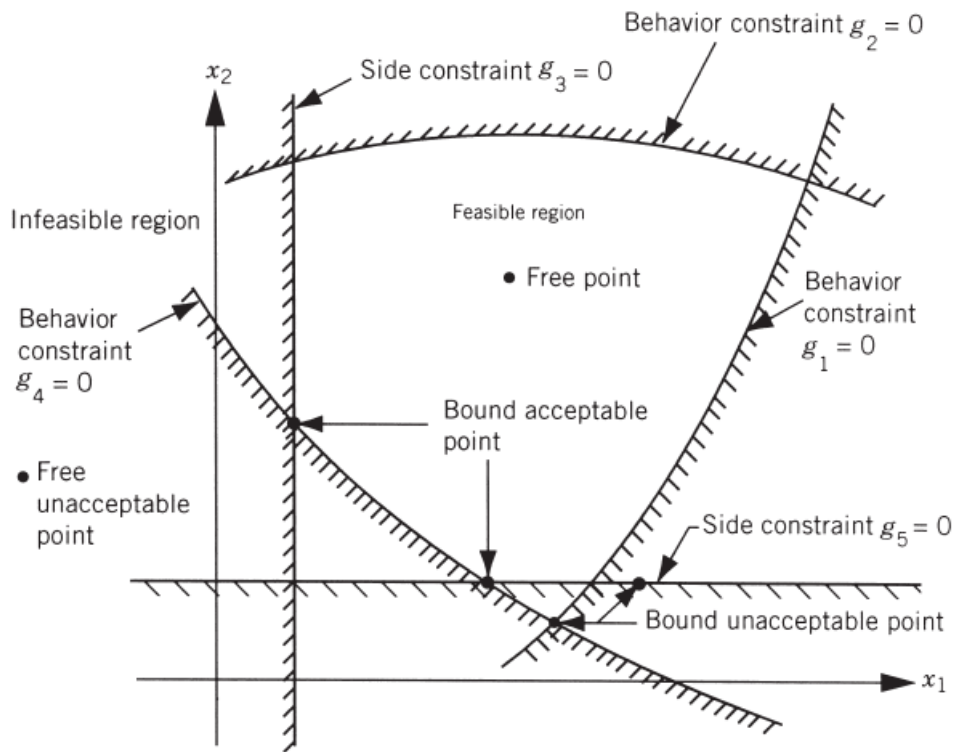


Figure 2-7: Constraints in surfaces hypothetical two-dimensional design spaces (Rao, 2009)

2.6.2.2 Statement of Multi-Objective Optimisation Problem

The material weight does not necessarily correlate to a relevant measure of structural performance in actual applications of sizing optimisation challenges. In fact, in real-life design challenges that must be solved simultaneously, numerous conflicting and often incommensurable requirements are common. As a result, the designer is pushed to find a good balance between competing criteria. Such kind of problems are called multi-objective optimisation problems. The general design optimisation is modified to treat multi-objective optimisation problems as follows:

Minimise

$$f(x) = (f_1(x), f_2(x), \dots, f_k(x)) \quad (2.13)$$

Subject to

$$h_i(x) = 0; \quad i = 1 \text{ to } p \quad (2.14)$$

$$g_j(x) \leq 0; \quad j = 1 \text{ to } m \quad (2.15)$$

Where:

k is the number of objective functions

p is the number of equality constraints

m is the number of inequality constraints

$f(x)$ is a k -dimensional vector of objective functions

Recall that the feasible set S (also called the feasible design space) is defined as a collection of all of the feasible design points:

$$S = \{ x | h_i(x) \leq 0; \quad i = 1 \text{ to } p; \quad \text{and} \quad g_j(x) \leq 0; \quad j = 1 \text{ to } m \} \quad (2.16)$$

2.7 Optimisation Techniques

The inadequacy of available resources to match the incessant growth in human population has called for a radical decision on how to use resources optimally. This makes optimisation of structures and engineering design, in general, a subject of interest to researchers. A development which led to the emergent of several optimisation methods that can be classified into, analytical, numerical, and more recently stochastic methods (Kaveh and Ghazaan, 2018).

The concept of using mathematical tools to find optimum solutions to problems which can either be minimum or maximum values of a function is referred to as Analytical and Numerical optimisation techniques (Kaveh and Ghazaan, 2018). These mathematical methods give the optimum solution as the exact solution to the specified objective function provided all required constraints specified in the optimisation formulation have been satisfied.

Numerical methods categorically employ complex mathematical method (i.e mathematical programming) in their execution and therefore entails expressing the problems using an algorithm. The most prevalent numerical method used in solving optimisation problems is gradient techniques. Others include sequential linear programming, integer programming algorithm, and so on.

These common numerical methods have been in use since the 1970s to solve different kinds of small-scale optimisation problems (Eleftheriadis et al., 2018). Using numerical methods to handle larger-scale optimisation problems is very complex, extremely time-consuming, and might not return a satisfactory solution (Kameshki, 2003; Phan et al., 2017). This is because larger-scale optimisation problems usually consist of non-linear equations that cannot be solved using simple numerical methods (Phan et al., 2017).

Meanwhile, the stochastic methods also known as heuristic search methods are built for complex optimisation problems and thus can effectively return a satisfactory solution for large-scale problems. Stochastic methods are appropriate for discrete design variable problems and large space structures. They are based on random iteration strategies inspired by nature and do not need to compute gradients for the objective function and constraint as required for the numerical methods (Kaveh and Ghazaan, 2018). Genetic Algorithm (GA), Simulated Annealing (SA), Ant Colony Optimisation (ACO), Harmony Search (HSA), and Particle Swarm Algorithm (PSO) are some of the famous stochastic methods. Their key limitation is that they generally required more function evaluations making them very slow and faced with the prospect of getting trapped in the local minima as shown in Figure 2-8.

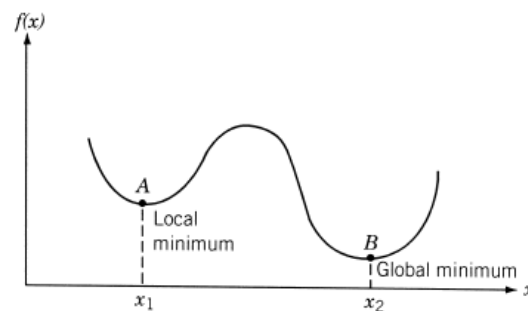


Figure 2-8: Illustration of local and global minima (Rao, 2009)

2.7.1 Genetic Algorithm (GA) Technique

Genetic algorithm is an evolutionary optimisation strategy that is fundamentally derived from Darwin's principle of survival of the fittest (Kameshki, 2003). It belongs to the family of stochastic methods that are mostly used in obtaining the global minimum with a high probability. Genetic algorithms employ the probabilistic transition rule to perform directional search and find the optimum solution from the individual population without the need for the gradients of the objective function. This algorithm is most appropriate in solving discrete and non-differentiable optimisation problems (Phan et al., 2017; Rao, 2009). The application of GA for structural design optimisation has mostly been for single objective functions while that for multi-objective functions is limited.

According to Torregosa and Kanok-Nukulchai (2002) and Kameshki (2003), GA is very efficient and fairly easy to use with the technique only applying chromosomes to conduct finite

point search. However, Khalifa (2011) pointed out that this approach can sometimes lead to a feasible local optimum solution, a situation that has been attributed to the enforcement of too severe penalty function as revealed by Torregosa and Kanok-Nukulchai. Another major drawback of this efficient stochastic optimisation method, GA is its slow speed of execution leading to increased computational time and stopping criterion (i.e.complexity in achieving the desired generation size).

GA operates on binary strings and thus only entails copying and partial swappings of the binary strings (Torregosa and Kanok-Nukulchai, 2002). In the process, an identified set of individuals that can potentially be the solution are regarded as Vectors. The components of these vectors often referred to as genes are formulated using the design variables of the optimisation problem (Jármái, 2000). In order to complete the formulation process, these genes' real values are broken down to form the strings.

Jármái (2000) further clarified that the GA methods make use of three distinct operators which are crossover, mutation, and reproduction operators to replicate the evolution rules. Each of these three operators plays a different role in determining the performance of the GA. For instance, the process of random collection of individuals from the entire population to form a new population is executed by the reproduction operators. The crossover on the other hand deals with the swapping of individual genetic information with another in the chromosomes. The mutation operator facilitates the characters modification of the evolving offspring (Jármái, 2000). All three operators perform continuous iterations on the existing population until the convergence criterion set is satisfied to form a new population. The convergence of the entire process is tested using a fitness value. Rao (2009) briefly summarized those operations of crossover, mutation, reproduction, and evaluation of fitness value must be performed to complete one generation as illustrated in Figure 2-9.

Jármái (2000) and Rao (2009) further explained that the operations of each of these distinct operators are very different. The crossover operator which can be single-point, two-points, or uniform crossover utilizes a particular probability (p_c). The mutation operator (i.e either single-point or bit-wise) uses another probability (p_m) to perform their operations. Meanwhile, the elitist, tournament selection, and roulette wheel are some of the identified reproduction operators.

Previous studies such as (Arora, 2016; Messac, 2015; Rao, 2009), (Torregosa and Kanok-Nukulchai, 2002; Kameshki, 2003; Yassami and Ashtari, 2015) have established that the outcome of the GA optimisation process is greatly influenced by the size of the population and generation for the genetic algorithms. Their studies revealed that high population size will significantly increase the time of computation even though this can sometimes enhance the possibility of obtaining the best optimum solution. Finally, to avoid the risk of abrupt termination of the search during the optimisation process, an adequate generation size will be required.

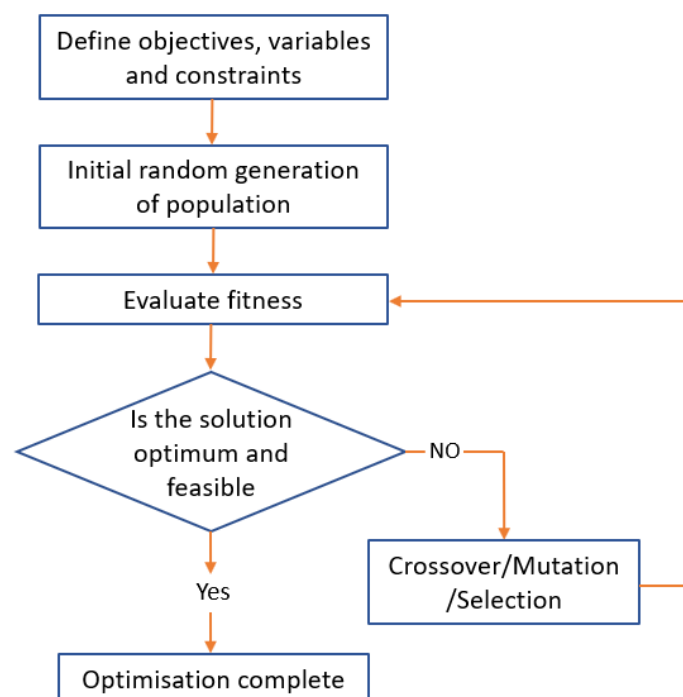


Figure 2-9: Steps of genetic algorithm process (Kameshki, 2003; Torregosa and Kanok-Nukulchai, 2002)

The efficacy of the application of GA optimisation techniques has been widely researched and many studies have affirmed that GA is very superior and produces the best results. This was demonstrated in the comparative study conducted by Ali and Seyan (2014) where a cost optimisation was performed for simply supported doubly reinforced beams using the GA, Generalised Reduced Gradient (GRG), and interior point algorithm. Their results show that the least optimum cost was achieved using the GA, which proved the superiority of GA over other methods.

Another comparative study involving the application of GRG, GA, and Sequential Quadratic Programming (SQP) with different compatible software packages performed by Yenjay (2005)

was reported in (Alkhadashi, 2016). The results also revealed that the best optimum solution was achieved with the GA, although with an increased computation time for a case study problem that is not large (Yenjay, 2005). However, the other two methods considered in the study (i.e GRG and SQP) did not return any results for some of the results as the solution was trapped in local minimum points.

In addition, Torregosa and Kanok-Nukulchai (2002) compared the application of elitist and non-elitist search procedures for the minimum weight of two planar frames to identify which of the two is the best GA approach. According to Torregosa and Kanok-Nukulchai (2002), the elitist search method gives a better outcome when used for an optimisation problem with limited number of generations or many variables. The GA in comparison with the SQP, GRG, and interior point algorithm is very efficient and performed with greater superiority (Ali and Seyan, 2014) and (Yenjay, 2005).

Meanwhile, Hasançebi et al (2009) contend that the GA is less effective compared to some of the other metaheuristic algorithms such as PSO and ACO. Their submission was based on the comparison of seven different metaheuristic algorithms to solve optimisation problems on real size moment steel frame as reported by (Kaveh and Ghazaan, 2018). Hasançebi et al (2009) concluded that the best result of the optimisation problems was achieved using the simulated annealing (SA) and the evolution strategy. This was in agreement with the findings from the earlier study of Khalifa (2011) which concluded that implementation of the survival of the fittest procedure in the GA techniques impact the effectiveness of the method because some good individuals may have been lost in the process. Khalifa (2011) study was on finding the optimum/minimum weight of semi-rigid and rigid steel frames using the GA and the Harmony Search Algorithm (HSA). As deduced from the finding of Khalifa's study, the HAS produced the best results in terms of the minimum weight. The minimum weight from the HAS compared to the GA is 5.18-11.8 % lighter for the rigid and 7.7-11.2% lighter for the semi-rigid frames.

2.7.2 Particle Swarm Optimisation Technique

Kennedy and Eberhart in 1995 first proposed the Particle Swarm Optimisation (PSO) technique based on inspiration from the social behaviour of bugs, birds, and fishes. The technique basically imitates the insect swarm behaviour, fish schooling as shown in Figure 2-10, and the behaviour of birds flock. PSO is grouped into the family of behavioural inspired algorithms

(Dogan, 2010). These primary organisms which the PSO method mimicked used specific social hierarchy in all aspects of their lives such as food and reproduction. Therefore, the application of the PSO technique in finding the optimum solution is analogous to the way insects or swarms find their food. According to Dogan (2010), the representation of the individual organism of either a swarm or flock is referred to as the particle in PSO while the swarm is used to represent the whole population.



Figure 2-10: Fish schooling (Dogan, 2010)

The behaviour of each organism within the swarm cluster is strictly governed by the individual and cluster intelligence which prompt the individual swarm to easily adapt its behaviour to match that of its neighbours. Arora (2016) explained that the intelligence of individual swarm facilitates members to follow any food path established by any member of the cluster irrespective of their distance apart. This concept of imitation is exactly what the PSO method implies. The PSO technique has been proved and well established as a robust and efficient technique for optimizing civil engineering problems. It also has proven records of successful applications to other engineering fields including mechanical, artificial neural network training, multi-objective optimisation, and fuzzy system control. It is very easy to apply because of it requires lesser algorithm parameters (Arora, 2016).

The effectiveness of the PSO techniques in handling multi-objective optimisation problems on rigid steel frames has also been studied by (Soori and Salajegheh, 2019). The constraints imposed were based on AISC-LFRD, the study also considered earthquake effect in the optimisation formulation. The key conclusion from the study is that PSO is very efficient in handling such real-size problems.

In addition, Esfandiary et al (2016) performed a multi-objective function cost and weight optimisation of RC frames by applying a multi-criterion decision-making PSO (DMPS). The optimisation outcomes were obtained in just three runs within a short time for three benchmark problems considered in this study. This also buttressed the submission of Arora (2016) about the simplicity of PSO.

Arora (2016) further expatiated that the PSO has some typical attributes of the evolutionary computation techniques like GA. The PSO process usually begins with a randomly developed individual (i.e particles) which represent a set of solutions. Thereafter followed by updating the generations through the velocity update that directs the movement of the particles around the design space to find the optimum solution to the problem.

During the course of searching for a global optimum, the particles follow up on their individual best position (also known as particle best) and also record the best position that has already been achieved by the swarm (also known as global best). The velocity update mentioned by Arora (2016) enforces the particle movement in the direction of the best-known position i.e. either its own or the global best, in each iteration. This process is continuous with successions of iterations until the global optimum is achieved or the iteration size is exhausted (Arora, 2016; Dogan, 2010).

As demonstrated in Figure 2-11, Rao (2020) presents an overview of the overall process of PSO application for discrete problems. The process begins by specifying some initial parameters such as the size of the population, the maximum iteration number, initial particle, and velocity values. Once the process begins, the value of the velocity and position of the particle will keep updating in each time step. If the solution reaches a convergence point, the returned lowest value of the objective function is chosen as the best/optimum solution. Meanwhile, if the solution did not converge, the next iteration is started. During each iteration, an objective function for each solution will be evaluated and the global optimum is obtained when the values of the minimum objective function are the same for all points.

The main advantages of the PSO are that the technique uses a reduced computational time and also has a better convergence rate (Dogan, 2010). As established from the application of PSO to perform minimum weight optimisation by Dogan (2010), PSO produces better design when using continuous variables as an alternative to discrete variables. On the other hand, PSO has

possibility of encountering disorder in the process convergence due to change of just one parameter and this can considerably affect the outcome of the process.

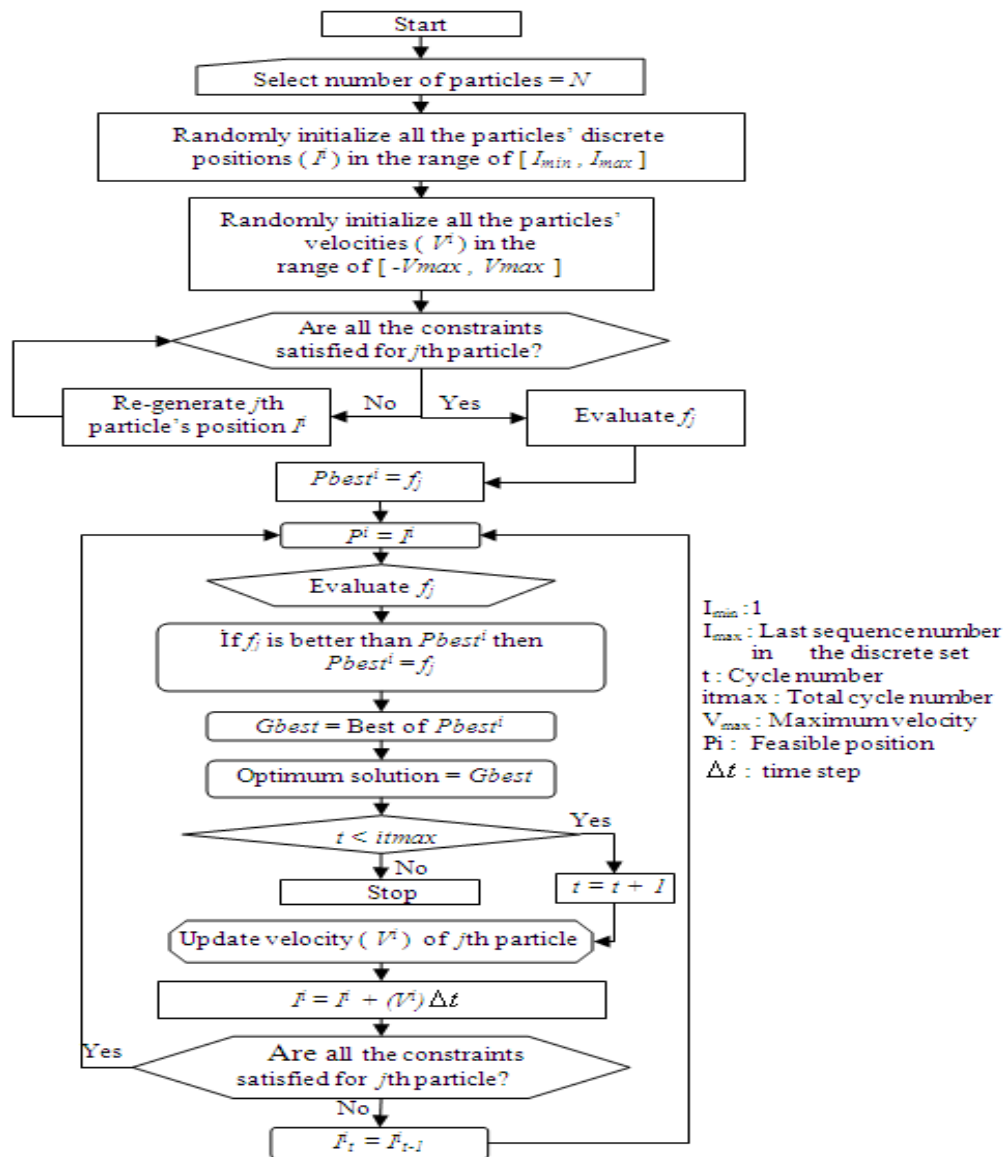


Figure 2-11: Particle swarm process representation for discrete problems (Dogan, 2010)

2.7.3 Harmony Search Optimisation Technique

This is one of the most recent stochastic search techniques and the concept was based on musically pleasing simulation to solve combinatorial optimisation problems. The Improved Harmony Search Algorithm (HSA) finds the optimum solution to design problems by employing a procedure that is similar to the process through which a musician search for pleasing harmony. Similar to the way through which the aesthetic quality depends on the pitch

of the instrument in music, the value of the objective function depends on the values given to each decision variable in the HSA optimisation process. A critical review of the existing works on optimisation techniques revealed that HSA technique has been successfully deployed to solve many practical optimisation problems in engineering fields. However, its application to the optimisation of structural engineering problems is very scanty, an indication that further study is required in this area. One of the earliest studies that applied HSA techniques to structural problems is (Lee and Geem, 2004). Lee and Geem (2004) find the minimum weight design of planar and space truss structures using HSA. Thereafter, Saka (2009) and Degertekin (2007) performed the optimum design of steel frames formulated according to BS5950 and LRFDAISC design codes with HSA. Owing to the success recorded in their earlier studies, Saka and Erdal (2009) and Erdal et al (2008) further deployed the method to find the optimum W-sections for the transverse and longitudinal beams of grillage systems. The above mentioned studies where HSA has been used comprises a relatively small number of design variables and the common observation from all the reviewed studies shows that HSA is very quick and efficient method for solving small-scale optimisation problem.

Hasançebi et al. (2009) carried out a comprehensive evaluation of the performance of the HSA techniques in optimum design of real-size trusses and frames formulated according to ASD-AISC. The key conclusion from Hasançebi et al (2009) and (2010) is that the HSA technique is very unsatisfactory and slow for relative large-scale optimisation problems. Unlike other metaheuristic techniques that are characterized by high speed and efficiency, the HSA performance was described as substandard and characterized with slow rate of convergence and unreliable search efficiency (Hasançebi et al., 2009, 2010). As such an improvement to the application of HSA technique to structural optimisation problems was recommended and this is one of the key drivers that necessitate the current study presents in this thesis.

In this research, Genetic Algorithm (GA), Particle Swarm Optimisation (PSO) and Harmony Search Algorithm (HSA) methods developed in MATLAB are employed to examine their accuracy of the results and computational time required in single and multi-objective problems, for two and three-dimensional steel structures. Moreover, optimisation models have been developed for the evaluation of cost, embodied energy and embodied carbon for steel structures, to aid designers in assessing their energy consumption and carbon impacts due to their designs, during the initial stages.

The basic components of the harmony search algorithm can now be outlined in five steps as follows.

Step 1 Initialization of a Parameter Set: A harmony search optimisation parameter sets are initialized first. These parameters consist of four entities called as a harmony memory size (hms), a harmony memory considering rate ($hmcr$), a pitch adjusting rate (par) and a maximum search number N_{cyc} . It is worthwhile to mention that in the standard harmony search algorithm these parameters are treated as static quantities, suitable values are chosen within their recommended ranges of $hmcr \in (0.70 \sim 0.95)$ and $par \in (0.20 \sim 0.50)$. It should be mentioned that the selection of these values are problem dependent and it requires number of trials to identify the appropriate ones.

Step2 Initialization and Evaluation of Harmony Memory Matrix: A harmony memory matrix H is generated and randomly initialized next. This matrix incorporates (hms) number of feasible solutions. Each solution (harmony vector, I^i) consists of nv integer numbers between 1 to ns selected randomly each of which corresponds sequence number of design variables in the design pool, and is represented in a separate row of the matrix; consequently the size of H is ($hms \times nv$).

$$H = \begin{bmatrix} I_1^1 & I_2^1 & \dots & I_{nv}^1 \\ I_1^2 & I_2^2 & \dots & I_{nv}^2 \\ \dots & \dots & \dots & \dots \\ I_1^{hms} & I_2^{hms} & \dots & I_{nv}^{hms} \end{bmatrix} \begin{matrix} \phi(I^1) \\ \phi(I^2) \\ \dots \\ \phi(I^{hms}) \end{matrix} \quad (2.17)$$

I_i^j is the sequence number of the i^{th} design variable in the j^{th} randomly selected feasible solution. (hms) solutions shown in Equation (3.14) are then analyzed, and their objective function values are calculated. The solutions evaluated are sorted in the harmony memory matrix in the increasing order of objective function values, that is $\phi(I^1) \leq \phi(I^2) \leq \dots \leq \phi(I^{hms})$.

Step 3 Generating a New Harmony: A new harmony solution vector $\hat{I} = [I'_1, I'_1, \dots, I'_{nv}]$ is improvised by selecting each design variable from either harmony memory or the entire discrete set. The probability that a design variable is selected from the harmony memory

is controlled by a parameter called harmony memory considering rate ($hmcr$). To execute this probability, a random number r_i is generated between 0 and 1 for each variable I_i . If r_i is smaller than or equal to $hmcr$, the variable is chosen from harmony memory in which case it is assigned any value from the i -th column of the H, representing the value set of variables in hms solutions of the matrix, equation (2.18). Otherwise (if $r_i > hmcr$), a random value is assigned to the variable from the entire discrete set.

$$I'_i = \begin{cases} I'_i \in \{I'_i, I'_i, \dots, I_i^{hms}\} & \text{if } r_i \leq hmcr \\ I'_i \in \{1, \dots, ns\} & \text{if } r_i > hmcr \end{cases} \quad (2.18)$$

If a design variable attains its value from harmony memory, it is checked whether this value should be pitch-adjusted or not. Pitch adjustment simply means sampling the variable's one of the neighboring values, obtained by adding or subtracting one from its current value. Similar to $hmcr$ parameter, it is operated with a probability known as pitch adjustment rate (par), (Equation. 2.23). If not activated by par ; the value of the variable does not change.

$$I''_i = \begin{cases} I'_i \pm 1 & \text{if } r_i \leq par \\ I'_i & \text{if } r_i > par \end{cases} \quad (2.19)$$

Constraint handling: The new harmony solution vector obtained using above-mentioned rules is checked whether it violates design constraints. If this vector is severely infeasible it is discarded and another harmony vector is sought. However, if it is slightly infeasible, it is included in the harmony matrix. In this way the slightly infeasible harmony vector is used as a base in the pitch adjustment operation to provide a new vector that may be feasible. This is achieved by using large error values initially for the acceptability of the new design vectors. The error value is then gradually reduced during the design cycles until it reaches to its final value. This value is then kept the same until the end of iterations. This adaptive error strategy is found quite effective in handling the design constraints in large design problems.

Step 4 Update of Harmony Matrix: After selecting new values generating the harmony solution vector, its objective function value is calculated. After selecting the new values for each design variable, the objective function value is calculated for the newest harmony vector. If this value is better than the worst harmony vector in the harmony matrix, it is

then included in the matrix while the worst one is taken out of the matrix. The harmony memory matrix is then sorted in descending order by the objective function value.

Step 5 Termination: The steps 3 and 4 are repeated until a pre-assigned maximum number of cycles N_{cyc} is reached. This number is selected large enough such that within this number no further improvement is possible in the objective function.

2.8 Life Cycle Assessment of Steel Structures in Construction Stage.

Life Cycle Assessment (LCA) is one of the widespread tools used in assessing the environmental impact of systems. In this section, the (LCA) of steel structure from a construction standpoint is presented. The needs for the use of LCA in the building sectors and its importance to support decision-making within the built environment were discussed. Thereafter, the applications of LCA within the construction sector are described by critically reviewing some case studies involving the LCA formulation and application to buildings or building materials of steel structures. For instance, a critical review of the complete life-cycle energy usage of two- and three-dimensional steel structures was performed. The review considers the initial estimated embodied energy, embodied carbon, and the total cost involving the cost of material, transportation, production, and erection into account for the complete assessment of all activities (i.e the whole process of construction).

2.8.1 Cost Evaluation

The Steel Construction Institute (SCI) postulated that appropriate use of steel in building and construction can enhance attaining the economic pillar of sustainability in construction (SCI, 2008). Steel compared to other materials used in construction has low operational and investment costs (about 2-4% cost saving), aesthetics, and speedy construction. Nevertheless, the application of optimisation to steel design will ensure that more savings are made.

The total cost of the structure is greatly influenced by the cost of the frame of the buildings. Although the cost contribution from the superstructure or framework cost is just about 10% of the total cost of the building, but it has multiplying effects on other costs (Burgan et al., 2012). For instance, Burgan et al (2012) explained that a 100 mm reduction in the ceiling floor zone of a building results in 2.5% saving in the cost of cladding which subsequently leads to 0.5% saving in the overall cost of the building.

The Institution of Structural Engineer IStructE (2000) submitted that implementing a design decision that adopts uniform size for similar members of slightly variable span is usually cost-effective than having a variety of sizes. This is because the cost of designing, detailing, procuring, fabricating, and erecting typical steel sections is less than having many different sizes of steel that only prioritize having the lowest possible weight of steel. However, adopting uniform steel sizes as design approach often increases the weight of steel and thus increases the embodied carbon emissions of the structure (Moynihan and Allwood, 2014). Another drawback of using this approach is the underutilization of sections capacity as demonstrated in a study that confirms that the average utilization of beams' capacity is less than 50% for about 10,000 steel beams (Moynihan and Allwood, 2014).

However, pertinent study on the optimisation of steel designs has shown that performing cost optimisation may create a balance between cost and weight minimization. This is supported by Pavlovčič et al (2004) findings that reveal that some cost optimisation shows a linear relationship between minimum cost and weight. They compared classical design and volume-based cost optimisation with actual cost optimisation. In the optimisation formulation for the rigid steel frames, several cost contributors such as cost of materials, fabrication, transportation, erection, and finishing were included in the objective function. Pavlovčič et al (2004) applied a metaheuristic method, GA, and mathematical programming method, the Sequential Unconstrained Minimisation Technique (SUMT), to perform cost and volume optimisation of 2 bay two-storey frame and 5 bay four-storey frame.

There is a linear relationship between the results from cost and volume-based optimisation. The result shows that some costs like whole forming and blast cleaning costs were not significant on the results which are the same for all three optimisation methods applied. The significant cost contributors were the cost of materials, fabrication (including welding and/or bolting cost), erection, transportation, and finishing/painting costs, particularly for larger structures. Therefore, it is concluded that weight optimisation can be used for cost optimisation and save adequate cost provided that the objective functions incorporate the major costs at different stages in optimisation formulation.

2.8.1 Cost Model

This review considered five relevant studies, (Bel Hadj Ali et al., 2009; S.W.Jin et al., 2017; Marwaha, 2017; Pavlovcic et al., 2004; Sarma and adeli, 2002) to understand the cost model. Each of the selected studies considered different cost contributors with all of them included the cost of materials, although with different boundaries. The cost of fabrication with slight differences in their respective processes was considered. Two studies which are (Marwaha, 2017; Pavlovcic et al., 2004) included the cost of transportation in their cost formulation. Similarly, the cost of welding on site was included by S.-W. Jin et al (2017) while Bel Hadj Ali et al (2009) and Pavlovčič et al (2004) accounted for the total cost spent throughout the erection stages. The treatment cost to protect fire and corrosion of steel was accounted for by (Marwaha, 2017). Meanwhile, Pavlovčič et al (2004) and Sarma and Adeli (2002) incorporated the cost of general painting in their cost model. In this study, only four costing elements (i.e the cost of materials, transportation, erection, and finishing) were considered. The cost of fabrication was ignored because of its complexity and limited available information. The formulation of the objective formulation as derived from the selected studies are as follows:

2.8.1.1 Material cost

As claimed by (Heinisuo et al., 2016) the cost of materials accounts for about 40% of the total cost of steel structures. Three studies (Marwaha, 2017; S.-W. Jin et al., 2017; Pavlovčič et al., 2004) state the objective function for the procurement cost of materials. The weight of the steel in all three studies greatly influences the material objective function because steel is usually priced by weight. Since there is a consensus of opinions across the three studies, the respective equation adopted were shown from Equations 2.20 –2.22.

- 1) (Marwaha, 2017)

$$C1 = M_c * m * L \quad (2.20)$$

- 2) (S. W. Jin et al., 2017)

$$C_{steel} = \sum_i C_{si} W_i \quad (2.21)$$

- 3) (Pavlovčič et al., 2004)

$$C_{steel-elm} = k_m \rho V_{el} \quad (2.22)$$

Where:

M_c, C_{si}, k_m are the material cost factors

m is the sections' mass

L is the sections' length

W is the sections' overall weight

ρ is the sections' mass density

V_{el} is the elements' volume

2.8.1.2 Transportation cost

Accordingly, the functions by each of them are shown from Equations 2.23 and 2.24.

1) (Marwaha, 2017)

$$C1 = T_c m L \quad (2.23)$$

4) (Pavlovčič et al., 2004)

$$C_{transp} = k_{transp} \rho V_{struct} \quad (2.24)$$

Where:

T_c and k_{transp} are transportation cost factors

V_{struct} is the structural elements' volume

An indication from both equations 2.23 and 2.24 is that the transportation cost also depends on the weight of steel as both equations involve mass and/or density. This is simply related because the carrying trucks transporting steels are classified using their tonnage capacity. Although Pavlovčič et al (2004), recognized that other factors such as the number of times the truck travelled between the factory and the site, the influence of the dimension of the cross-section also affect the total transportation cost. However, the inclusion of such factors in cost formulation can sometimes be difficult. As such, an easy approach of transport cost formulation based on the weight of steel is preferred (Pavlovčič et al., 2004).

2.8.1.3 Erection

Similarly, the cost of erection (see equation 2.25 - 2.27) significantly depends on the steel weight in addition to the labour cost and the cost incurred on the machine power used throughout the erection stages. Pavlovčič et al (2004) included additional factors like man-hour per quantity accounting for several other parameters such as the structure typology, the types of connection used, the condition of erections, and other factors that influence the erection process. These parameters identified by Pavlovčič et al (2004) are key important determining factors on how work will be done at the erection stage. Where the erection cost is priced by hours spent on the job, an increase in time will definitely increase the total cost of erection and this should be properly accounted for in the cost estimation. As an illustration, using multiple shorter span beams will require more erection time than using single long span beams, because the crane will spend more time carrying several pieces than just carrying one long piece. Therefore, the objective function formulation that accounts for this kind of scenario was adopted by Pavlovčič et al (2004).

1) (Bel Hadj Ali et al., 2009)

$$C_{erect} = c_{ere} M_{str} \quad (2.25)$$

2) (S. W. Jin et al., 2017)

$$C_{welding} = \sum_i C_{ws} l_{ws} \quad (2.26)$$

3) (Pavlovčič et al., 2004)

$$C_{erect} = k_{erect} T_{erect} \rho V_{struct} \quad (2.27)$$

Where:

c_{ere} , and k_{erect} are erection cost factors

M_{str} is the structure's mass

C_{ws} is the on-site welding cost factor

l_{ws} is the weld length

T_{erect} is the total erection time

2.8.1.4 Finishing

The finishing cost of steel structures includes the cost incurred on the protection of fire and corrosion estimated as follows.

1- Painting/Corrosion Protection

The (IStructE 2000) requires a provision of protective system for steel structures, particularly where they are exposed to moisture conditions to prevent corrosion. Steel corrosion is an electrochemical process that needs the simultaneous presence of oxygen and water. So, light protection, usually for cosmetic purposes is sufficient when steels are used for dry building shells (IStructE 2000). There are several available corrosion protection systems e.g painting, metallic coating. The key factor to be considered in selecting an appropriate steel protection system is the environmental condition to which they are subjected to (IStructE 2000).

In Equations 2.28 - 2.30, the cost formulation for a typical paint system of steel protection is given. The equations 2.28 - 2.30 show that the cost is a function of the total surface area of the steel. In Pavlovčič et al (2004), additional factors such as the time and cost of manpower were included in the formulation as shown in equation 2.30.

1) (Marwaha, 2017)

$$C_3 = C_c a L \quad (2.28)$$

2) (Sarma and Adeli, 2002)

$$\tilde{P}(\tilde{y}) = \sum_{i=1}^{Nt} \tilde{y}_{pi} l_i \quad (2.29)$$

3) (Pavlovčič et al., 2004)

$$C_{painting} = (k_{paint} T_{paint} + \sum_i (k_{m.paint,i} M_{paint,i})) 2A_{pl} \quad (2.30)$$

Where:

- C_c is the corrosion cost factor
- \tilde{y}_{pi} , a and A_{pl} is the total steel surface area
- l_i is the sections' length of the section

k_{paint}	is the cost of labour and equipment for paint
T_{paint}	is the painting time required for painting
$k_{m.paint}$	is the paint coat's cost,
M_{paint}	is the paint quantity used

2- Fire Protection

According to IStruct (2000), above 650⁰C fire will weaken structural steel members and thus there is need to provide fire protection using insulating materials to sustain their resistance to actions during such intense fire. The institution of IStruct highlighted that the period of fire resistance required and the member section factor (i.e. heated perimeter/cross-sectional area) determine the type and thickness of insulation to be applied.

Steel structural members use as beams, floor slabs and columns have to satisfy a specified fire resistance, typically around 60 to 120 minutes (IStructE 2000). Although, all steel buildings do not require protection because the need for fire protection depends on the other structural elements of the building like composite slab. To arrive at the decision, fire engineering analysis is usually carried out to establish if fire protection will be needed or not. Meanwhile, an efficient strategy for fire engineering can also prevent the need for fire protection. A typical case study is that of the Luxembourg Chamber of Commerce, where the fire assessment has shown that additional fire protection is not required for the building (Burgan et al., 2012).

The total cost of protecting fire will depend on the profile of the steel shape and the type of protections put in place. In addition, the cost of manpower, scaffolding, and materials are included in practice to arrive at the total cost of protection. As previously highlighted Burgan et al (2012), the amount of material for fire protection will depend on the surface area to be protected and the desired thickness of the protection. Studies have shown that typical column requires more coatings of fire resistance than a typical beam. For instance, an approximately 15 mm thick fire protection board will be sufficient to protect beams for around 90mins and only 60mins for columns. Equations 2.31 and 2.32 presents the expression given for fire protection by (Burgan et al., 2012).

$$C_B = (1.43h_{ib}(z) + 0.7)F_c + (0.27h_{ib}(z) + 0.6)mh \quad (2.31)$$

$$C_c = (1.72h_{jc}(z) + 0.53)F_c + (0.11h_{jc}(z) + 0.72)mh \quad (2.32)$$

Where: F_c and mh are the cost factor and man-hour respectively. Also, h_{ib} and h_{jc} are the height of beam and column elements respectively.

All the adopted functions for each stage are all defined in chapter 4, where the coefficients are also defined in the same chapter, table 4-1.

2.8.2 Evaluation of Embodied Energy and Embodied Carbon

Generally, a properly detailed and maintained steel structure can be used indefinitely. One of the great advantages of steel structure is the ability to offer reuse and recycled value. According to SCI (2008) about 10% of the total steel structure can be reused and still maintain their strength and quality while 84% of them can be recycled.

However, when estimating the overall performance of structural steel, The amount of embodied energy and carbon should be taken into account. Eleftheriadis et al (2018) reported that the carbon footprint of steel frames can be between 20-30% for 50-years lifespan. Shadram et al (2016) also asserted that the embodied energy of steel building can be up to 60% of the total energy use of the building.

Currently, there are emphases on energy-efficient buildings, but it has been widely accepted that the embodied energy and carbon will keep going up if not properly checked during the design of buildings. Dixit (2017) stated that energy-efficient buildings often require more materials than conventional ones to achieve a low rate of energy consumption. This in turn will increase the embodied energy within the buildings because of the higher quantity of the materials involved.

According to Dixit (2017), embodied energy (EE) is defined as the total amount of the primary energy utilized in constructing a building through the use of construction materials, products, and processes, along with related transportation, administration and services. It is also referred to as the sum of energy involved in manufacturing a product assuming that this energy is embedded within the product (Haynes, 2010). In structural engineering, the product is the building or structure, and the manufacturing process involves the manufacturing of construction materials involved, the transportation from the factory to sites where they are needed and the coupling together of the materials to make the final products (i.e the building).

To have a complete account of the embodied energy, the energy needed during demolition, recycling and the final disposal of the waste is also considered. EE belongs to the class of non-renewable energy that cannot be regained after it has been used, so newer sources are always needed for any subsequent uses.

Meanwhile, Embodied Carbon (EC) is the resultant lifecycle greenhouse gas emissions that occur during all processes involved (i.e from the manufacturing of raw materials to the demolition of the final products). EC is often used interchangeably with carbon footprint and is normally expressed as an equivalent of carbon dioxide CO₂ (Chae and Kim, 2016). In addition, the total energy used and the corresponding emission from the operation stage of the building are referred to as Operational Energy (OE) and Operational Carbon (OC) respectively. OE and OC account for the energy consumed and emissions resulting from lighting, heating, cooling, and ventilation of the buildings.

UKGBC (2018) argued that lowering the EE and EC can be more measurable and satisfying than lowering the OC and OE. The argument was because energy saving on OE and OC spread over a long period of time whereas the EE and EC savings are immediate and can be quantified instantly. The UK Green Building Council UKGBC (2018) reiterated that the positive impacts of savings made from EE and EC on the environment are more valuable than that of OE and OC because the long-term savings from OC might be small in comparison with what is needed at that time. Most importantly, the behaviour of the end-users and their awareness of sustainability determine the amount of OC savings attainable which is an indication that the estimated savings might not actually happen during usage. As such, UKGBC (2014) affirmed that reducing the EE and EC will significantly reduce the environmental emissions which will improve environmental sustainability.

The LCA of a building as illustrated in Figure 2-12 demonstrates the framework for comparing different services, materials, and components for a building structure (Zhou and Azar, 2018). The process entails different stages often referred to as system boundaries for estimating the EC and EE of a particular project. Usually, the Initial Embodied Energy (IEE), Recurring Embodied Energy (REE) and Demolition Energy (DE) are the boundaries recognized.

The IEE comprises the EE of materials used for the building, energy incurred during transportation and construction. Meanwhile, REE is the amount of energy used during the operation of the building more than the main OE. REE includes the energy consumed because

of repair, renovation, replacement and the general maintenance of the building. The final boundary is the DE which is described as the amount of energy used in the demolition, reuse and recycling or disposal (Dixit, 2017).

The critical review of pertinent studies shows that efforts are been concentrated on estimating the embodied energy of material extraction and processing because of its high impacts but energy from other stages in the life cycle are being neglected. Vukotic et al (2010) employed LCA technique to investigate the impact of different lifecycle stages on the amount of the embodied carbon of the structural elements. They revealed that the major contributors to EE were material sourcing and waste handling after demolition. On the other hand, the EE from the transportation, labour, construction, and demolition stages were minor. Nevertheless, the contribution from all the stages should be considered for the appropriate estimation of EE and EC.

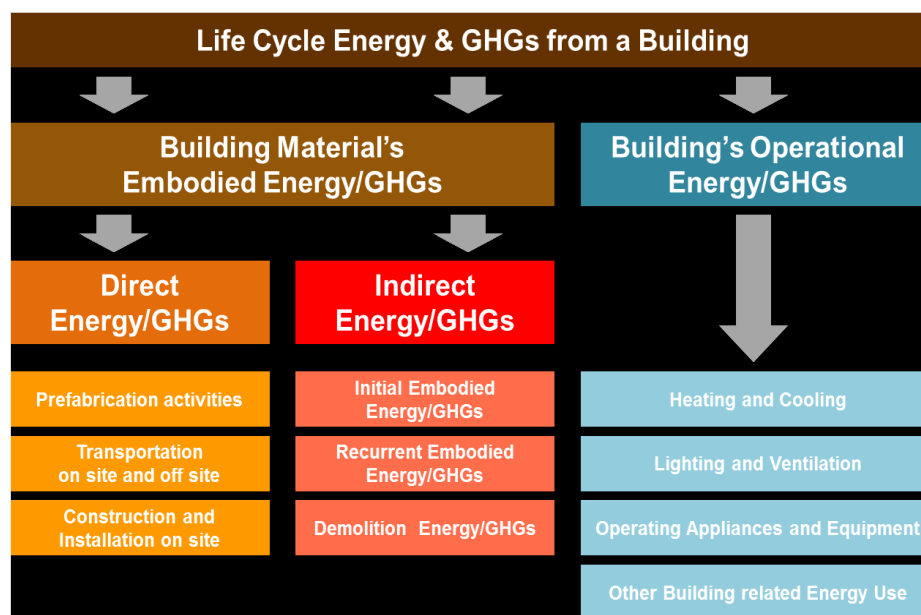


Figure 2-12: Life cycle energy and emissions of a building (UKGBC, 2014)

2.8.3 Embodied energy and carbon models

The optimum embodied energy has been attributed to the minimum weight of steel by many researchers but still, some other studies contend this claim as established by Foraboschi Foraboschi et al (2014) and also cited by Eleftheriadis et al (2018). Foraboschi et al (2014) reported that the embodied energy of steel does not relate to the least weight when optimizing various tall building structural floors.

The IEE system boundary is considered in this study and the EE and EC functions were obtained from the previous studies. The function given by each of them for each phase is given as follows with a justification for choosing it.

2.8.3.1 Material

The contribution of materials used in constructing building elements to the total EE is widely accepted as the highest. This is substantiated by Zhou and Azar (2018) findings that 75% of the EE of the building is from the materials used in the building. Specifically, steel is known for higher embodied energy compared to other materials used in buildings. So, it is highly required that the embodied energy of the materials should be included in the estimation of EE and EC.

Equations 2.33 - 2.36 presents the calculation of embodied energy of the material used in building as found in literatures.

1) (Zhou and Azar, 2018)

$$EE_m = \sum_i m_i EE_i \quad (2.33)$$

2) (Brütting et al., 2018)

$$E = I_p M \quad (2.34)$$

3) (Brütting et al., 2020)

$$E = EC^p \times M \quad (2.35)$$

4) (Mao et al., 2013)

$$E_1 = M \times f_i^b (1 + \varepsilon_i) \quad (2.36)$$

Where:

E, E_1 and EE_m is the total embodied energy

EE_i, I_p, EC^p and f_i^b is the factor for the embodied energy

ε_i is the waste factor

All the equations presented are a function of the mass of the structure and the EE factor because the mass of materials is widely used in their quantification. The amount of energy used in the extraction of raw materials, the type of energy used during production, transportation between the factory and the site are accounted for in the embodied energy factor. The factors vary for different locations and the adopted database.

2.8.3.2 Transportation

As pointed out earlier, the embodied energy from the transportation of materials and other accessories associated with the building construction is vital in assessing the life cycle of a building. Dixit (2017) found that excluding the transportation effect from the LCA will reduce the total embodied energy by 5-7%. The major determinant from the transportation energy is the fuel used by the conveying means and the amount of emission the system is generating. So, an accurate estimation will account for all energy from the movement of materials, equipment, labor, wastes and others (Shadram et al., 2016).

An approximate estimation of the transportation energy is usually carried out by calculating the travel distance using map (Nässén et al., 2007). However, Dixit (2017) submitted that this approach does not accurately account for all the EE from the transportation. Dixit argued that several other factors such as the type and size of the vehicle in use, the weight of the materials or equipment been transported determine the amount of the embodied energy involved in the transportation stage. Meanwhile, finding all these factors is sometimes challenging and as such the EE is predicted using the amount of materials and the travel distance (Zhou and Azar, 2018). The expressions found in existing literature are given from Equations 2.37 – 2.40.

$$E_t = 1.66 \sum_i m_i E E T_i D_i + n E_{LP} \quad (2.37)$$

$$E_T = (d_{T,N} + d_{T,F}) I_T M_{new} \quad (2.38)$$

$$E_T = E C^T (d_p + d_s) \quad (2.39)$$

$$E_2 = \sum \frac{M_j L f_k}{1000} \quad (2.40)$$

Where:

E_t, E_2, E_T is the transportation energy

EET_i, EC^T, I_T, f_k	is the consumption factor for transportation
E_{LP}	is the consumption factor during loading and unloading.
$d_{T,N}, D_i, L, \text{ and } d_p$	is the total distance traveled from factory to fabrication site
$d_{T,F}, d_s$	is the total distance traveled from fabrication site to construction site.
n	is the number of distribution centres

Even though there are variations in all the equations 2.37 – 2.40 but all of them depend on the mass of the material's structure and the distance traveled. The distance traveled is very important because the energy consumption and emission will be a function of how far the vehicles as traveled.

The impact of having different distribution centers was considered in the formulation by (Zhou and Azar, 2018). This was because energy is expended during the loading and unloading of steel from one distribution center to the other. Although, this is difficult to account for because the number of centers the steel has passed through might be difficult to ascertain. The easily measured distance is from the steel shop to the fabrication shop or directly to the construction site. Zhou and Azar (2018) also involved energy consumption for the return trip of the vehicle which was taken to be around 66% of the energy expended for the full trip. The inclusion of return trips was also suggested by Hammond and Jones (2011) for proper estimation although they mentioned that not all projects would require the return trip.

Meanwhile, the return trip consideration was not included in the approach by Brütting et al (2018) and Brütting et al (2020) but their calculation considered the distance from the steel industry to fabrication and to the usage site while multiplying with the corresponding transportation factor.

Therefore, the formulation of the objective function presented in this study included only the distance from shop to site and the return trip. Since the energy incurred on the transportation depends on the steel weight, the additional steel material that would be counted as waste during the transportation is also added to give a full measure of the weight.

2.8.3.3 Erection stage

In structural steel construction, the erection stage encompasses the assemblage of the components, sub-assemblies and products to construct the finished building. Dixit (2017) claimed that if the energy incurred at the erection stage is excluded from the estimation, the total EE will be reduced by about 10%. The energy used at the erection stage depends on the type of equipment used during the process. Different types of equipment such as hoisting, welding machine, crane, earthmoving machinery, and others require different forms of energy like electricity, steam, fuel, etc. for their operations. So, the amount of energy used, and their emission will depend on the type of energy used. Even though, steel construction process is quite simple and does not require extensive use of equipment, a proper account of any energy incurred at the erection stage should be accounted for as shown in equation 2.41 – 2.43.

1) (Brütting et al., 2018)

$$E_t = \sum_i \epsilon_i = \sum_i T_i ECF_i \quad (2.41)$$

2) (Brütting et al., 2018 and 2020)

$$E = M EC^A \quad (2.42)$$

3) (Mao et al., 2013)

$$E = \sum \sum \frac{R_r \times f_n}{1000} \quad (2.43)$$

Where:

T is the amount of time for lifting and installation process

ECF is the energy consumption factor

EC^A is the energy consumption factor for hoisting crane

f_n is the emission factor

R_r is the total amount of resources or energy usage

Brütting et al (2020) accounted for just the mass and the hoisting crane energy because the study conducted a comparison with a similar structure and as such, the common processes in the two cases were not considered. Both Brütting et al (2020) and the baseline study used only

the mass because of the assumption that the mass of the building is the key determinant to the total energy used by the crane. Therefore, the energy form differences for various types of equipment is ignored.

On the other hand, Mao et al (2013) considered all the major equipment required for the erection of structure and their individual energy type used during their operation. Thereafter, a factor of energy usage was assigned to each equipment in relation to the three main forms of energy considered (i.e fuel, electricity, and water). Although, Mao et al (2013) explanation is not related to the construction of steel structures.

Finally, an estimation of the embodied energy during the erection process that accounts for the equipment type and the total time spent on the erection process was conducted by (Zhou and Azar, 2018). The study used the amount of fuel expended by the equipment (ECF_i) to calculate the energy consumption and emission of the equipment. Zhou and Azar (2018) approach is similar to the model of off-road and non-road prescribed by California Air Resources Board (CARB) and Environmental Protection Agency (EPA) US respectively for fuel consumption of equipment (Akbarnezhad and Xiao, 2017). From the critical review of all methods, the approached adopted by Zhou and Azar seems very practicable and thus was adopted in this study. Meanwhile, the estimation of erection time was carried out using a complicated program in BIM. So, to avoid the complexity due to the limited time and resources, this study adopted the time as a function of the steel weight.

2.9 MATLAB Software

The structural engineers aim to have a satisfactory optimum design that can achieve any optimisation goals. However, most of the existing methods to implement these optimisation techniques are very complex making it difficult to achieve optimum design (Marwaha, 2017). But the increasing development of powerful computation tools and the ease of their accessibility has advanced the application of optimisation techniques to solve real size problems (Torregosa and Kanok-Nukulchai, 2002). One of such simple and effective optimisation packages is MATLAB.

MATLAB is a global optimisation toolbox with various optimisation algorithms (e.g GA, PSO, SA etc) embedded within the program (Messac, 2015). This Toolbox comprises a library of programs or m-files, that is fit for solving least squares curve fitting, minimization, and other

problems (Rao, 2009). They can also be used to solve both constrained and unconstrained optimisation problems (Arora, 2016).

Cazacu and Grama (2014) effectively utilized MATLAB to conduct optimisation of steel truss and compared the results of optimisation using the built-in optimisation command and self-developed algorithm. The GA optimisation method was used in addition to the FEA incorporated into the MATLAB for analysis of truss. Cazacu and Grama (2014) concluded that the result obtained from the built-in optimisation command was better although it was indicated that the Lagrangian barrier method for handling constraints in MATLAB is less superior.

2.10 Review of Related Past Works

This section presents a critical review of previous studies pertinent to the optimisation of cost, weight, and embodied energy of steel frame structures. The review is conducted to identify the gap in existing studies and thus form the basis for proposing this research work.

Mela and Heinisuo (2014) used PSO algorithm to perform the optimum design of high strength steel (HSS) beams, columns, and trusses with two different objective functions which are optimum weight and cost. The study adopted a hybrid approach by creating steel sections with different steel grades for the webs and flanges. The results of the optimisation achieved 17-24% cost savings in HSS columns, 10-21% cost saving in large truss, and the least savings from the HSS beams. The least savings from the beam was because of the imposition of displacement constraints in the optimisation formulation for the HSS beams. Mela and Heinisuo (2014) highlighted that PSO algorithms employed in their study is very efficient in optimizing structural elements but require higher computational time for relatively complex problems with different design variables and two objective functions. Their study concluded that HSS which are steels with yield strength above 355 N/mm^2 is a very good option for cost reduction particularly when the hybrid sections are applied.

Similarly, an automated approach was employed by Eleftheriadis et al (2018) to perform cost and embodied energy optimisation models that can support early decision making during structural steel design. They developed a parametric model in C++ using Monte Carlo methodology to perform design and optimisation for the best floor grid arrangement. Optimum cost and embodied carbon of materials, design, fabrication, erection, and fire protection are used as the objective functions. The results obtained from the automated optimisation approach

were compared against the results obtained from the manual optimisation. The result of the optimisation shows that there is a proportional relationship between cost and embodied energy. As expected, the speed of execution of the model is higher than that of the normal optimisation method.

Furthermore, Brütting et al (2018) presented a novel optimisation formulation to optimize the weight, embodied energy, GHG emissions of structural truss made of reused stock elements. The estimation of the EE and GHG accounted for the additional energy required for the systematic deconstruction of reused elements. They used a sequential approach of a mixed-integer linear program and their results showed that the deconstruction energy of reused elements significantly impacts their EE and GHG emissions. The optimisation of the reused elements results in 29% reduction in energy compared to the use of new steel materials.

Brütting et al (2020) extends the earlier study of Brütting et al (2018) by employing the optimisation model to steel frames. Brütting et al (2020) only account for the GHG in the objective function formulation and considered the generation of multiple members from stock elements when optimally cut. The life cycle processes added were erection and demolition in which processes common to the two elements were neglected. The optimisation method used is the mixed-integer linear programming (MILP) and the results obtained show that the reused elements have a lower element capacity utilization and GHG emissions that are 35% lesser than using new elements. This agrees with the earlier outcomes from Brütting et al (2018). Although, Brütting et al (2020) contends that the emissions from the reused steel elements may increase if the full LCA boundary of the system is considered in the optimisation formulation.

Mao et al (2013) studied the single objective function of material weight and lifecycle cost and multi-objective function of both to optimize structures prone to earthquake damage. The optimisation formulation considered damage due to earthquake intensities using EC3 specifications to optimize a rigid ten-storey steel frame. Single objective optimisation of initial weight revealed that the structure will be more prone to future earthquake damages. Therefore, the total life cycle cost of the structure will increase because of the additional cost that will be incurred on repair and renovation of the damages. Meanwhile, the result from multi-objective function gives balance between the two functions. Thus, multi-objective function optimisation can lead to structures with lower life cycle costs and also with required earthquake resilience.

Furthermore, Barraza et al (2017) compared the performance of GA and PSO in optimizing structural steel frames. Multi-objective optimisation of the performance of structural steel frames subjected to earthquake was examined based on the AISC specifications. Both the GA and PSO methods give good structural performance for the frames, but PSO performed better. Plevris et al (2011) applied the specification from EC3 with design constraints for shear, bending, and deflection to performed optimisation of the steel frame. They submitted that the optimisation results obtained from both GA and PSO methods are the same, but the PSO showed a faster rate of convergence. Even though, some of the PSO solutions were trapped in the local minima.

The outcome of the review presented in this study revealed that there are gaps for further areas of research to be explored. Firstly, limited researches were found on the optimisation of steel frames according to Eurocodes with the majority of the reviewed studies being according to AISC-LFRD. The design method varies from different codes therefore, the resulting optimisation based on different codes will be varied. Therefore, this study will perform optimisation of steel frame according to Euro code. This study will contribute to the benchmark problems for steel frame optimisation according to the Euro code.

The review also observed that many studies performed single optimisation of either cost or embodied energy of steel frames but there is limited evidence of multi-objective optimisation for steel frames. Therefore, this study will perform both single optimisation using (weight) and also the combination of both of them with the calculation of embodied carbon and cost.

In terms of the best optimisation techniques, the review noted that GA, PSO, and HSA are widely accepted because of their capability of handling non-linear programming problems. Plevris et al (2011) and Messac (2015) submitted that the use of GA, PSO and HSA will enhance the possibility and speed of achieving global optimum through a random selection of variables, which are usually near the optimum. Here, further study to investigate the best optimisation approach between GA, PSO and HSA for performing single and multi-objective optimisation problems is presented. In addition to the validation of earlier claims, this study will also remove the hassle of using complex programming methods, like MILP.

Finally, the review finds that MATLAB is emerging to be another everyday use computational tool like Excel. MATLAB is very advantageous and superior to Excel tools because it is faster and prevents any chance for alteration that can disrupt the entire program, which is one of the

drawbacks of Excel. However, MATLAB is not easy to use as Excel because of the Excel popularity among designers. Therefore, the study will implement a built-in function in MATLAB with the integration of the analysis and design of the frame. The implementation of the built-in function will reduce the complexity of MATLAB, reduce the computation time for the optimisation and thus ensure the easy implementation of cost and embodied optimisation in practice.

2.11 Summary

In this chapter, a general overview of steel materials and frames was introduced. The reason why steel-framed structures should undergo structural optimisation was highlighted. Differences between conventional design and design optimisation of steel structures were presented. Then, the literature survey was divided into two main categories: optimisation techniques and life cycle assessment. To choose an efficient optimisation approach and build up model formulas for all the life cycle aspects during the construction phase. Different optimisation approaches were discussed, and the life cycle assessment from a construction standpoint was widely reviewed. According to the survey, GA, PSO, and HSA performed well in structural problems, though their performance varied from study to study. However, there isn't enough information about their application in multi-objective function problems. Moreover, the literature on LCA led this study to shape the adopted formulas and coefficients that are presented in chapter 4.

CHAPTER THREE. ANALYSIS AND DESIGN PROCEDURE

3.1 Introduction

This chapter describes the structural analysis method and design code of practice used in this study, and their integration into MATLAB. A structure's analysis is a process for determining the balance between external actions on the structure as a whole and the internal response of the structure's elements. As stated by Issa and Mohammad (2008), that the most suitable structural analysis methods for programming could either be the stiffness method or the flexibility method. Therefore, this research has adopted the Direct Stiffness method for performing the structural analysis of the two-dimensional and three-dimensional structures. The second part of this chapter considers the design procedure adopted in this research.

3.2 Analysis

The direct stiffness method is the most common implantation of the finite element methods. It covers the application of the method in the analysis of linear elastic structures subjected to static loads. This method is applicable to 1D, 2D, and 3D structures. Most finite element analysis programs adopt this method due to its ease of programming on a computer as well as its suitability for the computer analysis of large structures. In current practice, a self-coded program with the Direct Stiffness Method can do all the analysis of rigid 2D and 3D structures. For the analysis of the steel structures, MATLAB code was developed, including an integration of the direct stiffness method. The required analysis input was established in an excel file, which was then called out by the MATLAB code. Examples of the inputs that can be edited are the nodal load, point load, distributed load, variable load, fixity of members, and so on. Table 3.1 below gives the input required in Excel.

Table 3.1: Input for the excel file

Material properties	Steel grade, Poisson ratio, steel density
Joints	Joint coordinates X and Y for two dimensional frames with the joint numbering
Members connectivity	Starting and ending nodes of members, element number and structural element type (i.e., column or beam)
Base fixity	Node number and fixity type
Nodal loads	Magnitude of load specified with the node designation
Point loads	Designation of the element loaded, magnitude and direction of load
Distributed loads	Designation of the element loaded, magnitude and direction for total and variable loads on each element.

3.2.1 Analysis of 2D Structures

The following are the procedures for using the direct stiffness method that was employed in the analysis.

- 1) Creating an analytical model to easily identify the joints and members as well as both ends of each member of the structure. The elements were then named separately. An illustration of that is shown in Figure 3-1 (a).
- 2) After that, local and global axes were used to define the direction of the forces and displacements of the entire structure and individual members. Axial force, shear force, and bending moments occur in individual members, whereas loadings and joint displacements occur in the global axis. Both axes are shown in Figure 3-1 (b), with the local axis represented by x-y and the global axis denoted by the traditional cartesian coordinate system (X-Y).
- 3) To transfer the force, displacement, and stiffness matrix from one coordinate to another, a transformation matrix $[T]$ and its transpose $[T^T]$ were necessary due to the two-coordinate system of the plane frames. The transformation from local to global displacements is shown in equation 3.1, where the used angles of the members (α) is 0° and 90° for horizontal and vertical members respectively as shown in Figure 3-1 (c). Thus, the transpose of the transformation matrix for each member becomes as given in equation 3.2.

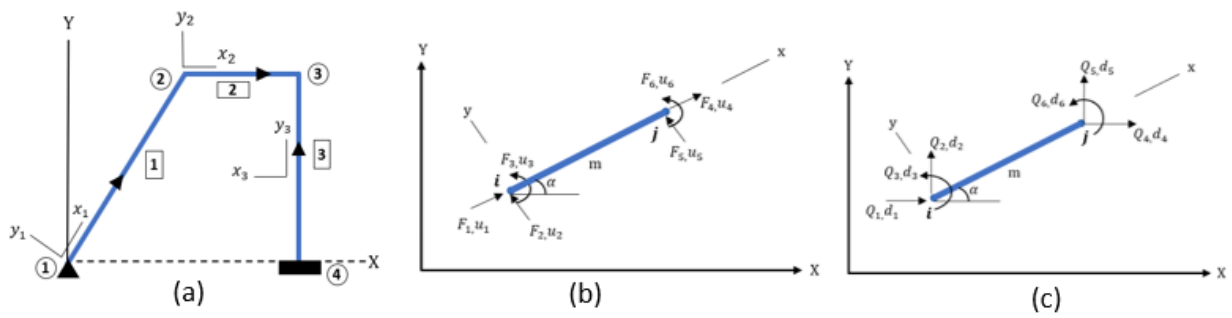


Figure 3-1: (a) Representation of nodes and numbering of members (b) Displacements and member end forces in local coordinates (c) Displacements and member end forces in global coordinates.

$$[T] = \begin{bmatrix} C & S & 0 & 0 & 0 & 0 \\ -S & C & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & C & S & 0 \\ 0 & 0 & 0 & -S & C & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \quad (3.1)$$

Where, C and S are $\cos\alpha$ and $\sin\alpha$ respectively.

$$[T^T] = \begin{bmatrix} C & -S & 0 & 0 & 0 & 0 \\ S & C & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & C & -S & 0 \\ 0 & 0 & 0 & S & C & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \quad (3.2)$$

As a result, the displacement vectors in both axes have the following relationship:

$$\{d\} = [T]\{D\} \quad (3.3)$$

Where d and D are the local and global displacement vectors, respectively.

That of force relation is:

$$\{F\} = [T^T]\{f\} \quad (3.4)$$

Where f and F refer to the force vectors in the local and global coordinates respectively.

The conversion of the stiffness matrix of the member $[k_l]$ to global stiffness matrix $[K]$ is:

The member's stiffness matrix $[k_l]$ is converted to the global stiffness matrix $[K]$ using the following formula:

$$[K] = [T^T][k_l][T] \quad (3.5)$$

Where k_l is given as follow:

$$[K_l] = \begin{bmatrix} \frac{EA}{L} & 0 & 0 & -\frac{EA}{L} & 0 & 0 \\ 0 & \frac{12EI}{L^3} & \frac{6EI}{L^2} & 0 & -\frac{12EI}{L^3} & \frac{6EI}{L^2} \\ 0 & \frac{6EI}{L^2} & \frac{4EI}{L} & 0 & -\frac{6EI}{L^2} & \frac{2EI}{L} \\ -\frac{EA}{L} & 0 & 0 & \frac{EA}{L} & 0 & \frac{6EI}{L^2} \\ 0 & -\frac{12EI}{L^3} & -\frac{6EI}{L^2} & 0 & \frac{12EI}{L^3} & 0 \\ 0 & \frac{6EI}{L^2} & \frac{2EI}{L} & 0 & -\frac{6EI}{L^2} & \frac{4EI}{L} \end{bmatrix} \quad (3.6)$$

Where:

E :is the young's modulus of elasticity

A :is the member's cross-sectional area

L :is the length of the elements

I :is the moment of inertial

- 4) Afterwards the number of unknown displacements was determined by identifying the frame's degrees of freedom. Each end of the members of the plane frame has three degree of freedom, where only the displacements at the unrestrained ends needed to be determined since the restrained ends had zero displacements.
- 5) The global stiffness matrix and the degrees of freedom of each member identified in the previous steps were then utilised to assemble the overall stiffness matrix of the frame, $[K_s]$.

The above steps were inputted as a form of equations and matrixes in the respective section of the MATLAB code. The code would then calculate what is required by calling out the properties of the section and some other values that have already been defined. A summery of the self-written program in MATLAB was merged and presented in Appendix-C. However, A small section from the program code is presented in Figure 3-2 showing the equations for local stiffness matrix $[k_l]$, transformation matrix $[T]$, transpose of the transformation matrix $[T^T]$, global stiffness matrix $[K]$, and the total stiffness matrix $[K_s]$. The program would then compute

the next steps based on the calculations of these parameters. The program is attached in Appendix c.

```

%Finding Stiffness Matrix of the Structure
for i=1:numberElements

    k1(:,:,i)=[EA(i)/l1(i) 0 0 -EA(i)/l1(i) 0 0; 0 (QA*12*EI(i))/(l1(i)^3) (ZA*-6*EI(i))/l1(i)/l1(i)^2 0 (QA*-12*EI(i))/(l1(i)^3) (ZA*-6*EI(i))/l1(i)^2;...
    0 (ZA*-6*EI(i))/l1(i)^2 (VA*4*EI(i))/l1(i) 0 (ZA*6*EI(i))/l1(i)^2 (UA*2*EI(i))/l1(i); (-EA(i))/l1(i) 0 0 (EA(i))/l1(i) 0 0;...
    0 (QA*-12*EI(i))/(l1(i)^3) (ZA*6*EI(i))/l1(i)^2 0 (QA*12*EI(i))/(l1(i)^3) (ZA*6*EI(i))/l1(i)^2;...
    0 (ZA*-6*EI(i))/l1(i)^2 (UA*2*EI(i))/l1(i) 0 (ZA*6*EI(i))/l1(i)^2 (NA*4*EI(i))/l1(i)]; % Local Stiffness Matrix

    T(:,:,i)= [cosa(i) sena(i) 0 0 0 0; -sena(i) cosa(i) 0 0 0 0; 0 0 1 0 0 0; 0 0 0 0
    cosa(i) sena(i) 0; 0 0 0 -sena(i) cosa(i) 0; 0 0 0 0 0 1]; % Transformation Matrix

    Tt(:,:,i)=T(:,:,i)'; % Transpose of Transformation Matrix

    KgElement(:,:,i)=Tt(:,:,i)*k1(:,:,i)*T(:,:,i); % Global Matrix of Each Element

    stiffness(elementDof(i,:),elementDof(i,:))= stiffness(elementDof(i,:),elementDof(i,:))+Tt(:,:,i)*k1(:,:,i)*T(:,:,i); % Total Stiffness Matrix
end

```

Figure 3-2: Part of Stiffness Matrix assembly procedure in MATLAB R2020a

- 6) The next step is identifying the fixed end reaction (FER) of the members in global axis, due to the nodal or intermediate loads applied on them.
- 7) The global displacement of the members will then be calculated using the global force vector and the assembled stiffness matrix as shown in equation 3.7.

$$\{F\} = [K_s]\{D\} \quad (3.7)$$

- 8) After that, the internal forces of all the members will be computing using equation 3.8

$$\{F\} = [K_s]\{D\} + \{F_{ER}\} \quad (3.8)$$

- 9) The final step of the plane frame analysis is converting the global internal forces obtained to their local counterparts.

Consequently, the outcome of all the above steps will be the internal forces of all the members of the frame which are shear force (V), axial force (N), and bending moment (M) of the members which will then be used in the design stage of the frame.

3.2.2 Analysis of three dimensional (3D) Structures

The analysis of 3D structures is quite similar to the analysis of 2D frames. Each node of any 3D frame element has three displacements and three rotations with respect to the three global axes. Overall, there are six degrees of freedom at each node of 3D frame element; three displacements in the x, y and z axes, and three rotations with respect to these axes. Therefore, any three-dimensional frame element with two nodes has altogether twelve degrees of freedom, as shown in Figure 3-3 below.

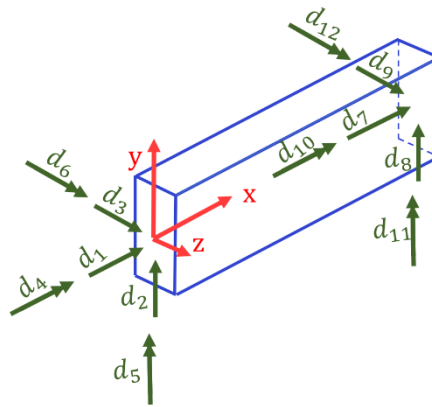


Figure 3-3: Segment of space frame element showing forces and displacements at the nodal coordinates

The same steps and equations of the two-dimensional frames are applicable in the analysis of three dimensional. However, as stated before that the three-dimensional structures have twelve degrees of freedom, therefore different local stiffness matrix $[k_l]$ and transformation matrix $[T]$ will be applied. Equation 3.9 shows the stiffness matrix of a beam segment of a space frame.

$$K_l = \begin{bmatrix} \frac{EA}{L} & 0 & 0 & 0 & 0 & 0 & -\frac{EA}{L} & 0 & 0 & 0 & 0 & 0 \\ 0 & \frac{12EI_z}{L^3} & 0 & 0 & 0 & \frac{6EI_z}{L^2} & 0 & -\frac{12EI_z}{L^3} & 0 & 0 & 0 & \frac{6EI_z}{L^2} \\ 0 & 0 & \frac{12EI_y}{L^3} & 0 & -\frac{12EI_y}{L^2} & 0 & 0 & 0 & -\frac{12EI_y}{L^3} & 0 & -\frac{6EI_y}{L^2} & 0 \\ 0 & 0 & 0 & \frac{GJ}{L} & 0 & 0 & 0 & 0 & 0 & -\frac{GJ}{L} & 0 & 0 \\ 0 & 0 & -\frac{6EI_y}{L^2} & 0 & \frac{4EI_y}{L} & 0 & 0 & 0 & \frac{6EI_y}{L^2} & 0 & \frac{2EI_y}{L} & 0 \\ 0 & \frac{6EI_z}{L^2} & 0 & 0 & 0 & \frac{4EI_z}{L} & 0 & -\frac{6EI_z}{L^2} & 0 & 0 & 0 & \frac{2EI_z}{L} \\ -\frac{EA}{L} & 0 & 0 & 0 & 0 & 0 & \frac{EA}{L} & 0 & 0 & 0 & 0 & 0 \\ 0 & -\frac{12EI_z}{L^3} & 0 & 0 & 0 & -\frac{6EI_z}{L^2} & 0 & \frac{12EI_z}{L^3} & 0 & 0 & 0 & -\frac{6EI_z}{L^2} \\ 0 & 0 & -\frac{12EI_y}{L^3} & 0 & \frac{6EI_y}{L^2} & 0 & 0 & 0 & \frac{12EI_y}{L^3} & 0 & \frac{6EI_y}{L^2} & 0 \\ 0 & 0 & 0 & -\frac{GJ}{L} & 0 & 0 & 0 & 0 & 0 & \frac{GJ}{L} & 0 & 0 \\ 0 & 0 & -\frac{6EI_y}{L^2} & 0 & \frac{2EI_y}{L} & 0 & 0 & 0 & \frac{6EI_y}{L^2} & 0 & \frac{4EI_y}{L} & 0 \\ 0 & \frac{6EI_z}{L^2} & 0 & 0 & 0 & \frac{2EI_z}{L} & 0 & -\frac{6EI_z}{L^2} & 0 & 0 & 0 & \frac{4EI_z}{L} \end{bmatrix} \quad (3.9)$$

Transformation matrix:

By considering the forces at one end of the 3D beam element. Figure 3-4 (a) shows the two reference systems, the X,Y,Z and x,y,z axes representing the global and local system of coordinates respectively. Figure 3-4 (b), (c) and (d) show the force vectors F_x^l , F_y^l and F_z^l with their components X, Y, Z along the global coordinates. The vectors represent forces or displacements at the nodal coordinates of one of the joints of the structure along the local axes . For more clarification it is important to add the projections of the vectors along the global axes X,Y,Z. For example, the global vectors component F_x^g , F_y^g and F_z^g of vectors F_x^l , F_y^l and F_z^l along the global axes of the components X, Y and Z coordinates are given by:

$$F_x^g = F_x^l \cos \theta_{xX} + F_y^l \cos \theta_{yX} + F_z^l \cos \theta_{zX} \quad (3.10)$$

$$F_y^g = F_x^l \cos \theta_{xY} + F_y^l \cos \theta_{yY} + F_z^l \cos \theta_{zY} \quad (3.11)$$

$$F_z^g = F_x^l \cos \theta_{xZ} + F_y^l \cos \theta_{yZ} + F_z^l \cos \theta_{zZ} \quad (3.12)$$

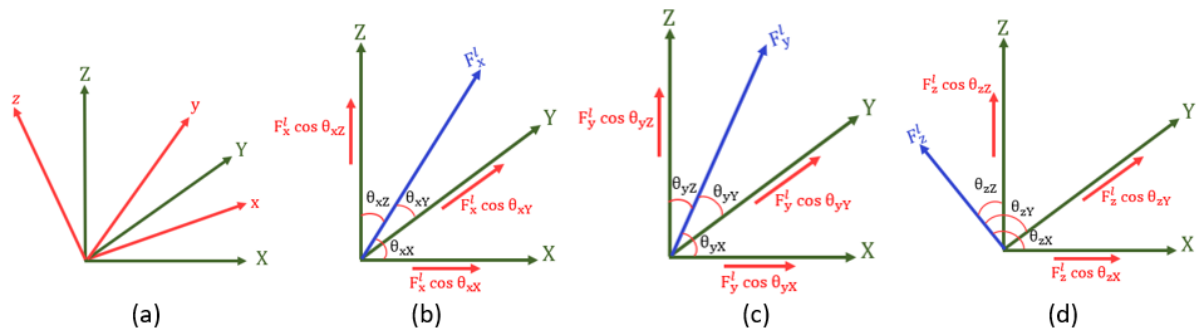


Figure 3-4: Force component in the local coordinates along with its components in the global coordinates

Defining

$$C_{xX} = \cos \theta_{xX} \quad C_{yX} = \cos \theta_{yX} \quad C_{zX} = \cos \theta_{zX}$$

$$C_{xY} = \cos \theta_{xY} \quad C_{yY} = \cos \theta_{yY} \quad C_{zY} = \cos \theta_{zY}$$

$$C_{xZ} = \cos \theta_{xZ} \quad C_{yZ} = \cos \theta_{yZ} \quad C_{zZ} = \cos \theta_{zZ}$$

$$[t] = \begin{bmatrix} C_{xX} & C_{yX} & C_{zX} \\ C_{xY} & C_{yY} & C_{zY} \\ C_{xZ} & C_{yZ} & C_{zZ} \end{bmatrix} \quad (3.13)$$

The rotation matrix [T] of three-dimensional elements is defined as

$$T = \begin{bmatrix} [t] & 0 & 0 & 0 \\ 0 & [t] & 0 & 0 \\ 0 & 0 & [t] & 0 \\ 0 & 0 & 0 & [t] \end{bmatrix} \quad (3.14)$$

Therefore, the total transformation matrix T for a space element is defined as:

$$T = \begin{bmatrix} \cos \theta_{xx} & \cos \theta_{yx} & \cos \theta_{zx} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ \cos \theta_{xy} & \cos \theta_{yy} & \cos \theta_{zy} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ \cos \theta_{xz} & \cos \theta_{yz} & \cos \theta_{zz} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & \cos \theta_{xx} & \cos \theta_{yx} & \cos \theta_{zx} & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & \cos \theta_{xy} & \cos \theta_{yy} & \cos \theta_{zy} & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & \cos \theta_{xz} & \cos \theta_{yz} & \cos \theta_{zz} & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & \cos \theta_{xx} & \cos \theta_{yx} & \cos \theta_{zx} & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & \cos \theta_{xy} & \cos \theta_{yy} & \cos \theta_{zy} & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & \cos \theta_{xz} & \cos \theta_{yz} & \cos \theta_{zz} & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \cos \theta_{xx} & \cos \theta_{yx} & \cos \theta_{zx} & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \cos \theta_{xy} & \cos \theta_{yy} & \cos \theta_{zy} & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \cos \theta_{xz} & \cos \theta_{yz} & \cos \theta_{zz} & 0 \end{bmatrix} \quad (3.15)$$

where θ_{xx} , θ_{yx} , and θ_{zx} , are the angles measured from global axes X , Y , and Z , with respect to the local axis x , respectively. The two-node of space frame element has six DOFs per node. Given the nodal displacements, these displacements are to be calculated by equation 3.7. The internal forces of the members are also to be found using equation 3.8.

3.2.3 Programming the stiffness method

The stiffness method is now the most common procedure used for structural analysis. The program for different types of elements vary only slightly by requiring different stiffness matrices and a few other details. Figure 3-5 incorporates the structural analysis workflow summarising and showing the process of forming the direct stiffness method of 2D and 3D structures in the written program. For a more user-friendly data input process and prevention of code tampering during data input process, all the input data was modelled in excel. The analysis input parameters are stated in Table 3.1 All this data will be called by MATLAB to start computing the internal forces of the structure.

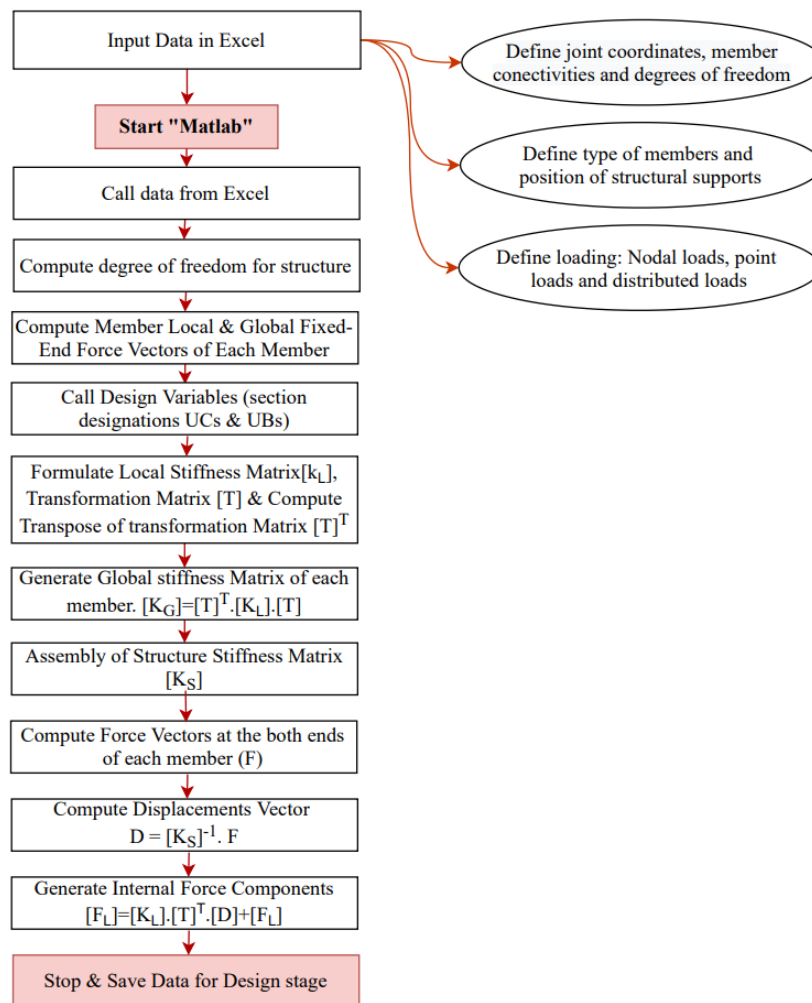


Figure 3-5: Analysis workflow chart

3.3 Design Procedure

The priority of constructing any structural steel frame is safety, where the designers should exactly determine and select the appropriate steel sections for the members. To ensure this, structural standards and codes of practice should be adopted, where many failure modes of the structure are taken into account when designing (Mohammad and Hemin, 2018). The structural design depends on the behaviour of the applied material, where the limit state design is divided into two main categories: ultimate limit state (ULS) and serviceability limit state (SLS). The ULS is defined as the state of design of a structural system at which the ultimate collapse due to loading occurs. The loading scenario in the ultimate limit state is the maximum possible factored loading which represents the worst-case scenario. The relevant verifications in the ULS design are shear, bending, buckling, etc. Whereas SLS is defined as the state of design

beyond which a structural system loses operationally its serviceability for the actual service load that the structure is subjected to. The serviceability state design is mainly related to the deflection checks (Issa, 2010).

EN 1993 (EC3) specifies the general guidelines for the design of all forms of steel structures, as well as explicit recommendations for the design of building structures and the constraints for assessing the objective functions in this study. The MATLAB code was designed in such a manner that during the design stage, the attained values from structural analysis and the parameters inputted in excel are called. Therefore, this part describes the procedure for designing of the structural space structures as specified in EC3.

3.3.1 Design Strength

A steel grade has to be adopted before starting the design. After that a steel section is designated to the members of the structure from the steel table. The properties of the Universal beams (UKB) and Universal columns (UKC) cross sections were assigned in accordance with the UK National Annex) (BS EN 10365: 2017).

3.3.1.1 Cross Sectional parameters and Properties

Steel cross sectional properties can greatly influence all strength formulae and determine the adequacy of a member under applied action. Therefore, the properties should be specified/calculated before initiating the analysis or design of the member. The properties values of the sections have already given in the sections table; thus the written code will call them out from their matrixes database saved in MATLAB when required. The definitions of the required properties are given as:

A :is the steel cross-sectional area

I :is the second moment of area

b :is the width of section

h is the depth of section

t_f :is the thickness of flange

t_w :is the thickness of web

c_w/t_w :is the ratio for local buckling in web

c_f/t_f :is the ratio for local buckling in flange

W_{el} :is the elastic section modulus

W_{pl} :is the plastic section modulus

i :is the radius of gyration

The following are the remaining parameters that must be given or calculated:

E :is the modulus of elasticity of the section = 210,000 N/mm²

ν :is the poison ratio in elastic stage = 0.3

G :is the shear modulus defined by

$$G = \frac{E}{2(1+\nu)}$$

3.3.1.2 Partial Factor of Safety

Structural factors of safety are specified for actions as well as for resistance of the steel sections. The safety factor for different actions for unfavourable conditions and the partial factors adopted here for resistance are given in Table 3.2:

Table 3.2: Partial factor and load factor of safety (Ref: Concise Eurocodes, 2009)

Safety factor for permanent action	γ_G	1.35
Safety factor for primary variable action	γ_Q	1.5
Partial factor for resistance of cross-sections	γ_{M0}	1
Partial factor for resistance of member to instability	γ_{M1}	1
Partial factor for resistance of cross-section in tension to fracture	γ_{M2}	1.25

3.3.1.3 Yield Strength

To attain the yield strength, a steel grade must be adopted. The yield strength is the strength at which the steel will go from elastic behaviour to plastic behaviour. For a particular steel grade the yield strength is obtained in relation to its flange thickness. The nominal values of the yield strength (f_y) for different steel grades are given in Table 3.3, however for this study, only S275 and S355 steel grade will be used.

Table 3.3: Nominal values of yield strength f_y and ultimate tensile strength f_u (EC3-EN 1993-1-1)

Nominal steel grade	Nominal thickness of the element, t (mm)			
	$t \leq 40$ mm		40 mm $< t \leq 80$ mm	
	f_y (N/mm ²)	f_u (N/mm ²)	f_y (N/mm ²)	f_u (N/mm ²)
S235	235	360	215	360
S275	275	430	255	410
S355	355	490	335	470
S450	440	550	410	550

3.3.1.4 Section Classification

The next step for design is to classify the steel cross sections that are assumed for the structural members. The cross section of a member is classified according to section's width to-thickness ratio when they are subject to compression stress due to either bending moment or axial forces. The Eurocode-3 classified the sections into four classes in order to determine whether local buckling effects the members capacity without calculating their local buckling resistance. The classes of are defined, as follow:

Class 1: The cross section is able to perform plastic hinge with rotation capacity which required by plastic analysis without reduction of the resistance. The resistance of the bending is therefore equal to the design value of the plastic moment, $W_{pl,min}f_y/\gamma_{M0}$, where this bending resistance can be maintained whilst rotation required for plastic design occurs at that cross-section.

Class 2: This type of cross sections able to develop their plastic moment resistance but limited to rotation capacity of local buckling which required for plastic design.

Class 3: In this type, the material yield strength of the cross section is attainable in the extreme compression fibres reaching the capacity of the design strength. This type doesn't have a plastic moment capacity, such a cross section can resist the design value of the elastic moment, $W_{el,min}f_y/\gamma_{M0}$.

Class 4: This type of cross sections contains slender members where the local buckling is most likely to be occurred before the attainment of the material yield strength on the extreme fibres in one or more parts of the cross-section.

Consequently, the behaviour of the four classes can be seen below in Figure 3-6. It illustrates the moment-rotation behaviour of the cross-section.

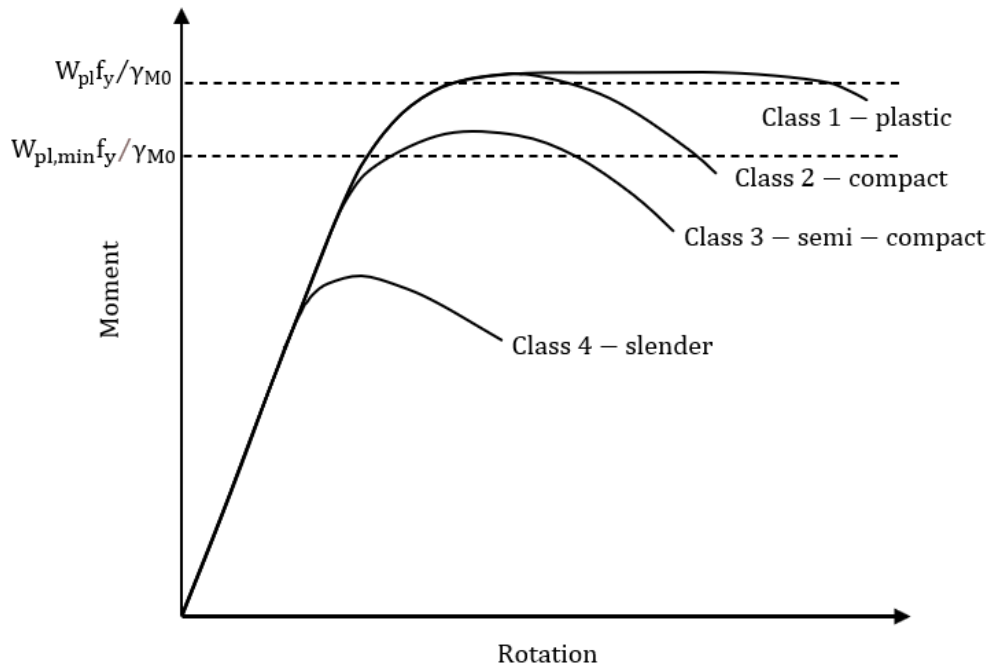


Figure 3-6: Moment rotation behaviour of cross-sections of different classes (Trahair et al., 2008)

Rigid frame members will be subjected to both compression and bending. Before classifying any cross-section, the given ratio for the chosen class must be satisfied, if not, the next class is checked. Limits of the cross-section classes in EC3 are expressed in terms of proportions of the cross-section elements as following:

Class 1:	web:	$d/t_w \leq 72\epsilon$
	Compression flange	$c/t_f \leq 9\epsilon$
Class 2:	web:	$d/t_w \leq 83\epsilon$
	Compression flange	$c/t_f \leq 10\epsilon$
Class 3:	web:	$d/t_w \leq 124\epsilon$
	Compression flange	$c/t_f \leq 14\epsilon$

According to EC3-section 5.5.2 (5), any section fails to meet the limitations for class 3 should be classified as class 4.

Class 4: web: $d/t_w > 124\epsilon$
 Compression flange $c/t_f > 14\epsilon$

Where:

d: is the depth of the web

t_w : is the web thickness

C: is the width of outstand flange

t_f : is the flange thickness

f_y : is the steel yield strength

ϵ : is the factor depending on steel strength

$$\epsilon = (235/f_y)^{0.5}$$

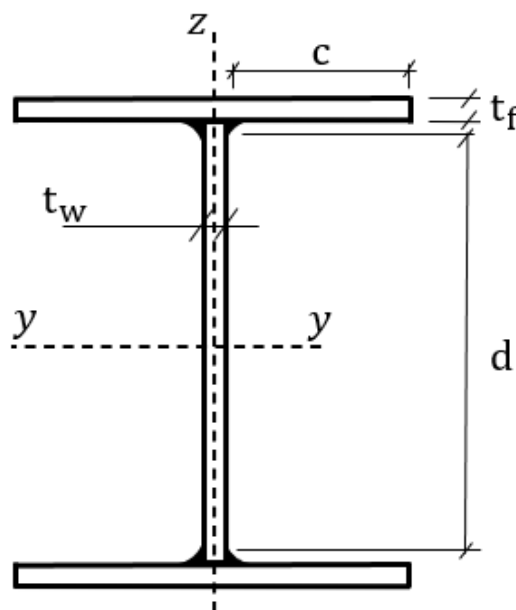


Figure 3-7: Gross cross-section of a beam

Therefore, these ratios of the web and the flange will automatically be calculated by calling out the properties of the chosen sections stored in MATLAB.

3.3.2 Ultimate Limit State

Ultimate limit state is concerned with the safety of the structure and presents direct risk to occupant's safety, it refers to the occurrence that can lead to partial or total collapse of the building. Examples of this includes excessive deformation, loss of stability, loss of equilibrium, development of mechanisms and fatigues. The limit states considered in this study are the compression, shear, bending, flexural and lateral buckling limit states.

3.3.2.1 Compression check

Compression capacity of selected section for structural members of steel frame must be checked since they could be subjected to axial or lateral loadings. Therefore, the criteria below must be fulfilled (EC3, clause 6.9):

$$\frac{N_{Ed}}{N_{c,Rd}} \leq 1.0 \quad (3.16)$$

Where:

N_{Ed} :is the design value of compression force at each cross-section, using the axial force obtained from analysis

$N_{c,Rd}$:is the design resistance value of the cross-section

According to EC3, the design resistance for a compression will be determined based on the classification of the cross-section. When the steel cross section is classified as either class 1, 2, or 3 then the design compression resistance will be determined as:

$$N_{c,Rd} = \frac{Af_y}{\gamma_{M0}} \quad (3.17)$$

and for class 4 cross-section

$$N_{c,Rd} = \frac{A_{eff} f_y}{\gamma_{M0}} \quad (3.18)$$

Where:

A_{eff} :is the effective section area

A :is the gross cross-sectional area

Effective section area can be determined by using EC3-5, where the flange and web of the I-section should be obtained separately. The effective section area can be obtained from:

$$A_{eff} = \sum A_{c,eff} \quad (3.19)$$

Where:

$A_{c,eff}$:is the effective section area of a flat compression structural element comprising the cross-section (web and flange), which will be attained from its gross cross-sectional area (A_g):

$$A_{c,eff} = \rho A_g \quad (3.20)$$

ρ :is the reduction factor, it can be obtained for an internal compression element (web) as:

$$\rho = 1.0 \quad \text{for} \quad \bar{\lambda}_p \leq 0.673$$

$$\rho = \frac{\bar{\lambda}_p - 0.055(3 + \psi)}{\bar{\lambda}_p^2} \leq 1.0 \quad \text{for} \quad \bar{\lambda}_p > 0.673 \quad (3.21)$$

And for an outstanding compression element (flange) can be obtained by:

$$\rho = 1.0 \quad \text{for} \quad \bar{\lambda}_p \leq 0.748$$

$$\rho = \frac{\bar{\lambda}_p - 0.188}{\bar{\lambda}_p^2} \leq 1.0 \quad \text{for} \quad \bar{\lambda}_p > 0.748 \quad (3.22)$$

$\bar{\lambda}_p$:is the factor of stress ratio and can be obtained by:

$$\bar{\lambda}_p = \frac{\bar{b}/t}{28.4\epsilon\sqrt{k_\sigma}} \leq 1.0 \quad (3.23)$$

\bar{b} :is the appropriate width and to be determined according on the element cross-section. For a flange it is the outstanding part of the flange and it is the depth between fillets for the web in I-section.

t :is the thickness of the flange or web

k_σ : is a factor for the buckling corresponding to stress ratio ψ and to the boundary conditions. For internal compression elements the factor can be taken as 4.0, where and a value of 0.43 can be taken for the outstanding compression elements in case of uniform compression as given in tables 4.1 and 4.2 of EC3, Part 1-5.

3.3.2.2 Moment Capacity check

The beam-column members of the structure are subjected to an axial and lateral load, also transmit moments between their ends or both. Therefore, to make sure the bending resistance of the cross sections adequate the following equation must be satisfied.

$$\frac{M_{Ed}}{M_{c,Rd}} \leq 1.0 \quad (3.24)$$

Where:

M_{Ed} : is the design value of bending moment at each cross-section, using the value obtained from analysis.

$M_{c,Rd}$: is the design resistance of cross section in bending about one principal axis.

According to EC3, the design resistance for a bending moment will be determined based on the classification of the cross-section. The design bending moment resistance $M_{c,Rd}$ of cross sections that under class 1 and 2 is based on the full plastic section modulus, where the moment resistance of cross sections that under class 3 is based on the elastic section modulus. Whereas the moment resistance of cross sections with class 4 utilises the effective section modulus.

For class 1 and 2 cross sections:

$$M_{c,Rd} = M_{pl,Rd} = \frac{W_{pl} f_y}{\gamma_{M0}} \quad (3.25)$$

For class 3 cross sections:

$$M_{c,Rd} = M_{el,Rd} = \frac{W_{el,min} f_y}{\gamma_{M0}} \quad (3.26)$$

For class 4 cross sections:

$$M_{c,Rd} = \frac{W_{eff,min} f_y}{\gamma_{M0}} \quad (3.27)$$

Where:

$M_{pl,Rd}$:is the design plastic resistance moment

W_{pl} :is the design plastic section modulus of the cross-sections, which is obtained from the properties table of the steel-sections.

$M_{el,Rd}$:is the design elastic resistance moment

$W_{el,min}$:is the minimum elastic section modulus of the cross-sections, which is obtained from the properties table of the steel-sections.

$W_{eff,min}$:is the minimum effective section modulus

The moment capacity might be influenced by the magnitude of the shear force. This effect can be neglected if the shear force value less than half the plastic shear resistance. Otherwise, the effect of shear on the moment resistance should be taken into account, by reducing the moment resistance as represented below:

$$M_{c,Rd} = W_y \frac{(1-\rho) f_y}{\gamma_{M0}} \quad (3.28)$$

where:

W_y :is the section modulus and it can be expressed as:

For class 1 and class 2:

$$W_y = W_{pl}$$

For class 3:

$$W_y = W_{el,min}$$

For class 4:

$$W_y = W_{eff,min}$$

ρ :is a factor for shear effect reduction and can be obtained by:

$$\rho = \left(\frac{V_{Ed}}{0.5V_{pl,Rd}} - 1 \right)^2 \quad (3.29)$$

3.3.2.3 Shear check

The resistance of cross-sections to shear should be carried out, although hot rolled steel sections are strong in shear resistance, therefore shear failure would generally be the last failure mode of the structure. However, members with short span can fail in shear and because of this consideration is given to all spans. Therefore, to make sure the shear resistance of the cross sections adequate the following equation in clause 6.17 of EC3 must be satisfied.

$$V_{Ed}/V_{pl,Rd} \quad (3.30)$$

Where:

V_{Ed} :is the shear force design value of each cross-section.

$V_{pl,Rd}$:is the plastic design shear resistance of each cross-section.

$$V_{pl,Rd} = A_v \frac{f_y}{\sqrt{3}} / \gamma_{M0} \quad (3.31)$$

A_v :is the shear area of I and H sections as shown in Figure 3-8 and can be taken as follows:

$$A_v = A - 2b t_f + (t_w + 2r)t_f \quad \text{with } A_v \geq \eta \cdot h_w \cdot t_w \quad (3.32)$$

Where:

r :is the root radius;

η :is a coefficient value defined in EN 1993-1-5 and recommended to be considered as 1.2 for steel grades between S235 and S460 and equal to 1 for steel grades over S460.

h_w :is the depth of the web.

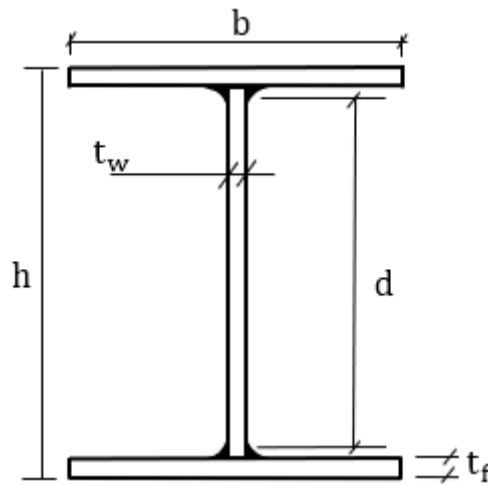


Figure 3-8: Shear area of steel cross-section according to EC3

For cross-sections with class 2 and class 3:

$$\tau_{Ed} \leq \frac{f_y}{\sqrt{3b} \gamma_{M0}} \quad (3.33)$$

Where:

τ_{Ed} :is the tangential stress due to shear force and can be obtained as:

$$\tau_{Ed} = \frac{V_{Ed} S}{I t} \quad (3.34)$$

I :is the second moment of area of the cross-section

S :is the first moment of the area above on either side of the examined point, in other words considering only the portion between the point at which the shear is required and the boundary of the cross-section

t :is the thickness at the examined point

3.3.2.4 Shear Buckling check

EC3 recommends checking the shear buckling resistance of the web without intermediate stiffeners at the point where the maximum shear occurs in addition to shear resistance. To check the structural elements for shear buckling, EC3 has stated the limits and suggested using the procedures outlined in EN 1993-1-5 (EC3, 1-5). If the web's height-to-thickness ratio exceeds the following limit for unstiffened webs, then it's necessary to check for shear buckling:

$$\frac{h_w}{t_w} = 72 \frac{\varepsilon}{\eta} \quad (3.35)$$

Where:

h_w :is the web's height

η :is the shear area factor, which can be considered as 1.2 for steel grades up to S460.

According to EC3-1-5, the applied shear forces must not exceed the section's shear buckling resistance.

$$V_{Ed} \leq V_{b,Rd} \quad (3.36)$$

where:

$V_{b,Rd}$:is the buckling shear resistance

An increase in the h_w/t_w ratio reduces the shear resistance of an unstiffened slender section. Shear buckling resistance is influenced by both the flange and the web. according to EC3 (1-5), the design shear resistance should be taken as follows for both stiffened and unstiffened webs.

$$V_{b,Rd} = V_{bw,Rd} + V_{bf,Rd} \leq \eta \frac{f_{yw}/\sqrt{3}}{\gamma_{M1}} h_w t_w \quad (3.37)$$

where:

$V_{bw,Rd}$:is the web's buckling shear resistance

$V_{bf,Rd}$:is the flange's buckling shear resistance

The web's contribution can be obtained as:

$$V_{bw,Rd} = \chi_w \frac{f_{yw}/\sqrt{3}}{\gamma_{M1}} h_w t_w \quad (3.38)$$

Where:

χ_w :is the web contribution to shear buckling factor, which can be found in table 5-1 or figure 5.2 of EC3 (1-5). Its value can be also obtained as:

$$\chi_w = \begin{cases} \eta & \bar{\lambda}_w < 0.83 \\ 0.83/\bar{\lambda}_w & \bar{\lambda}_w \geq 0.83 \end{cases} \quad (3.39)$$

where:

$\bar{\lambda}_w$:is the modified web plate slenderness, and can be obtained as:

$$\bar{\lambda}_w = 0.76 \sqrt{\frac{f_{yw}}{\tau_{cr}}} \quad (3.40)$$

τ_{cr} :is the critical elastic local buckling stress. EC3 didn't specify its value, where it was formulated by (Trahair et al., 2008) as following:

$$\tau_{cr} = \frac{\pi^2 E k_\tau}{12(1-\nu^2)(d_w t_w)^2} \quad (3.41)$$

ν :is the Poisson's ratio of steel sections, where EC3 specifies it as 0.30

k_τ :is the buckling coefficient, where it can be obtained as:

$$k_\tau = \begin{cases} 5.34 + 4 \left(\frac{d_w}{L}\right)^2 & \text{for } L \geq d_w \\ 4 + 5.34 \left(\frac{d_w}{L}\right)^2 & \text{for } L < d_w \end{cases} \quad (3.42)$$

L :is the member's length

However, for simplicity the following equation is used in EC3 (1-5) to obtain the slenderness value for members with transverse stiffness at support:

$$\bar{\lambda}_w = \frac{1}{86.4\epsilon} \frac{h_w}{t_w} \quad (3.43)$$

When the flange's bending moment resistance is more than the applied bending moment, then the flange's contribution should be calculated as follows:

$$V_{bf,Rd} = \frac{f_{yf}}{\gamma_{M1}} \frac{b_f t_f^2}{c} \left[1 - \left(\frac{M_{Ed}}{M_{f,Rd}} \right) \right] \quad (3.44)$$

Where:

$M_{f,Rd}$:is the flange's bending moment resistance

f_{yf} :is the flange's yield strength

c :is a coefficient, and can be defined as:

$$c = a \left(0.25 + 1.6 \frac{f_{yf}}{f_{yw}} \frac{b_f}{h_w^2} \frac{t_f^2}{t_w} \right) \quad (3.45)$$

a : is the distance between transverse stiffeners

3.3.2.5 Bending moment with axial compression effect check

In order to check the capacity of the cross-section to withstand compressive stress, it is necessary to consider the compressive stress generated simultaneously by bending moments and axial forces. In this case, allowance should be made for the effect of axial force on the moment resistance of the cross-section. EC3 states that the allowance is not needed for the effect of axial force on the moment resistance about the major axis if the following criteria are satisfied:

It is crucial to examine the compressive stress generated concurrently by bending moments and axial forces in order to check the cross-capacity section's to sustain compressive stress. In this scenario, the influence of axial force on the cross-moment section's resistance should be taken into account. According to EC3, if the following criteria are met, the allowance for the influence of axial force on the moment resistance about the major axis is not required:

$$\frac{N_{Ed}}{N_{pl,Rd}} \leq 0.25 \quad (3.46)$$

and

$$N_{Ed} \leq \frac{h_w t_w f_y}{2\gamma_{M0}} \quad (3.47)$$

Otherwise, for a doubly symmetric I or H steel cross-section with class 1 or class 2 classifications, the following equation allows the effect of axial force on the plastic moment resistance about the major axis:

$$\frac{M_{Ed}}{M_{N,Rd}} \leq 1 \quad (3.48)$$

where:

$$M_{N,Rd} = M_{pl,Rd} = \frac{1-n}{1-0.5a} \leq M_{pl,Rd} \quad (3.49)$$

$$n = \frac{M_{Ed}}{N_{pl,Rd}} \quad (3.50)$$

$$a = \frac{A-2b_f}{A} \leq 0.5 \quad (3.51)$$

For cross-sections with class 3 and 4 the following condition must be satisfied:

$$\frac{N_{Ed}}{A_y f_y / \gamma_{M0}} + \frac{M_{Ed} + N_{Ed} e_{Ny}}{W_y f_y / \gamma_{M0}} \leq 1 \quad (3.52)$$

where:

A_y :is the effective area for cross-section under class 4 and the gross area for cross-sections under class 3 .

e_{Ny} :is the distance between the neutral axis of effective cross-section and the neutral axis of gross cross-section. As there is no reduction in gross cross-sectional area, it is recommended to be zero for cross-sections under class 3,

W_y :is the cross-section modulus, it can be considered as:

$$W_y = \begin{cases} W_{el,y} & \text{For a Class 3 cross – section} \\ W_{eff,y} & \text{For a Class 4 cross – section} \end{cases} \quad (3.53)$$

$W_{el,y}$:is the elastic section modulus

$W_{eff,y}$:is the effective section modulus

3.3.2.6 Flexural buckling resistance check

Any compression member of the structure must be able to resist the overall buckling. Depending on its slenderness and stiffness, the overall buckling should be checked. The general verification for any compression member against buckling, according to EC3, is as follows:

$$N_{Ed} \leq N_{b,Rd} \quad (3.54)$$

Where:

$N_{b,Rd}$:is the compression member's design buckling resistance, which is provided by:

For cross-sections under class 1, 2, and 3 the following can be used:

$$N_{b,Rd} = \chi \frac{A f_y}{\gamma_{M1}} \quad (3.55)$$

For class 4 cross-sections:

$$N_{b,Rd} = \chi \frac{A_{eff} f_y}{\gamma_{M1}} \quad (3.56)$$

χ :is the reduction factor due to the flexural buckling, which can be provided by:

$$\chi = \frac{1}{\phi + \sqrt{\phi^2 - \bar{\lambda}^2}} \leq 1 \quad (3.57)$$

$$\phi = 0.5[1 + \alpha(\bar{\lambda} - 0.2) + \bar{\lambda}^2] \quad (3.58)$$

Where:

α :is an imperfection factor and depends on the buckling curve

$\bar{\lambda}$:is the non-dimensional slenderness, it can be considered as follows:

For cross-sections under Class 1, 2, and 3:

$$\bar{\lambda} = \sqrt{\frac{A f_y}{N_{cr}}} = \frac{L_{eff}}{i} \frac{1}{\lambda_1} \quad (3.59)$$

For class 4 cross-sections:

$$\bar{\lambda} = \sqrt{\frac{A_{eff} f_y}{N_{cr}}} = \frac{L_{eff}}{i} \frac{\sqrt{A_{eff}/A}}{\lambda_1} \quad (3.60)$$

A :is the gross area of the cross-section

A_{eff} :is the effective area of the cross-section

N_{cr} :is the elastic critical buckling force

L_{eff} :is the segment length, which is located between two adjacent cross sections that are restrained

i : is the radius of gyration about the axis where the buckling plane is located

λ_1 : is the slenderness value determined to obtain the relative slenderness, $\bar{\lambda}$. According to EC3, it can be calculated using the following equation:

$$\lambda_1 = \sqrt{\frac{\pi^2 E}{f_y}} = 93.3\varepsilon \quad (3.61)$$

The buckling curve for a rolled section with a double symmetric shape is specified in Table 3.4.

Table 3.4: Determination of buckling curve for rolled steel cross-sections (EC3-EN 1993-1-1)

Limits		Buckling about axis	Buckling curve	
			S235 S275 S355	S460
$h/B_f \leq 1.20$	$t_f \leq 100 \text{ mm}$	Major	a	a
		Minor	c	a
	$t_f > 100 \text{ mm}$	Major	d	c
		Minor	d	c
$h/B_f > 1.20$	$t_f \leq 40 \text{ mm}$	Major	a	a_0
		Minor	b	a_0
	$40 \text{ mm} < t_f \leq 100 \text{ mm}$	Major	b	a
		Minor	c	a

Where the imperfection factor (α), for the appropriate buckling curve can be achieved from Table 3.5 below:

Table 3.5: Imperfection factors used for different buckling curves (EC3-EN 1993-1-1)

Buckling curve	a_0	a	b	c	d
Imperfection factor, α	0.13	0.21	0.34	0.49	0.76

3.3.2.7 Lateral torsional-flexural buckling check

Failure of steel structural members, especially beams, can be caused by lateral buckling, which is generated from flexural compression stress. Therefore, the laterally unrestrained structural elements must be checked against the lateral torsional buckling. To ensure that the member has appropriate resistance, EC3 recommended the following check:

$$M_{Ed} \leq M_{b,Rd} \quad (3.62)$$

where:

M_{Ed} :is the bending moment's design value

$M_{b,Rd}$:is the buckling resistance bending moment, which can be attained from:

$$M_{b,Rd} = \chi_{LT} W_y \frac{f_y}{\gamma_{M1}} \quad (3.63)$$

Where W_y should be obtained as following:

For cross-sections under class 1 and 2:

$$W_y = W_{pl,y} \quad (3.64)$$

For cross-sections under class 3:

$$W_y = W_{el,y} \quad (3.65)$$

For class 4 cross-section:

$$W_y = W_{el,y} \quad (3.66)$$

χ_{LT} :is a reduction factor due to torsional-flexural buckling

$$\chi_{LT} = \frac{1}{\Phi_{LT} + \sqrt{\Phi_{LT}^2 - \bar{\lambda}_{LT}^2}} \quad \text{but } \chi_{LT} \leq 1 \quad (3.67)$$

Where,

$$\Phi_{LT} = \frac{1}{2} [1 + \alpha_{LT} (\bar{\lambda}_{LT} - 0.2) + \bar{\lambda}_{LT}^2] \quad (3.68)$$

α_{LT} is an imperfection factor due to torsional-flexural buckling. The recommended values of the imperfection factor α_{LT} corresponding to the appropriate buckling curve can be obtained from Table 3.6 and Table 3.7.

Table 3.6: Recommended imperfection factors for lateral torsional buckling curves (EC3-EN 1993-1-1)

Buckling curve	a	b	c	d
Imperfection factor, α_{LT}	0.21	0.34	0.49	0.76

To find out the category of the lateral torsional buckling curves for any steel cross-section, EC3 recommended values based on the type of the cross-section and some other limits as given in Table 3.7.

Table 3.7: Lateral torsional buckling curves recommended for different cross-sections (EC3-EN 1993-1-1)

Cross-Section	Limits	Buckling Curve
Rolled I-section	$h/b > 2$	b
	$h/b \leq 2$	c
Welded I-section	$h/b > 2$	c
	$h/b \leq 2$	d

The slenderness ratio for lateral-torsional buckling $\bar{\lambda}_{LT}$ can be obtained as:

$$\bar{\lambda}_{LT} = \sqrt{W_y \frac{f_y}{M_{cr}}} \quad (3.69)$$

$\bar{\lambda}_{LT}$ is the slenderness ratio for lateral-torsional buckling

M_{cr} is the elastic critical buckling moment for lateral-torsional buckling.

Eurocode3 doesn't pointed out the procedures to obtain the elastic criteria buckling moment value for lateral torsional buckling M_{cr} . However, Concise EC3 guide stated that the following formula can be used as shown in eq. 33. Alternatively, a free software called LTBeam from CTICM can be used to determine the M_{cr} .

$$M_{cr} = C_1 \frac{\pi^2 E I_z}{L^2} \sqrt{\frac{I_w}{I_z} + \frac{L^2 G I_t}{\pi^2 E I_z}} \quad (3.70)$$

where:

E :is the modulus of elasticity and in this study was considered as 2100000 N/mm²

G :is the shear modulus and in this study was considered as 81000 N/mm²

L :is the buckling length of the element, which is the distance between lateral supports

I_w :is the warping constant

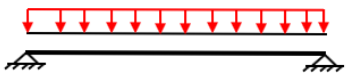

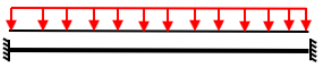





I_z ;is the second moment of inertia about the minor axis

I_t :is the torsional buckling constant

C_1 :is the coefficient allows for the shape of the bending moment diagram.

The coefficient C_1 is not given in the Eurocode3. However, Concise EC3 guide provided an estimated values for certain moment conditions as shown in Table 3.8.

Table 3.8: Recommended values of the coefficient C_1 for various moment conditions (EC3-EN 1993-1-1)

Intermediate Transverse Loading		C_1
		1.13
		2.60
		1.35
		1.69

Based on the national organisations for the steel construction industry (BCSA and TATA Steel), the coefficient C_1 can also be determined by using a source called NCCI ACCESS STEEL to choose C_1 factor under any moment condition.

The other technique of calculating the lateral torsional buckling reduction factor, χ_{LT} , is less conservative than the previous one and is only applicable to rolled steel cross-sections. The required non-dimensional slenderness can be calculated as follows:

$$\chi_{LT} = \frac{1}{\Phi_{LT} + \sqrt{\Phi_{LT}^2 - \beta \bar{\lambda}_{LT}^2}} \quad \text{but} \quad \begin{cases} \chi_{LT} \leq 1,0 \\ \chi_{LT} \leq \frac{1}{\bar{\lambda}_{LT}^2} \end{cases} \quad (3.71)$$

where:

$$\Phi_{LT} = \frac{1}{2} [1 + \alpha_{LT} (\bar{\lambda}_{LT} - \bar{\lambda}_{LT,0}) + \beta \bar{\lambda}_{LT}^2] \quad (3.72)$$

For rolled steel cross-sections, EC3 recommended the following values for the parameters $\bar{\lambda}_{LT,0}$ and β .

$$\bar{\lambda}_{LT,0} = 0.4$$

$$\beta = 0.75$$

However, for considering the moment distribution between the lateral restraints of any element, EC3 proposed that the reduction factor due to torsional-flexural buckling χ_{LT} should be modified as follows:









$$\chi_{LT,mod} = \frac{\chi_{LT}}{f} \quad \text{but} \quad \begin{cases} \chi_{LT,mod} \leq 1,0 \\ \chi_{LT,mod} \leq \frac{1}{\bar{\lambda}_{LT}^2} \end{cases} \quad (3.73)$$

Where the value of f is recommended to be calculated as following:

$$f = 1 - 0.5(1 - k_c) [1 - 2(\bar{\lambda}_{LT} - 0.8)^2] \quad (3.74)$$

k_c is a correction factor which relies on the bending moment diagram shape and can be specified according to Table 3.9.

Table 3.9: Recommended values of k_c for various moment conditions (EC3-EN 1993-1-1).

Moment distribution	K_c
 $\psi = 1$	1.0
 $-1 \leq \psi \leq 1$	$\frac{1}{1.33-0.33\psi}$
  	0.94 0.90 0.91
  	0.86 0.77 0.82

3.3.2.8 Combined Bending and Shear check

According to EC3, clause 6.2.8, the presence of shear might diminish the plastic moment resistance. When the shear force is 50% more than the shear resistance, a reduced yield strength is utilised to calculate the resulting resistive moment, $M_{v,Rd}$.

For rolled I and H steel cross-section with bending about major axis, the reduced resistance moment $M_{y,v,Rd}$ can be determined as:

$$M_{y,v,Rd} = \frac{\left(W_{pl,y} - \frac{\rho A_w^2}{4t_w} \right) f_y}{\gamma_{M0}} \leq M_{y,c,Rd} \quad (3.75)$$

3.3.2.9 Flexural and Lateral Torsional Buckling (Combined compression and bending) check

There are two approaches given in the Eurocode-3; General method and Interaction method. Only the Interaction method was considered in this study. Due to the combined bending and flexure, the verification can be done by two expressions as following:

For the major (y-y) axis is:

$$\frac{N_{Ed}}{N_{b,y,Rd}} + k_{yy} \frac{M_{y,Ed}}{M_{b,Rd}} + k_{yz} \frac{M_{z,Ed}}{M_{cb,z,Rd}} \leq 1 \quad (3.76)$$

For minor (z-z) axis is:

$$\frac{N_{Ed}}{N_{b,z,Rd}} + k_{zy} \frac{M_{y,Ed}}{M_{b,Rd}} + k_{zz} \frac{M_{z,Ed}}{M_{cb,z,Rd}} \leq 1 \quad (3.77)$$

Where:

N_{Ed} :is the design compression force value.

$M_{y,Ed}, M_{z,Ed}$:are the design maximum moments values about the axes y-y and z-z, respectively.

$M_{cb,z,Rd}$:is the design bending moment about the minor (z-z) axis

$k_{zy}, k_{zy}, k_{zy}, k_{zy}$:are the interaction factors comprising elements which address the shape of the bending moment diagram and the influence of the axial load, which may be determined from Table 3.10 and Table 3.11.

$M_{b,Rd}$:is the design buckling resistance moment.

$N_{b,y,Rd}$ and $N_{b,z,Rd}$:are the flexural buckling resistance about the major (y-y) and minor (z-z) axes respectively, given by:

For cross-sections under class 1, 2 and 3

$$N_{b,Rd} = \frac{\chi A f_y}{\gamma_{M1}} \quad (3.78)$$

For cross-sections under class 4

$$N_{b,Rd} = \frac{\chi A_{eff} f_y}{\gamma_{M1}} \quad (3.79)$$

Table 3.10 and Table 3.11 specify how to select the interaction factors $k_{zy}, k_{zy}, k_{zy}, k_{zy}$ for I-sections and H-sections.

Table 3.10: Interaction factors recommended for members that are not susceptible to torsional-deformation (EC3-EN 1993-1-1)

Interaction factor	Type of sections	Design assumptions	
		Elastic cross-sectional properties class 3 and class 4	Plastic cross-sectional properties class 1 and class 2
K_{yy}	I-sections	$C_{my} \left(1 + 0.6 \bar{\lambda}_y \frac{N_{Ed}}{\chi_y N_{RK}/\gamma_{M1}} \right)$	$C_{my} \left(1 + (\bar{\lambda}_y - 0.2) \frac{N_{Ed}}{\chi_y N_{RK}/\gamma_{M1}} \right)$
	RHS-sections	$\leq C_{my} \left(1 + 0.6 \frac{N_{Ed}}{\chi_y N_{RK}/\gamma_{M1}} \right)$	$\leq C_{my} \left(1 + 0.8 \frac{N_{Ed}}{\chi_y N_{RK}/\gamma_{M1}} \right)$
K_{yz}	I-sections RHS-sections	K_{zz}	$0.6K_{zz}$
K_{zy}	I-sections RHS-sections	$0.8K_{yy}$	$0.6K_{yy}$
K_{zz}	I-sections	$C_{mz} \left(1 + 0.6 \bar{\lambda}_y \frac{N_{Ed}}{\chi_z N_{RK}/\gamma_{M1}} \right)$	$C_{mz} \left(1 + (2\bar{\lambda}_y - 0.6) \frac{N_{Ed}}{\chi_z N_{RK}/\gamma_{M1}} \right)$ $\leq C_{mz} \left(1 + 1.4 \frac{N_{Ed}}{\chi_z N_{RK}/\gamma_{M1}} \right)$
	RHS-sections	$\leq C_{mz} \left(1 + 0.6 \frac{N_{Ed}}{\chi_z N_{RK}/\gamma_{M1}} \right)$	$C_{mz} \left(1 + (\bar{\lambda}_z - 0.2) \frac{N_{Ed}}{\chi_z N_{RK}/\gamma_{M1}} \right)$ $\leq C_{mz} \left(1 + 0.8 \frac{N_{Ed}}{\chi_z N_{RK}/\gamma_{M1}} \right)$
For I- and H-sections and rectangular hollow sections under axial compression and uniaxial bending $M_{y,Ed}$ the coefficient K_{zy} may be $K_{zy} = 0$			

Table 3.11: Interaction factors recommended for members susceptible to torsional-deformation (EC3-EN 1993-1-1)

Interaction factor	Type of sections	Design assumptions	
		Elastic cross-sectional properties class 3 and class 4	Plastic cross-sectional properties class 1 and class 2
K_{yy}	I-sections RHS-sections	$C_{my} \left(1 + 0.6 \bar{\lambda}_y \frac{N_{Ed}}{\chi_y N_{RK}/\gamma_{M1}} \right)$ $\leq C_{my} \left(1 + 0.6 \frac{N_{Ed}}{\chi_y N_{RK}/\gamma_{M1}} \right)$	$C_{my} \left(1 + (\bar{\lambda}_y - 0.2) \frac{N_{Ed}}{\chi_y N_{RK}/\gamma_{M1}} \right)$ $\leq C_{my} \left(1 + 0.8 \frac{N_{Ed}}{\chi_y N_{RK}/\gamma_{M1}} \right)$
K_{yz}	I-sections RHS-sections	K_{zz}	$0.6K_{zz}$
K_{zy}	I-sections RHS-sections	$\left[1 - \frac{0.6 \bar{\lambda}_z}{(C_{mLT} - 0.25) \chi_z N_{RK}/\gamma_{M1}} \frac{N_{Ed}}{\chi_z N_{RK}/\gamma_{M1}} \right]$ $\geq \left[1 - \frac{0.05}{(C_{mLT} - 0.25) \chi_z N_{RK}/\gamma_{M1}} \frac{N_{Ed}}{\chi_z N_{RK}/\gamma_{M1}} \right]$	$\left[1 - \frac{0.1 \bar{\lambda}_z}{(C_{mLT} - 0.25) \chi_z N_{RK}/\gamma_{M1}} \frac{N_{Ed}}{\chi_z N_{RK}/\gamma_{M1}} \right]$ $\geq \left[1 - \frac{0.1}{(C_{mLT} - 0.25) \chi_z N_{RK}/\gamma_{M1}} \frac{N_{Ed}}{\chi_z N_{RK}/\gamma_{M1}} \right]$ for $\bar{\lambda}_z < 0.4$: $K_{zy} = 0.6 + \bar{\lambda}_z \leq 1 - \frac{0.1 \bar{\lambda}_z}{(C_{mLT} - 0.25) \chi_z N_{RK}/\gamma_{M1}} \frac{N_{Ed}}{\chi_z N_{RK}/\gamma_{M1}}$
K_{zz}	I-sections	$C_{mz} \left(1 + 0.6 \bar{\lambda}_y \frac{N_{Ed}}{\chi_z N_{RK}/\gamma_{M1}} \right)$	$C_{mz} \left(1 + (2\bar{\lambda}_y - 0.6) \frac{N_{Ed}}{\chi_z N_{RK}/\gamma_{M1}} \right)$ $\leq C_{mz} \left(1 + 1.4 \frac{N_{Ed}}{\chi_z N_{RK}/\gamma_{M1}} \right)$
	RHS-sections	$\leq C_{mz} \left(1 + 0.6 \frac{N_{Ed}}{\chi_z N_{RK}/\gamma_{M1}} \right)$	$C_{mz} \left(1 + (\bar{\lambda}_z - 0.2) \frac{N_{Ed}}{\chi_z N_{RK}/\gamma_{M1}} \right)$ $\leq C_{mz} \left(1 + 0.8 \frac{N_{Ed}}{\chi_z N_{RK}/\gamma_{M1}} \right)$

Where:

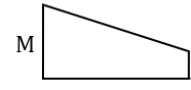
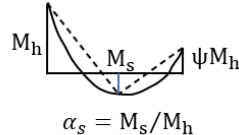
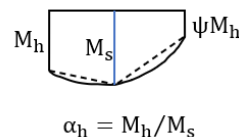
χ_y and χ_z :is the reduction factors due to flexural buckling.

N_{RK} :is the characteristic value of resistance to compression.

C_{my} , C_{mz} and C_{mLT} :are the factors of the equivalent uniform moment of the members, and they can be specified using Table 3.12.

Table 3.12 specifies the factors of the equivalent uniform moment C_{my} , C_{mz} and C_{mLT} , where ψ is the ratio of end moments in the segment, in other words, it is ratio of the smaller bending moment divided by the larger bending moment.

Table 3.12: Equivalent uniform moment factors C_m for members not susceptible to torsional (EC3-EN 1993-1-1)

Moment diagram	range		C_{my} and C_{mz} and C_{mLT}	
			Uniform loading	Concentrated load
	$-1 \leq \psi \leq 1$		$0.6 + 0.4\psi \geq 0.4$	
 $\alpha_s = M_s/M_h$	$0 \leq \alpha_s \leq 1$	$-1 \leq \psi \leq 1$	$0.2 + 0.8\alpha_s \geq 0.4$	$0.2 + 0.8\alpha_s \geq 0.4$
	$-1 \leq \alpha_s < 0$	$0 \leq \psi \leq 1$	$0.1 - 0.8\alpha_s \geq 0.4$	$-0.8\alpha_s \geq 0.4$
		$-1 \leq \psi < 0$	$0.1(1 - \psi) - 0.8\alpha_s \geq 0.4$	$0.2(-\psi) - 0.8\alpha_s \geq 0.4$
 $\alpha_h = M_h/M_s$	$0 \leq \alpha_h \leq 1$	$-1 \leq \psi \leq 1$	$0.95 + 0.05\alpha_s$	$0.90 + 0.10\alpha_h$
	$-1 \leq \alpha_h < 0$	$0 \leq \psi \leq 1$	$0.95 + 0.05\alpha_h$	$0.90 + 0.10\alpha_h$
		$-1 \leq \psi < 0$	$0.95 + 0.05\alpha_h(1 + 2\psi)$	$0.90 + 0.10\alpha_h(1 + 2\psi)$
For members with sway buckling mode the equivalent uniform moment factor should be taken $C_{my} = 0.9$ or $C_{mz} = 0.9$ respectively.				
C_{my} , C_{mz} and C_{mLT} should be obtained according to the bending moment diagram between the relevant braced points as follows:				
Moment factor	bending moment	points braced in direction		
C_{my}	y - y	z - z		
C_{mz}	z - z	y - y		
C_{mLT}	y - y	y - y		

3.3.3 Serviceability Limit State

Serviceability limit state (SLS) is related to the satisfactory performance of the structure at working load. It is mainly concerned with the functional use of the structure under normal service loads. There are two main types of serviceability limit states applicable to steel structures. They are deflection, durability, vibration, and fire resistance. Deflection checks are the only one considered in this research, to ensure that the structure stays functionally in use with minimum maintenance during its design life. Even if ultimate collapse may not happen, massive serviceability damage might lead to expensive maintenance, discomfort for the residents, and undesirable appearance of the structure.

Steel elements need to be checked for the deflections, as an excessive deflection can affect the appearance or the effective use of the structures. The deflection has been checked using the characteristic combination of actions but without including the permanent actions, where only the unfactored variable actions (life loads) are used to calculate the maximum deflection of the any member. It is to ensure that the total deflection must be less than the deflection limit as given below:

$$\delta_{total} \leq \delta_{limit} \quad (3.80)$$

EC3 specifies the vertical and horizontal deflection limits as shown in Table 3.13 and Table 3.14, respectively.

Table 3.13: Recommended vertical deflection limits due to characteristic combination (EC3-EN 1993-1-1)

Vertical deflection	
Cantilevers	Length/180
Beams carrying plaster or other brittle finish	Span/360
Other beams (except purlins and sheeting rails)	Length/180
Purlins and sheeting rails	To suit the characteristics of the particular cladding

Table 3.14: Suggested limits for horizontal deflection (EC3-EN 1993-1-1)

Horizontal deflection	
Tops of columns in single-storey buildings except portal frames	Height/300
Columns in portal frame buildings, not supporting crane runways	To suit the characteristics of the particular cladding
In each storey of a building with more than one storey	Height of that storey/300

Since this study considered all the members are fully fixed, the vertical deflection for the beams (δ) will be calculated as:

$$\delta = \frac{WL^2}{384EI} \quad (3.81)$$

The limit for deflection is not specifically given in EC3 but recommended values are given in clause 2.24 of the National annex to EC3. Assuming the beams and columns will support brittle finishes, the limit below are taken respectively for beams and columns. Vertical and horizontal deflection limits are not specifically given in Eurocode-3, but suggested values are provided in clause 2.24 of the National annex. The following equations are considered for beams and columns, respectively:

For vertical deflection

$$\delta_v < \frac{L}{360} \quad (3.82)$$

For horizontal deflection

$$\delta_h < \frac{H}{300} \quad (3.83)$$

Where:

L :is the beam span

H :is the column height

All the design steps were illustrated in a flowchart as shown in figure 3.9, checking all the requirements of steel structures according to Eurocode-3.

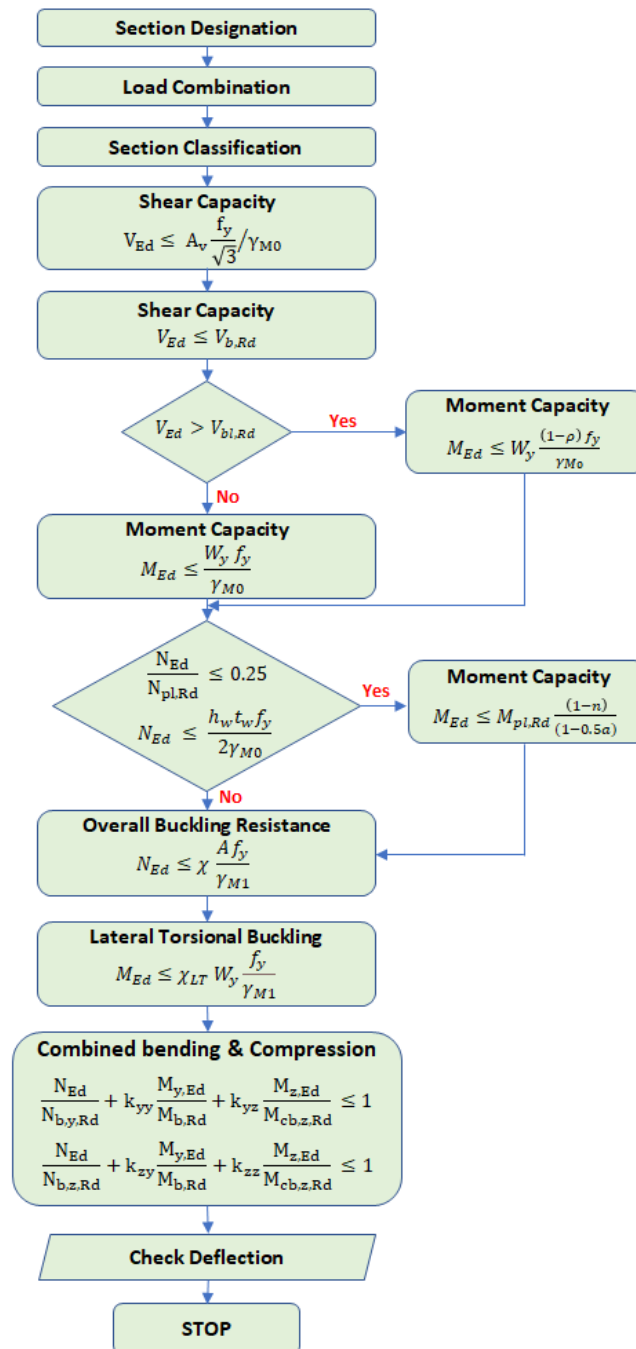


Figure 3-9: Design flowchart according to Eurocode-3

3.4 Summary

Structural analysis parameters and a set of design limitations must be specified in order to conduct the optimisation process. Firstly, the classical direct stiffness method for two- and three-dimensional structures was described, where a computer program is coded in MATLAB, which has the feature of analysis of 2D, and 3D structures was illustrated. In structural

engineering, the set of constraints includes the limitations imposed by code of practice. Since this study aims to use EC3 as a code of practice, all the required limitations for the design have been presented and discussed thoroughly.

CHAPTER FOUR. OPTIMISATION PROCEDURE

4.1 Introduction

Structural engineers are striving to design safe, economical, and environmentally friendly structures. Using the limitations provided as a set of design guidelines in the codes of practice aids in achieving a safe design. Formulating the design problem and solving it using one of the optimisation approaches is a systematic way of obtaining economical, low-embodied energy and less carbon emissions. It is practically impossible to formulate a design problem and solve it using intuition and experience because of the high degree of redundancy in steel structures. Therefore, tackling design challenges requires adopting the application of optimisation approaches to achieve the desirable goal.

This chapter addresses the fundamentals aspects and highlights the key characteristics of structural optimisation. This research has adopted three meta-heuristic approaches for optimising two and three-dimensional steel structures. For this study, optimization techniques known as Particle Swarm Optimisation (PSO), Genetic Algorithm (GA), and Harmony Search Algorithm (HSA) are selected for optimisation.

These optimisation approaches have gained widespread acceptance due to their ability to handle non-linear programming challenges. Their application increases the likelihood of rapidly attaining the global optimum via a random selection of variables that are often close to the optimum (Dogan, 2010; Messac, 2015). The performance of these methods in dealing with single and multi-objective problems will be further investigated. The GA, PSO, and HSA and all the optimisation requirements for EC3 code will be explained in detail, and their characteristics will be addressed. Since the traditional optimisation techniques are slow in operation, certain critical modifications were performed to improve the algorithm and speed up their operation. For implementing these optimisation methods, MATLAB is utilised as an optimisation tool.

Furthermore, the final section of this chapter has addressed the Life Cycle Assessment optimisation formula, considering all the aspects during the construction stage. The aspects of LCA are embodied energy, embodied carbon, weight, and cost of the structures. Each of these

aspects will be explained in detail, covering both material manufacturing, transportation, fire protection, corrosion protection, and erection stages.

4.2 General

The structural optimisation section is a core characteristic of the used algorithms. GA, PSO and HSA were developed and embedded into MATLAB program, integrated with the structural analysis and conventional design that are performed in other files of the program. The developed program is a combination of all processes such as the material properties, joint coordinates, members connectivity, load assignments, analysis, and constraint checks. Initially, all the genetic parameters required for the algorithms are input in a MATLAB Command Window. The involved genetic parameters are number of genes, size of population, maximum number of iterations, selection of probability coefficient, crossover and mutation probabilities, Elite count, crossover function, and the inflation rate. The self-coded program has some optional features that relate to the nature of any structural optimisation problem. The users can choose either multi-storey space structure, or portal frame structure. The program was created to deal with problems that contain both single-objective function and multi-objective function. The user is given the decision to whether deal with single-objective problem of weight minimisation or embodied energy minimisation, or multi-objective problem by combining them both in one function, where the calculation of cost and embodied carbon of the optimal options was considered.

After all data has been entered, the design optimisation process begins and usually lasts a few minutes, this depends on the size of the chosen structure and the number of design variables. Usually, few trials were set for each problem to identify the efficient genetic parameters. Following the design optimisation, the optimal solution is printed in a text file and displayed on the screen, where the whole process for genetic operations, analysis, and design constraints checks are included in the file. In addition to the results produces, Weight, embodied energy, embodied carbon, cost, and other relevant results are saved in distinct folders and exported into numerical spreadsheet for statical analysis.

The results of the structural analysis part of the self-coded program, has been compared and validated by a finite element analysis software called PROKON which is available on the university Software-Hub. Therefore, it is validated that the structural analysis matches the

accredited software, and this was presented in Chapter-5. After ensuring the analysis and design processes are carried out correctly, the developed algorithms are incorporated into the self-coded program in MATLAB. The algorithms used were then evaluated and validated by comparing typical examples of different optimisation problems with the results obtained in the literature. The obtained results showed good performance of the algorithms used. It is hoped that the approaches developed would aid in the push to integrate design optimisation into structural engineers' day-to-day work.

Finally, Comparisons of the algorithms for both single and multi-objective optimisation were carried out, to evaluate the performance of the developed meta-heuristic approaches in single and multi-objective function problems. Two criteria of comparison were set; the required time for finding the best solution and the accuracy of the results.

4.3 Optimisation Model

To develop an optimisation model for the any framed structure, all the equations for structural analysis, design, and optimisation were formulated and incorporated in the MATLAB environment. The entire code was split into distinct files based on their functionality. The automated workflow shown in Figure 4-1 summarises the process of developing the authors' written program for integrating the structural analysis, design, and optimisation stages of steel frames. For more user-friendly data input process and prevention of code tampering during data input process, all the input data was modelled in excel, which was then called out by the MATLAB code. The input parameters are stated previously. Examples of editable inputs are the joint coordinates, steel grades, the joint coordinates, beginning and terminating nodes for member connectivity, node numbers for member fixity, and member load (point load, distributed load, nodal load).

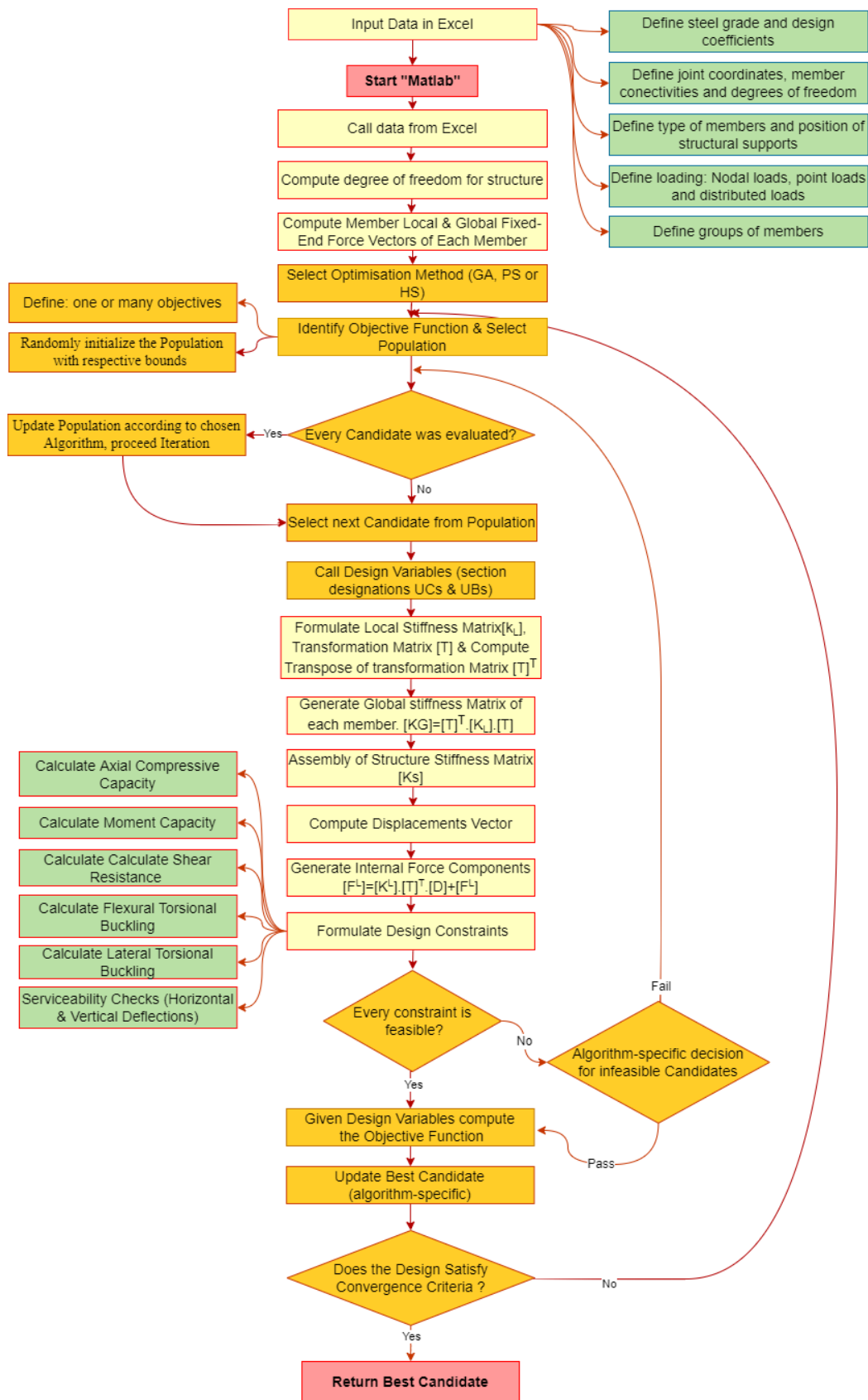


Figure 4-1: Automated workflow for the process of formulating the written program developed in this study

4.4 Optimisation Techniques and Tools

GA, PSO and HSA methods, which are coded and modified in MATLAB to optimise the design of 2D and 3D steel structures for minimum weight and embodied energy. This research employed these approaches since they are extensively used and regarded as accurate and capable for tackling linear and nonlinear programming issues (Ravat et al., 2021). Using these approaches increases the possibility of finding a global or local optimum by selecting random starting points near the optimum (Alkhadashi et al., 2022).

Additionally, the efficacy of these approaches is partially dependent on the development of robust and competent MATLAB functions for producing the optimal design. As a result, a MATLAB optimisation code that is proven to work will need the following m-files.

1. Analysis: A comprehensive analysis procedure of the 2D and 3D structures were prepared in MATLAB Command Window, so that MATLAB can smoothly perform all the structural analysis calculations.
2. Design: A complete design check procedure was prepared so that MATLAB can automatically check the analysis results against the design limits.
3. Optimisation: The structural optimisation problem was defined in a MATLAB function so that the algorithms used can find the optimal design for the structure. This may be accomplished by encoding the necessary data in a MATLAB script. Consequently, prior to harvesting the optimal design, the following procedures should be coded in MATLAB:
 - A) Coding the single and multi-objective functions: it is the objective or target function in the self-coded program that controls the desired aim of this research (minimum weight or/and embodied energy).
 - B) Coding the Design Variables: a set of discrete variables that will be used by the optimisation approaches coded in MATLAB to achieve the desired objective function of this study.
 - C) Coding the design constraints: These limitations are based on the provisions of Eurocode 3.

- D) Coding and developing the solving methods: Three optimisation methods were coded in MATLAB which will be tested to solve a range of problems in 2D and 3D steel structures.

For ease of work and accuracy of the written program in MATLAB, the self-coded program was divided into different files, where the main file contains the GA, PSO, and HSA optimisation processes. The commands for the optimisation as well as those that summon the objective and constraints files are contained here. GA, PSO, and HSA are used. To implement this, commercially available computer programming, MATLAB-R2020a version and Microsoft Excel, are used in conjunction with one another to perform the optimisation and analysis. Both programs are also readily available and licenced to the university. The optimisation was carried out by the university desktop computers, conducting one optimisation run at a time. More information about the developed optimisation methods used in this research is shown in the following flowcharts, illustrating how each method works.

4.4.1 Genetic Algorithm (GA)

The execution of the developed GA command in the file makes use of the in-built command. This command requires the setting of the population size, the number of generations, the number of population members that survive to the next iteration intact, and the operator types and the required probabilities. The default selection operator in MATLAB for discrete optimisation is tournament selection. Therefore, trials were conducted for the other operators as well as the population and generation sizes. The values for each of these parameters will be different for each problem, so they will be set later in the study. Below is a mathematical explanation of the process as well as an illustration of it, given in Figure 4-2.

At the first iteration, the population (POP) members are initialized randomly, i.e. member

$$x \in R^{1 \times n}, xi \in u_{[lb_i ub_i]}^{1 \times n}, i \in 1 \dots n. \quad (4.1)$$

Where n is the number of decision variables, and (lb, ub) are lower and upper bound vectors.

Starting every iteration, the cost function is being evaluated for every (N_p) candidate in POP. The N_b best options are chosen to become parents and N_e elite candidates are granted surviving to the next generation.

Children x^* are generated by crossing over two of the parent's $\{x^a, x^b\}$ genes.

The N_c pairs are chosen randomly and so are the genes, using $r \in u_{[0,1]}^{1 \times n}$:

$$x_i^* = \begin{cases} x_i^a & \text{if } r_i < 0.5, \\ x_i^b & \text{otherwise} \end{cases} \quad (4.2)$$

Besides crossover, some1 of the children are born via mutation of one parent x^a . Here, a

Gaussian distribution is used $r \in N_{[0, \frac{1}{\sqrt{3}}]}^{1 \times n}$:

$$x_i^* = \begin{cases} x_i^a & \text{if } r_i < 0.5, \\ x_i^b + r_i \cdot (ub_i - lb_i) & \text{otherwise} \end{cases} \quad (4.3)$$

Finally, the bounds must be enforced:

$$x_i^* = \begin{cases} lb_i & \text{if } x_i^* < lb_i, \\ ub_i & \text{if } x_i^* > ub_i, \\ x_i^* & \text{otherwise} \end{cases} \quad (4.4)$$

New population is composed of elite, crossover and mutation members and algorithm proceeds to the next iteration.

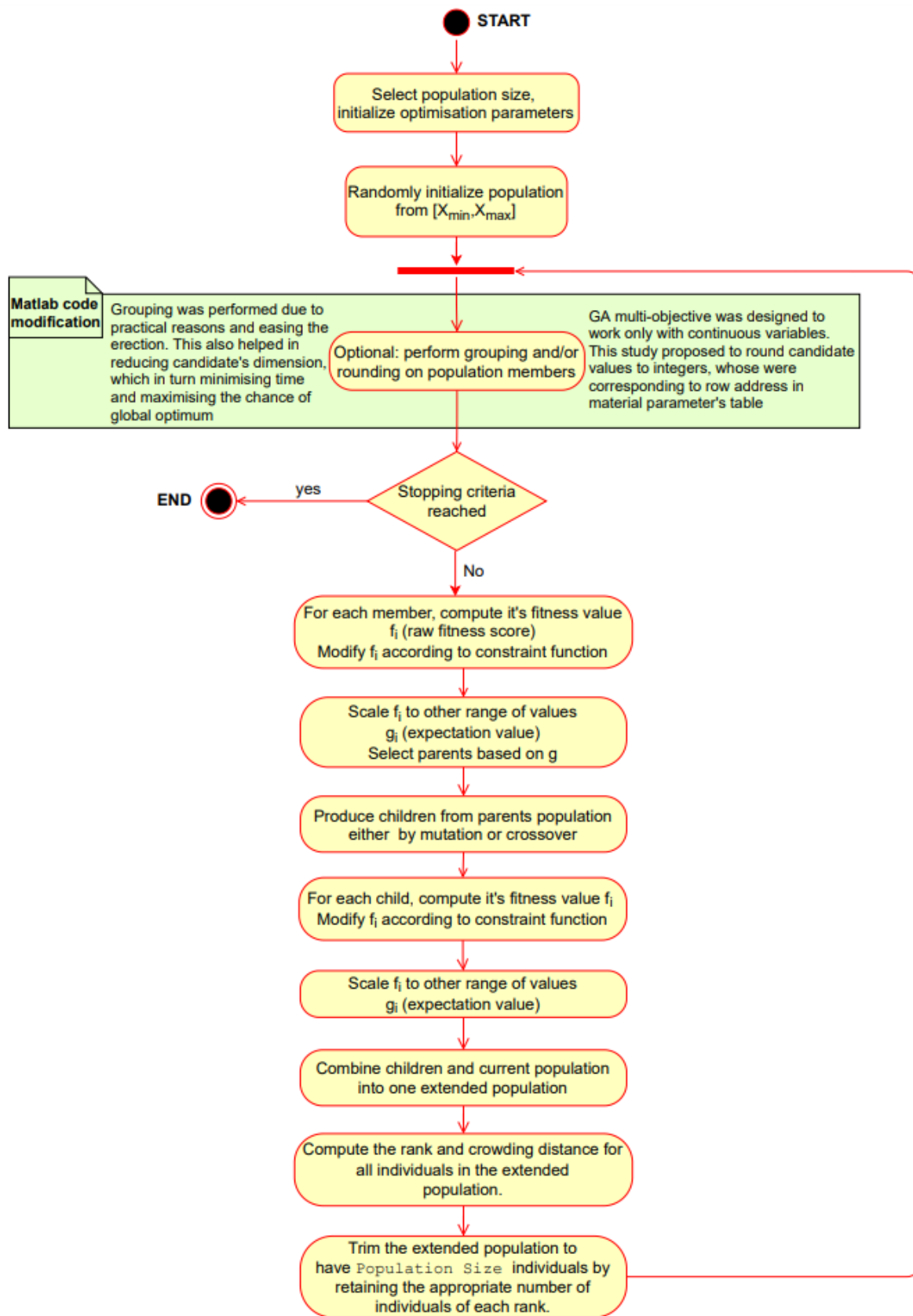


Figure 4-2: Schematic description for GA procedure

4.4.2 Particle Swarm Optimisation (PSO)

The PSO command also requires the setting of certain operators like the EliteCoun, the swarm size, number of iterations, and the inertia weight. Due to PSO being more suited for continuous variable optimisation, a rounding off technique was used in the code for initially obtaining an optimum solution by assuming the design variables to be continuous. These variables are then rounded off to the nearest integer value using heuristic methods in order to obtain an optimum discrete solution. As such, this method is expected to give close discrete global optimum solutions. The parameters used for the PS process will also be described later for each problem. Below is a mathematical explanation of the process as well as an illustration of it, given in Figure 4-3.

Particle Swarm Optimisation (PSO) tackles the optimisation problems using swarm intelligence. Generally, particles travel through the search domain communicating with each other their own fitness. Neighbouring members then responds accordingly, moving towards the local minimum.

First, the population members are initialized randomly. The candidates are composed of value vector x and velocity vector v .

$$\{x, v\} \in R^{1 \times n}, \{x_i, v_i\} \in u_{[lb_i, ub_i]}, i \in 1 \dots n, \quad (4.5)$$

Where n is the number of decision variables, and lb, ub are the lower and upper bound vectors respectively.

With each iteration, the algorithm randomly chooses N_s among N_n neighbours¹ separately, for every candidate. Then the local best member x^b among N_s is chosen. Particle velocity v^* is then updated using random variables $r_1, r_2 \in u_{[0,1]}^{1 \times n}$ according to weighted sum:

$$v_i^* = W \cdot v_i^* + p_1 \cdot r_1 \cdot (x_i^{bm} - x_i^*) + p_2 \cdot r_2 \cdot (x_i^b - x_i^*) \quad (4.6)$$

Where W, p_1, p_2 are inertia, global and local constant respectively. x^{bm} is the best other particle x^* has seen and memorized. Afterwards, position is being updated and bounds enforced:

$$x_i^* = x_i^* + v_i^* \quad (4.7)$$

$$x_i^* = \begin{cases} lb_i & \text{if } x_i^* < lb_i, \\ ub_i & \text{if } x_i^* > ub_i, \\ x_i^* & \text{otherwise} \end{cases} \quad (4.8)$$

At this point, the cost function is evaluated and if the global best has been lowered, the following operations are carried out:

$$c = \max\{0, c - 1\}, N_n = N_{nmin} \quad (4.9)$$

$$W = \begin{cases} 2 \cdot W & \text{if } c < 2 \\ ub_i & \text{if } c > 5 \\ W & \text{otherwise} \end{cases} \quad (4.10)$$

$$W = \begin{cases} lb_W & \text{if } W < lb_W, \\ ub_W & \text{if } W > ub_W, \\ W & \text{otherwise} \end{cases} \quad (4.11)$$

N_{nmin} is the minimum neighbourhood size. However, if there has not been a global improvement, these operations are performed instead:

$$c = c + 1, N_n = \min\{N + N_{min}, N_{pop}\} \quad (4.12)$$

N_{pop} being total swarm size. After looping through whole population, the algorithm proceeds to next iteration.

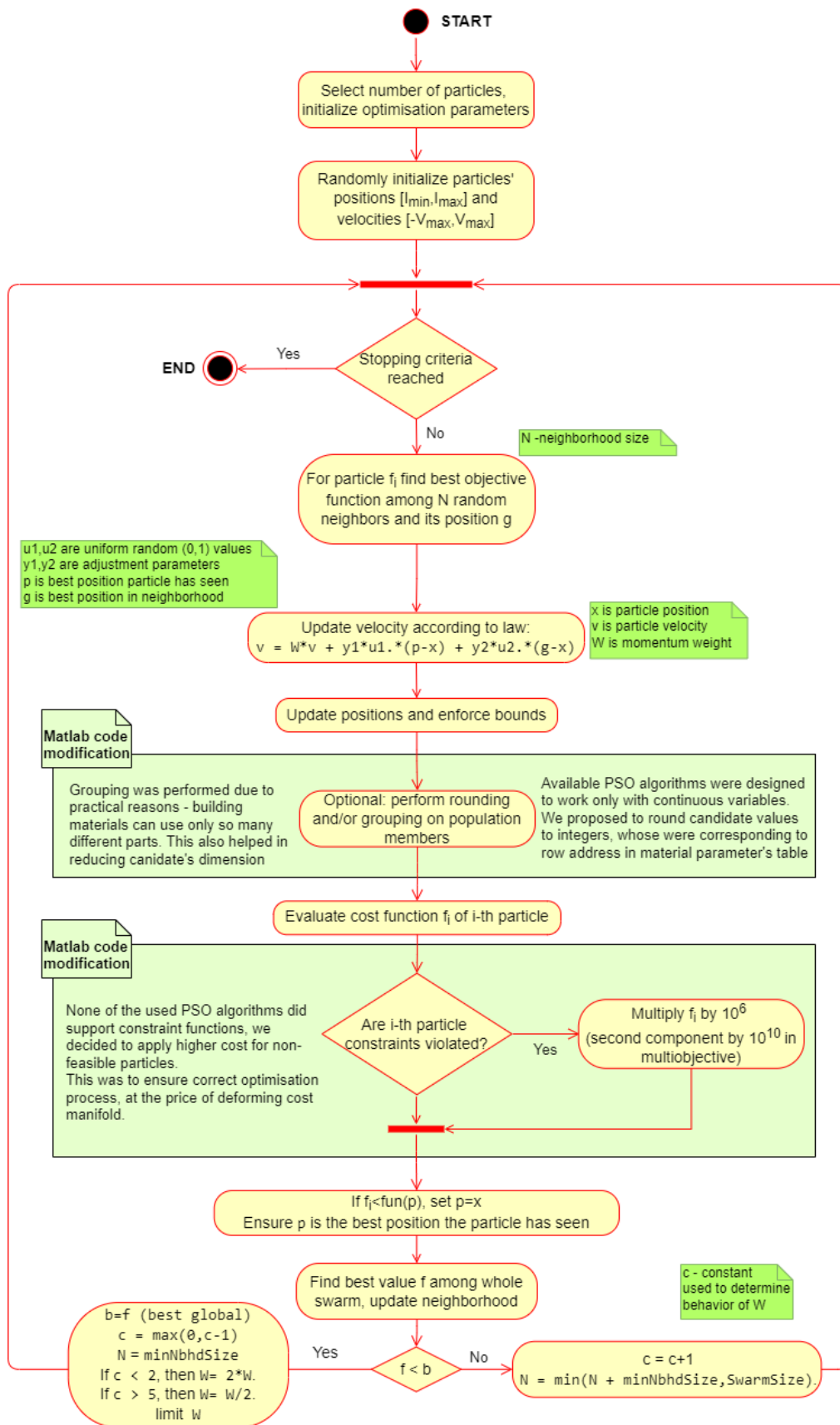


Figure 4-3: Schematic description for PSO procedure

4.4.3 Harmony Search Algorithm (HAS)

The HSA has the distinguishing features of algorithm simplicity and search efficiency. To set HAS command for single and multi-objective functions, certain operators have to be identified such as number of population size and maximum number of iterations. also, these commands require some specific parameters such as the percentage of elements in candidate vector that can be kept intact or modified (HMCR), percentage of elements chosen by HMCR that will be modified (PAR) and the percentage of decision range which is added during modification (BW). Below is a mathematical explanation of the process as well as an illustration of it, given in Figure 4-4.

Harmony Search (HS) is based on an idea, that the musicians (population) play a song in one "harmony". This results in taking into account the whole population at each iteration. First, the Harmony Memory (HM) is being initialized with random values

$$x \in R^{1 \times n}, lb_i < x_i < ub_i, \quad (4.13)$$

Where $i \in 1 \dots n$, n-number of decision variables.

Then, cost function is being evaluated for every candidate in HM. The one with worst fitness (x_w) is chosen to be replaced.

New candidate is created by the following steps:

- Take n random candidates $c^{(1..n)}$ from whole HM, the new base x^* is built by taking one "harmonic" from every c i.e.:

$$x_i^* = c_i^i \quad (4.14)$$

- The bitmask $CM \in Z^{1 \times n}$ is generated by comparison of random value $r \in u_{[0,1]}^{1 \times n}$ with parameter HMCR, such:

$$CM_i = \begin{cases} 1 & \text{if } r_i < HMCR, \\ 0 & \text{otherwise} \end{cases} \quad (4.15)$$

- Bitmask NH is defined as binary complement of CM
- Bitmask PA is defined using parameter PAR and mask CM (vector r is generated anew):

$$PA_i = \begin{cases} 1 & \text{if } r_i < PAR \text{ and } CM_i = 1, \\ 0 & \text{otherwise} \end{cases} \quad (4.16)$$

- Then mask CM is updated:

$$CM_i = \begin{cases} CM_i & \text{if } PA_i = 0, \\ 0 & \text{otherwise} \end{cases} \quad (4.17)$$

Finally, all of the above masks are composed. Parameter bw is introduced as a "generation variance", (new random vectors r^1 , r^2 are introduced):

$$x_i^* = CM_i \cdot x_i^* + PA_i \cdot x_i^* + bw \cdot r_i^1 + NH_i \cdot r_i^2 \quad (4.18)$$

$$r_i^1 = u_{[-1,1]} \quad (4.19)$$

$$r_i^2 = u_{[lb_i, ub_i]} \quad (4.20)$$

In regard to bound enforcement, the candidate is corrected afterwards

$$x_i^* = \begin{cases} c_i^i & \text{if } x_i^* < lb_i, \text{ or } x_i^* > ub_i, \\ x_i^* & \text{otherwise} \end{cases} \quad (4.21)$$

The worst candidate can be now replaced $x_w = x^*$, its fitness function evaluated, and one can proceed to next iteration

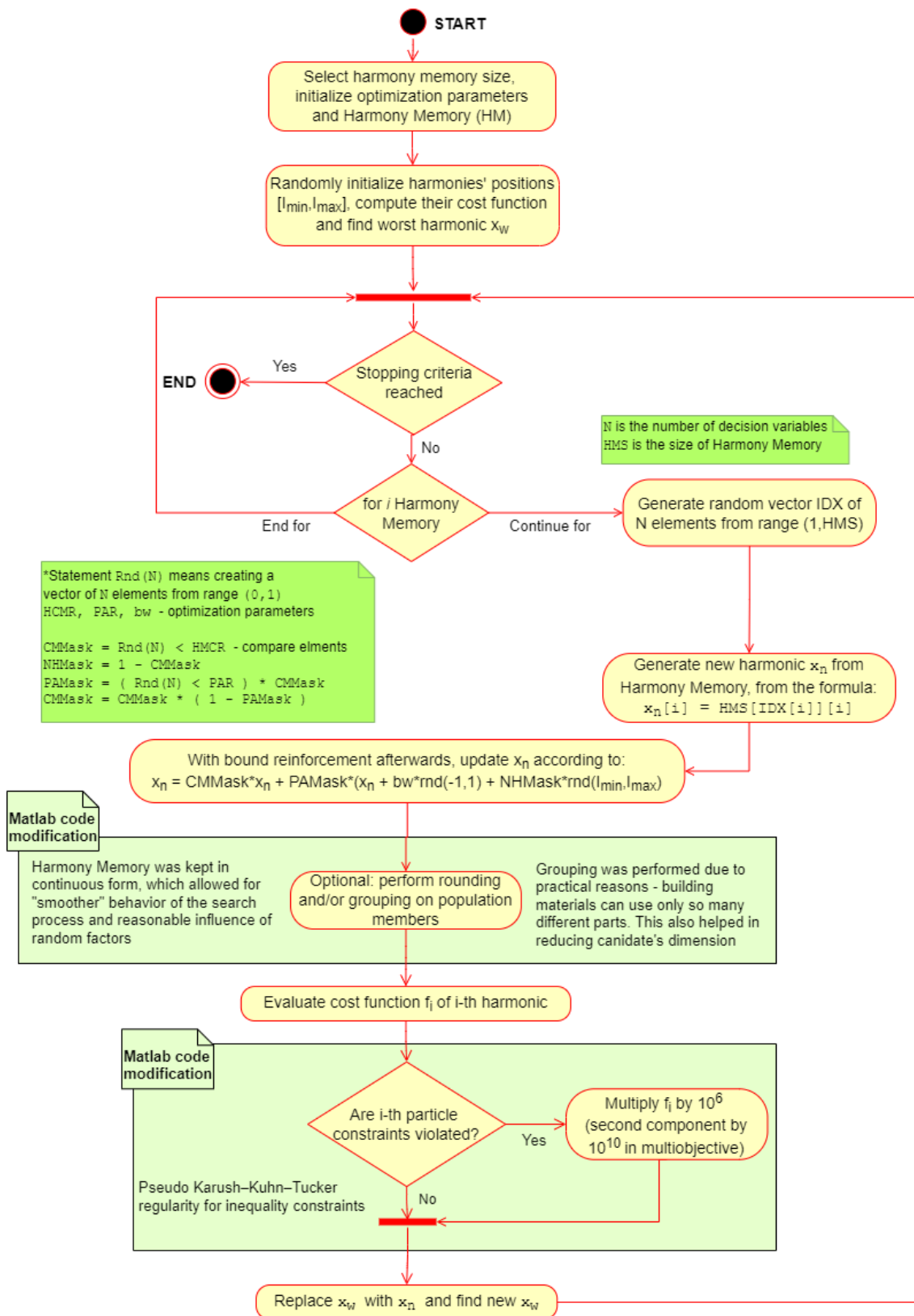


Figure 4-4: Schematic description for HSA procedure

4.5 Optimum Design Formulation

To build a reliable design formula, all the optimisation formulas should be generated. As a result, this section will identify all the requirements to develop a trustworthy formula, starting by figuring out all the components of the objective functions and ending with the identification of the constraints applied in this research. It will also explain each of these parts in detail, with enough information to help develop good understanding of the developed formulas.

There are some design coefficients that are constant throughout the process. These fixed parameters include the constants for the objective functions such as embodied energy and carbon factors. The values of the constants for embodied carbon, embodied energy, and cost are given in Table 4.1.

Table 4.1: Parameters used for the objective function.

Type	Symbol	Unit	Coefficient	Reference
Cost factor of steel S355	Kst_355	£/tonne	960	(Burgan et al., 2012)
Cost factor of steel S275	Kst_275	£/tonne	900	(Burgan et al., 2012)
Cost factor of transportation	Ktr	£/tonne	43.3	(Burgan et al., 2012)
Cost of corrosion protection	Cco	£/m ²	8.90a	(SPONS, 2019)
Cost factor for fire protection	Cfi	£/m ²	16	(SPONS, 2019)
Man-hours required for fire protection	mh	£/h	18.3	(SPONS, 2019)
Cost factor of erection	Ker	£/h	243	(SPONS, 2019)
Total time for lifting and installation	Ter	h/kg	0.0014	(Pavlovčič et al., 2004)
Embodied energy of steel	Est	MJ/kg	20.1	(Zhou and Azar, 2018; ICE, 2017)
Embodied carbon of steel	Eest	kgCo _{2e} /kg	1.55	(Jones and Hammond, 2019)
Embodied energy consumed by paint	ε _{pt}	MJ/m ²	21b	(Hammond et al., 2011)
Embodied carbon consumed by paint	ε _{ept}	kgCo _{2e} /m ²	0.87b	(Jones and Hammond, 2019)
Embodied energy factor for transportation	ε _{tr}	MJ/tonne.km	0.97	(Hong et al., 2014)
Embodied carbon factor for transportation	ε _{etr}	kgCo _{2e} /tonne.km	0.0715	(Hong et al., 2014)
Energy conversion factor for the equipment	ε _{eq}	MJ/L	38.3c	(Victoria, 2016)
Emission conversion factor for the equipment	ε _{eeq}	kgCo _{2e} /L	2.914	(Victoria, 2016)
Fuel density	KPL	kg/L	0.84	(Zhou and Azar, 2018)
Load factor in percentage	LF	%	0.54	(Zhou and Azar, 2018)
Gross horsepower of equipment	GHP	hp	163.6	(Zhou, 2018)
Amount of fuel consumed per brake	K	per brake hp/h	0.17	(Zhou and Azar, 2018)
Note / a including blast cleaning, b double coat paint, c assuming diesel engine				

4.5.1 Objective Function File

Single and multi-objective functions are considered in this research, where they can be defined as the minimum weight and embodied energy of the structure as well as the calculation of cost and embodied energy for the optimal options. In single objective function, weight and embodied energy were considered separately to investigate the effect of each objective on the optimal results and examine the performance of the applied optimisation methods in single objective problems. Whereas, in multi-objective function, weight and embodied energy functions were combined to investigate the optimal results and examine the performance of the used optimisation methods in multi-objective problems. Thus, the optimisation methods are stated in terms of weight, embodied energy, embodied carbon, and cost, which can be formulated as follows:

4.5.1.1 Single Objectives

1- Weight Function

The total weight of the structure depends on the cross-sectional area and length of an element, where the structural elements were divided into two groups, beams, and columns. The weight directly relates to the cost of the structure and as such a reduction in it would lead to reduction in the cost. The resulting objective function for the weight minimisation is given by:

$$W = \rho(\sum_{i=1}^{N_{sb}} A_i(x) \sum_{b=1}^{N_b} l_{ib} + \sum_{j=1}^{N_{sc}} A_j(x) \sum_{c=1}^{N_c} l_{jc}), \quad (4.22)$$

Where:

W is the weight of the structure

ρ is the density of the sections

N_{sb} is the total number of beam groups

N_{sc} is the total number of column groups

N_b is the total number of beams

N_c is the total number of columns

A_i is the area of beam section of each group (m^2)

l_{ib} is the length of beam element of each group (m)

A_j is the area of column section of each group (m²)

l_{jc} is the length of column element of each group (m)

2- Embodied Energy Function

The second objective function studied is the minimization of embodied energy. In this research, the embodied energy of the construction stage was evaluated, which consists of the material production, transportation, and erection. Therefore, the total embodied energy can be formulated as follows:

$$EE_T = EE_m + EE_t + EE_e \quad (4.23)$$

Where:

EE_T is the total embodied energy consumed (MJ)

EE_m is the embodied energy of the material at the manufacturing stage (MJ)

EE_t is the embodied energy of transportation (MJ)

EE_e is the embodied energy of erection (MJ)

The following is a breakdown of each of the three components:

A) Materials function

The embodied energy of a material is a function of the mass of the frame and the surface area to be covered by the paint. The embodied factor of steel and paint may differ from database to database depending on what was considered in the calculation. It is expected to include the energy required for raw material extraction, transportation of the material between the extraction site and the manufacturing facility, and the manufacturing process itself.

$$EE_m = EE_1 + EE_2 \quad (4.24)$$

EE_1 is the embodied energy formula of steel elements during production (MJ)

$$EE_1 = \left(\sum \varepsilon_{st} (1 + Wf) \rho \left(\sum_{i=1}^{Nsb} A_i(x) \sum_{b=1}^{Nb} l_{ib} + \sum_{j=1}^{Nsc} A_j(x) \sum_{c=1}^{Nc} l_{jc} \right) \right) \quad (4.25)$$

EE_2 is the embodied energy formula of paint coat (MJ)

$$EE_2 = \left(\varepsilon_{pt} \left(\sum_{i=1}^{Nsb} a_i(x) \sum_{b=1}^{Nb} l_{ib} + \sum_{j=1}^{Nsc} a_j(x) \sum_{c=1}^{Nc} l_{jc} \right) \right) \quad (4.26)$$

Where:

Wf is the percentage of waste, which considered as (5%).

ε_{st} is the embodied energy of steel in (MJ/tonne), given in table 4-1.

ε_{pt} is the embodied energy consumed during steel painting in (MJ/m²), given in table 1.

B) Transportation function

The second component is the transportation function, where the embodied energy here is the energy used by vehicles for transportation. Usually, a more comprehensive calculation should include other things being transported to the site other than material, e.g., labour and equipment. However, that would make the calculation more tedious as this differs from site to site. Distance travelled is essential here as well, as it can give an estimation of how much energy is consumed by the vehicle. That is, a shorter distance would mean lower energy consumption and vice versa. Lastly, the amount of materials transported will suggest the type of vehicle used or how many trips are required for the vehicle. As a result, the transportation function here is also a function of weight.

The function's assumptions include the usage of a fuel-consuming vehicle, the vehicle's full load and return journey, and allowance for material waste. As a result, it is stated as follows (Zhou and Azar, 2018):

$$EE_t = 1.66 \varepsilon_{tr} D (1 + Wf) \rho \left(\sum_{i=1}^{Nsb} A_i(x) \sum_{b=1}^{Nb} l_{ib} + \sum_{j=1}^{Nsc} A_j(x) \sum_{c=1}^{Nc} l_{jc} \right) \quad (4.27)$$

Where:

ε_{tr} is the embodied energy factor for transportation in (MJ/ tonne.km)

D is the distance travelled in km, which was considered as 100 km

C) Erection function

The erection process requires the use of machinery and labour to assemble the components. The energy consumption considered is usually by that of the equipment. For example, use of crane for hoisting of elements. However, each equipment can differ in the form of energy used, e.g., electricity, fuel, steam etc. for its operation. Each form of energy thus has different impact and as such, the amount used by the equipment is used for quantifying the energy consumption. The total time used for the erection also plays a major role in the calculation as that would quantify how long the equipment was working for. To estimate the time used, the amount of quantity is also included in the function below. Additionally, one mobile crane powered by diesel fuel was anticipated for the erection. The crane's properties are shown in table 1. According to (Brütting et al., 2020; Mao et al., 2013; Zhou and Azar, 2018b) the function is as follows:

$$EE_e = T_{er} ECF \rho \left(\sum_{i=1}^{Nsb} A_i(x) \sum_{b=1}^{N_b} l_{ib} + \sum_{j=1}^{Nsc} A_j(x) \sum_{c=1}^{N_c} l_{jc} \right) \quad (4.28)$$

$$ECF = \varepsilon_{eq} \times LMPH \quad (4.29)$$

$$LMPH = \frac{K \times GHP \times LF}{KPL} \quad (4.30)$$

Where:

T_{er} is the total time required for lifting and installation (hour/Kg), given in Table 4.1

ECF is a factor for energy consumption, given in Table 4.1

ε_{eq} is an energy conversion factor for the equipment, given in Table 4.1

LMPH is the fuel used for the equipment (Liter/hour).

KPL is the fuel density (Kg/litre), given in Table 4.1

GHP is the gross horsepower of the equipment's engine (hp), given in Table 4.1

LF is the load factor for diesel engines, given in Table 4.1

K is the amount of fuel used per brake (hp/hour), given in Table 4.1

3- Cost formula

As previously stated in the literature review, the cost of the frame is divided into four sections: material; transportation; erection; and finishing. The general cost formula is given as follows:

$$C_T = C_m + C_t + C_e + C_f \quad (4.31)$$

Where;

C_T is the total cost of the structure (£)

C_m is the material cost (£)

C_t is the transportation cost (£)

C_e is the erection cost (£)

C_f is the cost of finishing (£)

The following is a breakdown of each of these sections:

A) Materials

$$C_m = k_{st}\rho \left(\sum_{i=1}^{Nsb} A_i(x) \sum_{b=1}^{N_b} l_{ib} + \sum_{j=1}^{Nsc} A_j(x) \sum_{c=1}^{N_c} l_{jc} \right) \quad (4.32)$$

Where:

k_{st} is the cost factor of steel (£/tonne), given in Table 4.1

B) Transportation

$$C_t = k_{tr}\rho \left(\sum_{i=1}^{Nsb} A_i(x) \sum_{b=1}^{N_b} l_{ib} + \sum_{j=1}^{Nsc} A_j(x) \sum_{c=1}^{N_c} l_{jc} \right) \quad (4.33)$$

Where:

k_{tr} is the cost factor of transportation (£/tonne), given in Table 4.1

C) Erection

The cost of erecting a structure using one mobile crane and four workers is calculated as follows:

$$C_e = T_{er}k_{er}\rho \left(\sum_{i=1}^{Nsb} A_i(x) \sum_{b=1}^{N_b} l_{ib} + \sum_{j=1}^{Nsc} A_j(x) \sum_{c=1}^{N_c} l_{jc} \right) \quad (4.34)$$

T_{er} is the total time required for lifting and installation (hour/Kg), given in Table 4.1

k_{er} is the cost factor of erection (£/h), given in Table 4.1

D) Finishing

$$C_f = \sum C_{fi} + C_{co} \quad (4.35)$$

C_{fi} is the cost of fire protection (£), given in Table 4.1

C_{co} is the cost of corrosion protection (£), given in Table 4.1

- **Fire protection**

$$C_{fi} = C_{fb} + C_{fc} \quad (4.36)$$

C_{fb} is the cost of fire protection for beam elements (£)

C_{fc} is the cost of fire protection for column elements (£)

This study considered the use of intumescent paint to give a fire resistance of 60 minutes and thus:

$$C_{fb} = \sum_{i=1}^{N_{sb}} \sum_{b=1}^{N_b} \left((1.43h_{ib}(z) + 0.7)k_{fi} + (0.27h_{ib}(z) + 0.6)mh \right) \quad (4.37)$$

$$C_{fc} = \sum_{j=1}^{N_{sc}} \sum_{c=1}^{N_c} \left((1.72h_{jc}(z) + 0.53)k_{fi} + (0.11h_{jc}(z) + 0.72)mh \right) \quad (4.38)$$

Where:

k_{fi} is the cost factor for fire protection (£/m²), given in Table 4.1

mh is the man-hours required for fire protection (£/h), given in Table 4.1

h_{ib} is the height of beam elements in group i (m)

h_{jc} is the height of column elements in group j (m)

- **Corrosion**

$$C_c = k_{pt} \left(\sum_{i=1}^{N_{sb}} a_i(x) \sum_{b=1}^{N_b} l_{ib} + \sum_{j=1}^{N_{sc}} a_j(x) \sum_{c=1}^{N_c} l_{jc} \right) \quad (4.39)$$

Where:

- k_{pt} is the cost factor of paint coat and blast cleaning (£/m²), given in table 4-1
- a_i is the surface area per meter length of the beams (m²)
- a_j is the surface area per meter length of the columns (m²)

4- Embodied Carbon

Along with embodied energy, embodied carbon was evaluated in terms of GHG emissions. Therefore, since the same stages were considered for the evaluation of both embodied energy and embodied carbon, similar equations have been used, except the emission coefficients are different. A summary of the equations is given as:

$$EC_T = EC_m + EC_t + EC_e \quad (4.40)$$

Where:

- EC_T is the total embodied carbon, starting by manufacturing to erection (KgCO₂e)
- EC_m is the carbon resulting from the material manufacturing stage (KgCO₂e)
- EC_t is the carbon emitted during material transportation stage (KgCO₂e)
- EC_e is the carbon emitted during the erection stage of the structure (KgCO₂e)

A) Materials

$$EC_m = EC_1 + EC_2 \quad (4.41)$$

$$EC_1 = \left(\sum \epsilon e_{st} (1 + Wf) \rho \left(\sum_{i=1}^{Nsb} A_i(x) \sum_{b=1}^{N_b} l_{ib} + \sum_{j=1}^{Nsc} A_j(x) \sum_{c=1}^{N_c} l_{jc} \right) \right) \quad (4.42)$$

$$EC_2 = \left(\epsilon e_{pt} \left(\sum_{i=1}^{Nsb} a_i(x) \sum_{b=1}^{N_b} l_{ib} + \sum_{j=1}^{Nsc} a_j(x) \sum_{c=1}^{N_c} l_{jc} \right) \right) \quad (4.43)$$

Where:

- EC_1 is the embodied carbon of steel elements (KgCO₂e)
- EC_2 is the embodied carbon of paint coat (KgCO₂e)
- ϵe_{st} is the embodied carbon coefficient of steel (KgCO₂e/tonne)
- ϵe_{pt} is the embodied carbon coefficient of painting (KgCO₂e/m²)

B) Transportation

$$EC_t = 1.66 \sum \varepsilon e_{tr} D (1 + Wf) \rho \left(\sum_{i=1}^{Nsb} A_i(x) \sum_{b=1}^{N_b} l_{ib} + \sum_{j=1}^{Nsc} A_j(x) \sum_{c=1}^{N_c} l_{jc} \right) \quad (4.44)$$

Where:

εe_{tr} is the embodied carbon coefficient for transportation (KgCO₂e/tonne.km)

C) Erection

$$EC_e = \sum T_l ECF \rho \left(\sum_{i=1}^{Nsb} A_i(x) \sum_{b=1}^{N_b} l_{ib} + \sum_{j=1}^{Nsc} A_j(x) \sum_{c=1}^{N_c} l_{jc} \right) \quad (4.45)$$

Where:

$$ECF = \varepsilon e_{eq} \times LMPH \quad (4.46)$$

εe_{eq} is the emission conversion factor of the equipment with diesel engine (KgCO₂e/L)

4.5.1.2 Multi-Objective Function

This function combines the objective function of EE and weight and thus combines their variables. That is;

$$MULTI = EE_T + W \quad (4.47)$$

Where MULTI represents multi objective function.

4.5.2 Design Constraints

Various checks for steel frame design specified in Eurocode-3 were used as the constraints for the optimisation problem of this study. The brief description of these checks is described below.

4.5.2.1 Ultimate Limit state (ULS)

The ultimate limit state refers to the occurrence that can lead to partial or total collapse of the building. The limit states considered were Compression, Biaxial bending, Shear force, Bending and shear, and Lateral and torsional buckling. The expression for each of them are:

Compression

$$g_1 = \frac{N_{Ed}}{N_{c,Rd}} \leq 1.0 \quad (4.48)$$

Where:

N_{Ed} is the design compressive force

$N_{c,Rd}$ is the design compressive resistance of the steel cross-section

Bending

$$g_2 = \frac{M_{Ed}}{M_{c,Rd}} \leq 1.0 \quad (4.49)$$

Where:

M_{Ed} is the design bending moment

$M_{c,Rd}$ is the design bending resistance of the steel cross-section

Shear

$$g_3 = \frac{V_{Ed}}{V_{c,Rd}} \leq 1.0 \quad (4.50)$$

Where:

V_{Ed} is the design shear force

$V_{c,Rd}$ is the design shear resistance

Flexural buckling:

$$g_4 = \frac{N_{Ed}}{N_{b,y,Rd}} \leq 1.0 \quad (4.51)$$

$$g_5 = \frac{N_{Ed}}{N_{b,z,Rd}} \leq 1.0 \quad (4.52)$$

Where:

N_{Ed} is the design compressive force

$N_{b,y,Rd}$ is the design buckling resistance about y-y axis

$N_{b,z,Rd}$ is the design buckling resistance about Z-Z axis

Combined bending and axial compression buckling

$$g_6 = \frac{N_{Ed}}{N_{b,y,Rd}} + k_{yy} \frac{M_{y,Ed}}{M_{b,Rd}} + k_{yz} \frac{M_{z,Ed}}{M_{cb,z,Rd}} \leq 1 \quad (4.53)$$

$$g_7 = \frac{N_{Ed}}{N_{b,z,Rd}} + k_{zy} \frac{M_{y,Ed}}{M_{b,Rd}} + k_{zz} \frac{M_{z,Ed}}{M_{cb,z,Rd}} \leq 1 \quad (4.54)$$

Where:

$M_{y,Ed}, M_{z,Ed}$ are the design values for the maximum moments about the member's y-y and z-z axes, respectively.

$M_{cb,z,Rd}$ is the design bending resistance in the minor axis (z-z)

$k_{zy}k_{zy}k_{zy}k_{zy}$ are the interaction factors that are reliant on instability and plasticity.

$M_{b,Rd}$ is the design buckling resistance moment

4.5.2.2 Serviceability limit state

Serviceability limit checks will ensure that the structure remains functionally in use throughout its design life with minimal maintenance. A deflection check is the only one considered here. The check is carried out using un-factored imposed actions.

$$g_8 = \delta_v < \frac{L}{360} \quad (4.55)$$

$$g_9 = \delta_h < \frac{H}{300} \quad (4.56)$$

Where

L is the beam span

H column height

δ_v vertical deflection

δ_h horizontal deflection

4.6 Summery

The key characteristics of structural optimisation were highlighted. The concepts of optimisation components such as design variables, objective functions, and constraints were discussed. The procedure that shows how GA, PSO, and HSA work was explained. All aspects of modifications designed to improve the performance and quality of these methods were discussed. Moreover, all the requirements to develop trustworthy optimisation formulas representing the Life Cycle Assessment during the construction phase were comprehensively covered. This included all the components of the objective functions as well as the identification of the constraints applied in this research.

CHAPTER FIVE. Validation of the Program and the Methods Used

5.1 Introduction:

In this section, the developed algorithms are examined by comparing typical examples with the results obtained in published literature. In the selection of literature, different steel frameworks are included, as well as different methods of optimisation. Two two-dimensional frames and two three-dimensional structures were examined. The two-dimensional frames were investigated using a six-storey frame and a ten-storey frame with different grouping. Five storey and ten storey frames were chosen for the three-dimensional steel frames. Also included in the investigation is the effect of grouping on the weight savings as well as the suitability of the algorithms for large structure. Since weight optimisation was common to the previous studies adopted; only weight is considered as the objective function in this chapter too.

Furthermore, in order to ensure the structural analysis process is carried out correctly, the results of the self-coded program have been compared and validated for the structural analysis part by a finite element analysis software called PROKON 3. 1.08, which is available at Nottingham Trent University Software-Hub. Only one simple two-dimensional frame was presented in this chapter, where a large three-dimensional structure was also validated and presented in Appendix-A.

5.2 Structural Analysis validation

The frame represented in Figure 5-1 was analysed and validated. It is a three bay two storey rigid frame. For a comprehensive validation, the frame was subjected to different loads; Wind load, distributed load, point load and nodal load. From Figure 5-2, it can be observed that the internal forces attained using MATLAB matches the accredited software.

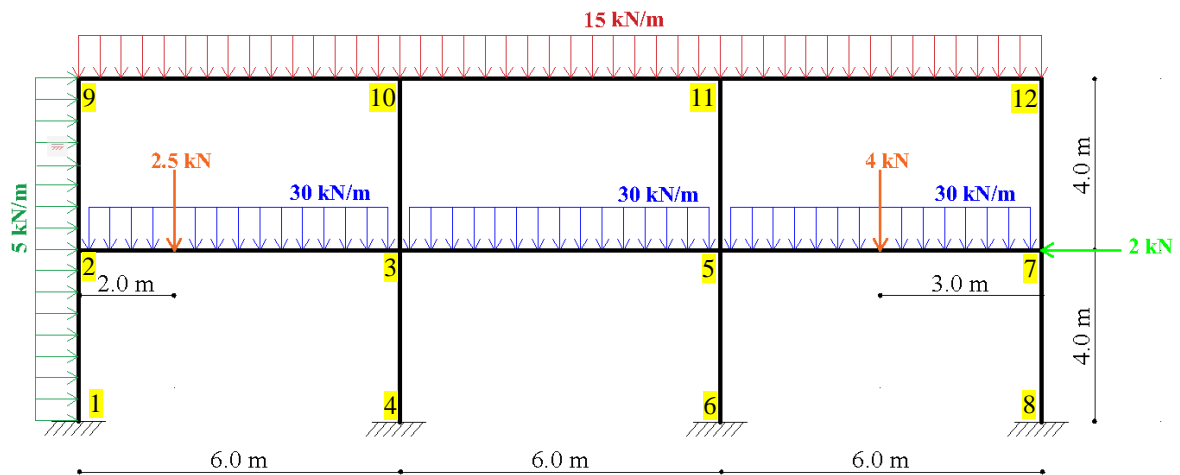


Figure 5-1: two-story, three bay rigid steel frame

PROKON results								MATLAB results				
BEAM ELEMENT END FORCES IN LOCAL ELEMENT AXES AT ULS								MATLAB Variable: UUU 21-Apr-2022				
Elem	Lcase	Axial kN	Y-Shear kN	X-Shear kN	Torsion kNm	M-yy kNm	M-xx kNm	MemberNo	JointNo	1 Axia_kN	2 Axia_kN_1	3 Axia_kN_2
1-	1	125.76	4.23	0.00	0.00	0.00	6.33	1	M-1'	125.7600	4.2300	6.3300
2-	1	-125.76	15.77	0.00	0.00	0.00	-29.39	2	M-2'	-125.7600	15.7700	-29.3900
3-	1	9.69	84.04	0.00	0.00	0.00	61.29	3	M-3'	9.6900	84.0400	61.2900
3-	1	-9.69	98.46	0.00	0.00	0.00	-108.30	4	M-4'	-9.6900	98.4600	-108.3000
4-	1	277.54	9.03	0.00	0.00	0.00	16.78	5	M-5'	277.5400	9.0300	16.7800
4-	1	-277.54	-9.03	0.00	0.00	0.00	19.34	6	M-6'	-277.5400	-9.0300	19.3400
5-	1	5.54	87.05	0.00	0.00	0.00	82.40	7	M-7'	5.5400	87.0500	82.4000
5-	1	-5.54	92.95	0.00	0.00	0.00	-100.09	8	M-8'	-5.5400	92.9500	-100.0900
6-	1	278.38	6.18	0.00	0.00	0.00	9.22	9	M-9'	278.3800	6.1800	9.2200
6-	1	-278.38	-6.18	0.00	0.00	0.00	15.51	10	M-10'	-278.3800	-6.1800	15.5100
7-	1	0.38	93.07	0.00	0.00	0.00	90.72	11	M-11'	0.3800	93.0700	90.7200
7-	1	-0.38	90.93	0.00	0.00	0.00	-84.28	12	M-12'	-0.3800	90.9300	-84.2800
8-	1	134.82	18.55	0.00	0.00	0.00	42.21	13	M-13'	134.8200	18.5500	42.2100
8-	1	-134.82	-18.55	0.00	0.00	0.00	32.00	14	M-14'	-134.8200	-18.5500	32.0000
9-	1	41.72	-6.08	0.00	0.00	0.00	-31.90	15	M-15'	41.7200	-6.0800	-31.9000
9-	1	-41.72	26.08	0.00	0.00	0.00	-32.40	16	M-16'	-41.7200	26.0800	-32.4000
10-	1	26.08	41.72	0.00	0.00	0.00	32.40	17	M-17'	26.0800	41.7200	32.4000
10-	1	-26.08	48.28	0.00	0.00	0.00	-52.07	18	M-18'	-26.0800	48.2800	-52.0700
11-	1	92.03	4.88	0.00	0.00	0.00	9.12	19	M-19'	92.0300	4.8800	9.1200
11-	1	-92.03	-4.88	0.00	0.00	0.00	10.41	20	M-20'	-92.0300	-4.8800	10.4100
12-	1	21.19	43.75	0.00	0.00	0.00	41.66	21	M-21'	21.1900	43.7500	41.6600
12-	1	-21.19	46.25	0.00	0.00	0.00	-49.17	22	M-22'	-21.1900	46.2500	-49.1700
13-	1	92.36	1.02	0.00	0.00	0.00	0.15	23	M-23'	92.3600	1.0200	0.1500
13-	1	-92.36	-1.02	0.00	0.00	0.00	3.94	24	M-24'	-92.3600	-1.0200	3.9400
14-	1	20.17	46.10	0.00	0.00	0.00	45.23	25	M-25'	20.1700	46.1000	45.2300
14-	1	-20.17	43.90	0.00	0.00	0.00	-38.61	26	M-26'	-20.1700	43.9000	-38.6100
15-	1	43.90	20.17	0.00	0.00	0.00	42.07	27	M-27'	43.9000	20.1700	42.0700
15-	1	-43.90	-20.17	0.00	0.00	0.00	38.61	28	M-28'	-43.9000	-20.1700	38.6100

Figure 5-2: Results attained from PROKON software and MATLAB

5.3 Validation of the Design Optimisation

5.3.1 Two-dimensional, six-story, two bay rigid steel frame

The frame represented in Figure 5-3 was investigated by (Saka, 2009) Saka and (Issa, 2010). It is a two bay six storey rigid frame grouped into six. The grouping was arranged in such a

way that all outer columns except the last floor make up 1 group, all inner columns except the last floor also made up another group, the floor beams form another group and finally the roof beams, uppermost inner and outer columns formed 3 separate groups. The beams of the frame have a factored gravity action of 50kN/m while the whole frame was subjected to a number of 25kN lateral concentrated actions at the top of each storey. For the two studies, the limiting constraints were based on BS5950 while this study used Euro code 2. The steel grade is S275 steel.

Both studies carried out weight minimisation, as such only the result for optimum weight is displayed here. (Saka, 2009) used GA and HSA for the optimisation while (Issa, 2010) used modified GA embedded in a bespoke program called DO-DGA. The parameters for the GA, PSO and HSA for this study are as follows:

- GA; 200 generation, 100 iteration, 100 elite counts, 0.8 mutation and 0.7 cross over probability.
- PSO: 100 iterations and 150 population size.
- HSA: 100 iterations, 200 harmony memory size, 0.3 for PAR, 0.3 for BW and 0.8 for HMCR.

Table 5.1 gives the result obtained in this study and those for the two previous studies. The table presents the sections by each algorithm as well as the resulting total weight and iterations. It is noticed that this study achieved a lighter frame with a difference of 7%. Even though Saka used HSA which has been identified as the most efficient in this study, the result obtained was still greater than all the results of the three algorithms used in this study.

It can be observed in Table 5.1 as well as Figure 5-4, that PSO converged first using the least iterations while GA used the highest. From Issa (2010), convergence occurred at 76 generations while it occurred at 84 generations for the GA used in this study. It suggests that his algorithm may be faster in convergence, but not as efficient as that in this study.

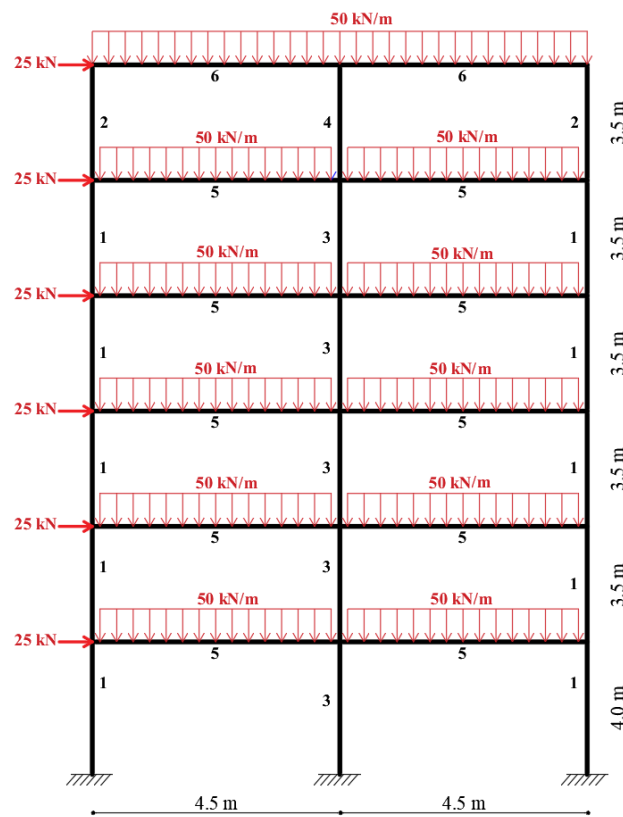


Figure 5-3: Six-story, two bay rigid steel frame

Table 5.1: The optimum solutions obtained by current study, (Isaa, 2010) and (Saka, 2007)

Element	Current study			Issa, 2010	Saka, 2009	
	GA	PSO	HSA	DO-DGA	HSA	GA
Column (1)	203 x 203 x 60	254 x 254 x 73	254 x 254 x 73	305 x 305 x 97	203 x 203 x 60	203 x 203 x 71
Column (2)	152 x 152 x 23	152 x 152 x 30	152 x 152 x 23	152 x 152 x 23	152 x 152 x 30	203 x 203 x 46
Column (3)	356 x 368 x 129	305 x 305 x 97	305 x 305 x 97	305 x 305 x 97	356 x 368 x 129	356 x 368 x 129
Column (4)	203 x 203 x 46	152 x 152 x 30	203 x 203 x 46	152 x 152 x 23	152 x 152 x 30	203 x 203 x 46
Beam (5)	457 x 152 x 60	457 x 152 x 60	457 x 152 x 60	457 x 152 x 52	457 x 191 x 67	457 x 152 x 52
Beam (6)	254 x 102 x 22	305 x 102 x 25	254 x 102 x 22	305 x 102 x 33	305 x 102 x 33	356 x 171 x 45
Total optimum weight of the structure (kN)	76.95	76.07	75.89	81.22	81.12	81.21
Number of iteration	100	100	100		-	-
Number of iteration for best result	84	39.00	61		-	-

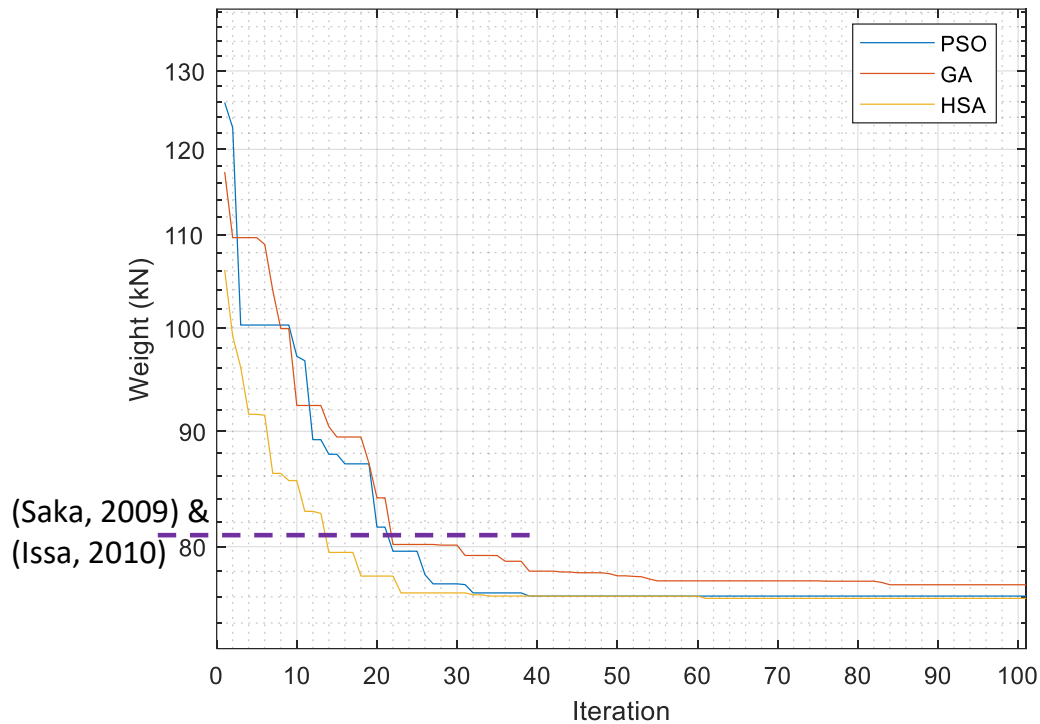


Figure 5-4: Search history for six-storey structure

The frame studied here was rearranged in order to investigate the effect of grouping on the savings possible. The first three stories have the side columns grouped together as group 1, the intermediate column as group 3 and the floor beams as group 5. The upper three stories have side columns as group 2; intermediate columns as group 4; and beams as group 6, shown in Figure 5-5.

The results obtained are presented in Table 5.2 and Figure 5-6. The savings observed were up to 16%. The number of iterations required for the optimisation also reduced especially for HSA (from 61 to 32) and GA (from 84 to 57). From figure 4, HSA and PSO converged at the same number of iterations while GA required more iterations.

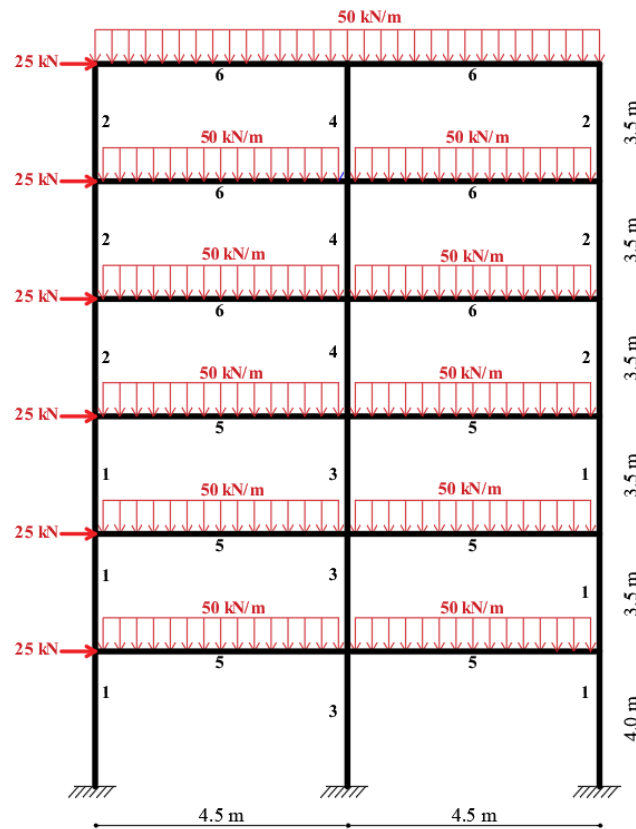


Figure 5-5: Grouping rearrangement of six-storey, two bay rigid steel frame

Table 5.2: The optimum solution after grouping rearrangement obtained by current study

Element	Current study		
	GA	PSO	HSA
Column (1)	203 x 203 x 46	305 x 305 x 97	203 x 203 x 46
Column (2)	203 x 203 x 52	152 x 152 x 30	203 x 203 x 46
Column (3)	356 x 368 x 129	254 x 254 x 73	356 x 368 x 129
Column (4)	254 x 254 x 73	305 x 305 x 97	254 x 254 x 73
Beam (5)	533 x 165 x 66	457 x 152 x 60	533 x 165 x 66
Beam (6)	356 x 127 x 39	406 x 140 x 39	406 x 140 x 39
Total optimum weight of the structure (kN)	71.24	72.53	69.98
Number of iteration	100	100	100
Number of iteration for best result	57	32.00	32

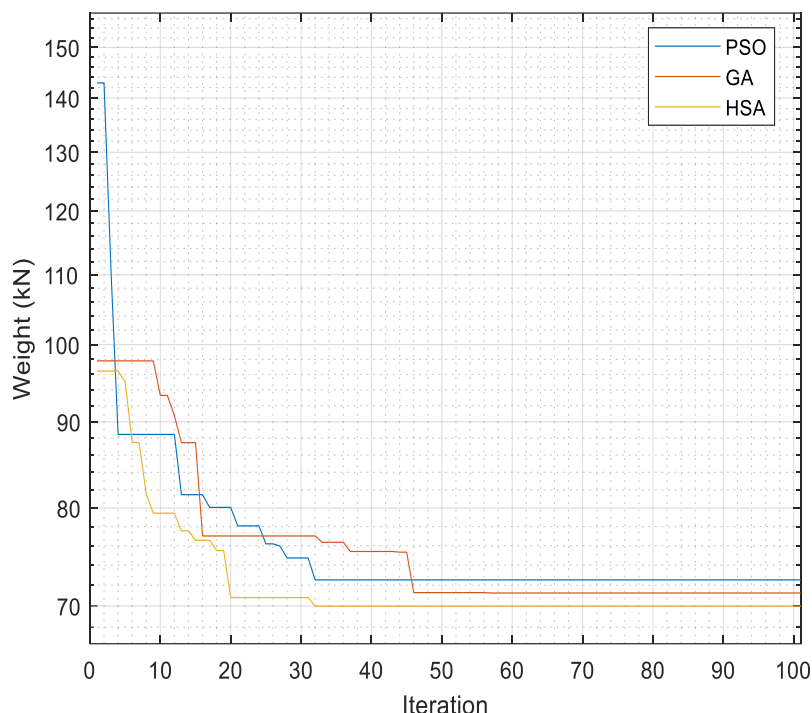


Figure 5-6: Search history for six-storey structure after rearrangement of grouping

5.3.2 Two-dimensional, ten-story, one bay rigid steel frame

The second example is the weight minimization of a one bay ten storey frame. This frame has been investigated by (Camp *et al.*, 2005) and (Issa, 2010). As shown in Figure 5-7, the frame has a span of 9.14m, the first-floor height as 9.14m and 3.66m for the rest of the storeys. The floor beams are subjected to factored loading of 88.2kN/m while the roof beams are subjected to 44.1kN/m. Concentrated lateral actions of 44.2kN/m are observed at the top of each storey and 22.4kN/m at the uppermost floor.

The grouping (9 groups) done by Camp *et al.* is adopted for the investigation. While Camp *et al.* used the AISC-LFRD method, Issa used BS5950. This study used Eurocode 2 (EC3) for defining the constraints of the frame. The frame is assumed rigid and the compression flanges of the beams are assumed fully restrained against lateral torsional buckling. Steel grade used is S275 and the sections are given in UKB and UKC sections.

The parameters used for each algorithm are as follows:

- GA: 100 generations, 200 population size, 100 elite counts and 0.7 crossover fraction
- PSO: 100 iterations and 300 population size

- HSA: 100 iterations, 300 as HMS, 0.3 as PAR, 0.8 as HMCR and 0.3 as BW.

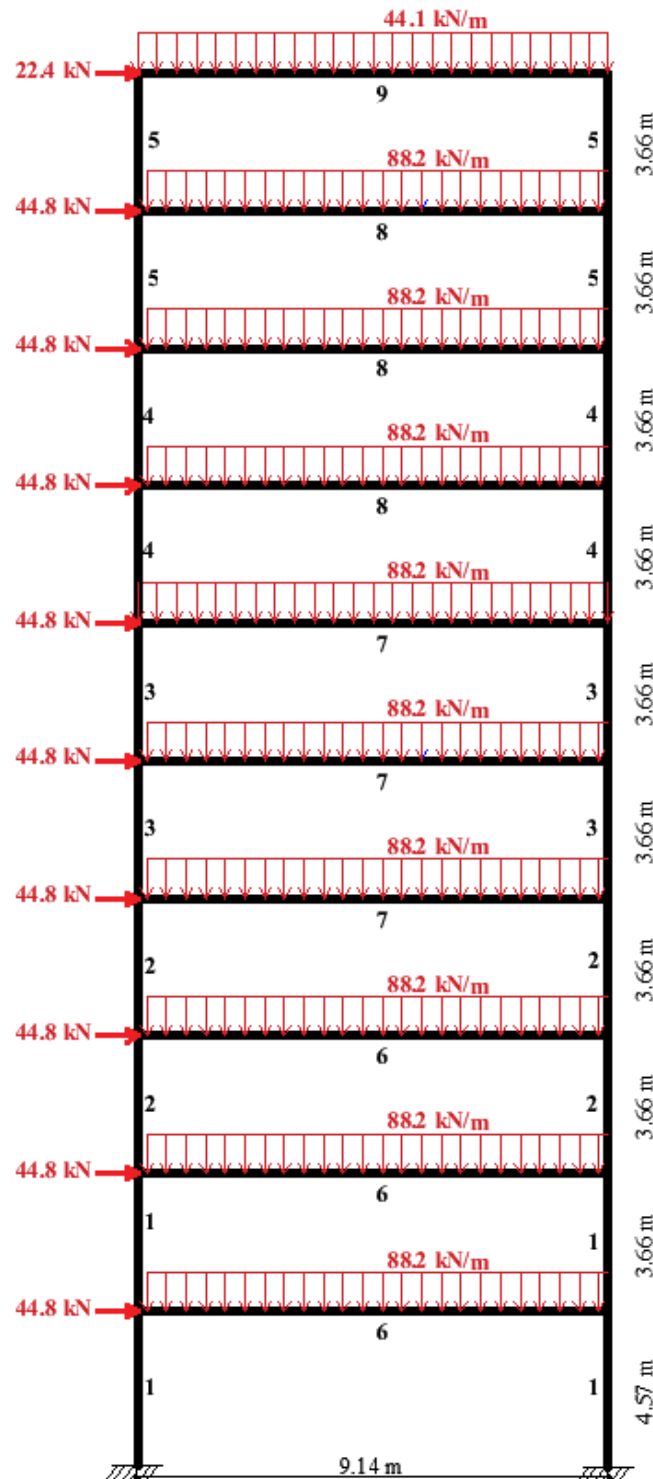


Figure 5-7: Representation of the layout of ten-story, one bay rigid steel frame.

From Table 5.3, the lightest frame was obtained by the GA of this study. The result obtained by Issa nearly matches those obtained in this study. While GA obtained a lesser value to that

of Issa, HSA and PSO obtained slightly higher values. The result by Camp et al using Ant Colony Optimisation (ACO) is approximately 12% higher than the other results. From Figure 5-8, while the DO-DGA converged to the optimum at 100 generations, the algorithms in this study converged to the optimum at 84, 65 and 85 respectively for GA, PSO and HSA. This shows the capability of the algorithms to efficiently obtain the optimum in lesser iterations.

Table 5.3: The optimum solutions obtained by current study, Issa, (2010) and Camp, et al (2005)

Element	Current study			Issa, 2010	Camp et al, 2005
	GA	PSO	HSA	DO-DGA	ACO
Column (1)	356 x 406 x 235	356 x 406 x 235	356 x 406 x 235	356x406x287	W14x233
Column (2)	356 x 368 x 177	356 x 368 x 177	356 x 368 x 177	356x406x235	W14x176
Column (3)	356 x 368 x 153	356 x 368 x 153	356 x 368 x 177	356x368x177	W14x145
Column (4)	356 x 368 x 129	305 x 305 x 137	356 x 368 x 129	356x368x129	W14x99
Column (5)	254 x 254 x 73	254 x 254 x 73	254 x 254 x 73	254x254x107	W12x65
Beam (6)	838 x 292 x 194	838 x 292 x 194	914 x 305 x 201	762x267x134	W30x108
Beam (7)	762 x 267 x 173	838 x 292 x 176	762 x 267 x 147	762x267x134	W30x90
Beam (8)	610 x 178 x 100	610 x 178 x 100	610 x 178 x 100	610x229x101	W27x84
Beam (9)	356 x 171 x 51	356 x 171 x 51	457 x 152 x 52	457x191x74	W21x44
Total optimum weight of the structure (kN)	247.76	251.27	249.32	249.285	283.994
Number of iteration	100	100	100	150	-
Number of iteration for best result	48	34	28	-	-

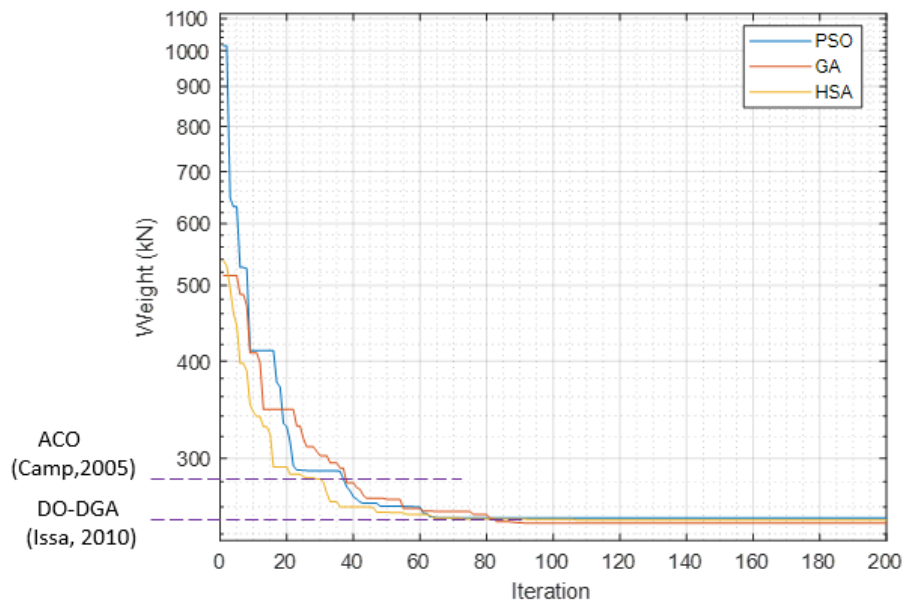


Figure 5-8: Search history for ten-storey structure

5.3.3 Three-dimensional, five-story, two bays steel structure

The first example for the validation of three-dimensional frames is the five story rigid frame initially designed by (Aydoğdu, 2010). The layout, grouping and loading of the frame is represented in Figure 5-9. It can be observed that the frame has two bays in both x and z direction. The frame with 54 joints and 105 members was divided into 11 groups. The unfactored dead and live load are respectively 2.88kN/m^2 and 2.39kN/m^2 . Aydoğdu also considered wind load and ground snow load as 2.5kN/m^2 and 0.755kN/m^2 respectively. The loading after using safety factors are then represented in the figure. While Aydoğdu limited the constraints according to ASCE standard, this study used EC2.

The previous study used harmony search and ant colony optimisation as the optimisation algorithm. All three algorithms in use in this study are checked against the previous study. The parameters set for each algorithm are:

GA: 200 generation size, 250 population size, 100 elite counts and 0.75 crossover fraction.

PSO: 200 iterations and 200 population size

HSA: 300 for HMS, 200 cycles, 0.35 for BW and PAR and 0.85 for HMCR.

From Table 5.4 and Figure 5-10, it can be observed that only the HSA for this study obtained a lighter frame than the ACO used by Aydoğdu. Both PSO and GA obtained frames that are above it although the weight was much lower than the HSA result obtained by Aydoğdu. Another point of note is the number of iteration fixed and required. While this study set the iteration size to 200, the previous study used 5000. The iterations that were actually required for this study were 70,122 and 193 which was just a minimal percentage of that by Aydoğdu (3,300 and 4,200). An observation from the parameters for HSA in Aydoğdu study reveals that the HMS set was just 20 while this study set 300. Past literature (Saka, 2007) has suggested that inadequate population size can lead to incomplete optimisation giving a local optimum. This comparison therefore demonstrates a superior performance of the algorithms in this study.

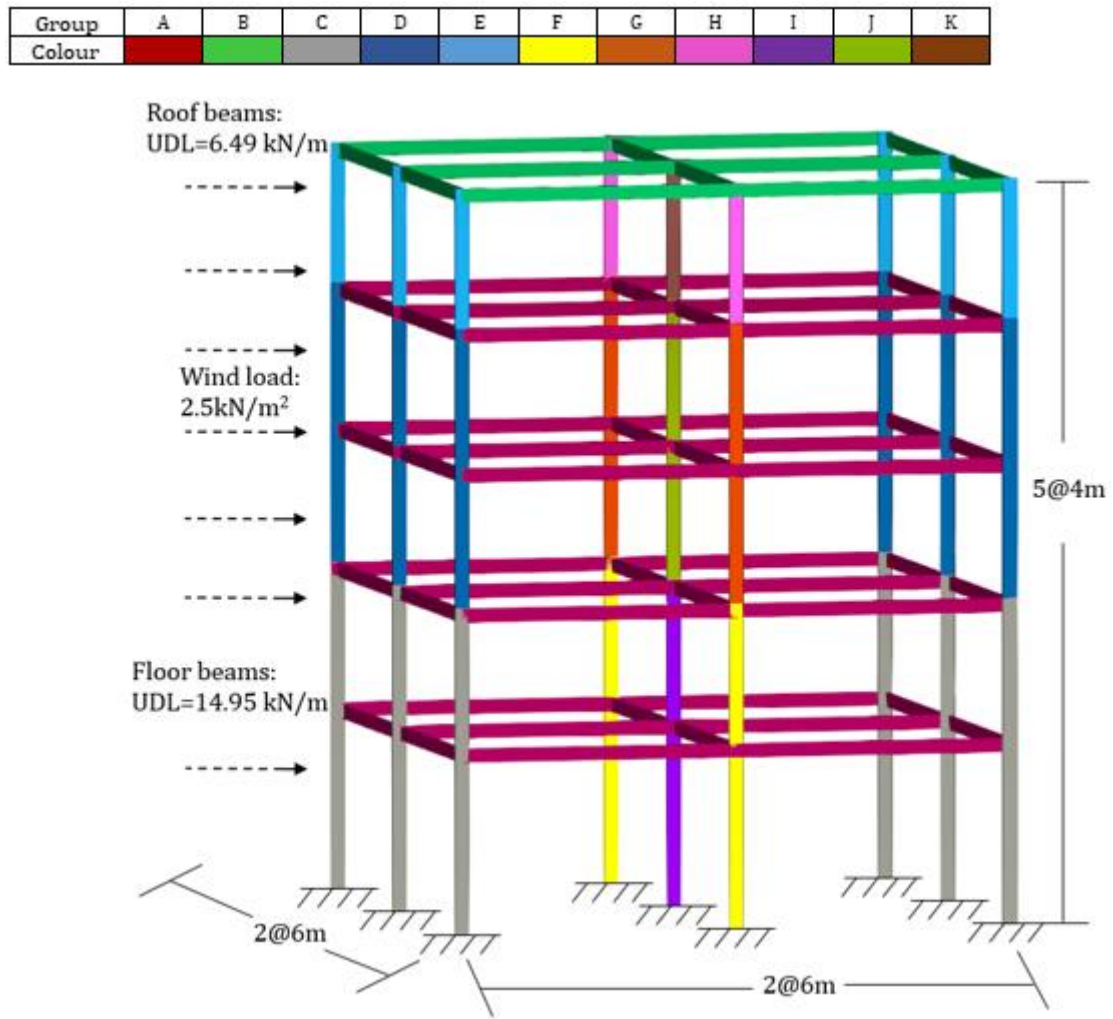


Figure 5-9: Five-story, two bay rigid steel structure

Table 5.4: The optimum solutions obtained by current study and Aydođdu (2010)

Element	Current study			Aydođdu (2010)	
	GA	PSO	HSA	ACO	HS
Column (A)	254 x 254 x 73	305 x 305 x 97	203 x 203 x 60	W460 x 52	W360 x 44
Column (B)	305 x 305 x 97	356 x 368 x 129	305 x 305 x 97	W200 x 35.9	W310 x 38.7
Column (C)	254 x 254 x 73	356 x 368 x 153	254 x 254 x 107	W200 x 35.9	W310 x 38.7
Column (D)	254 x 254 x 89	254 x 254 x 73	254 x 254 x 89	W310 x 38.7	W310 x 60
Column (E)	305 x 305 x 97	254 x 254 x 89	305 x 305 x 97	W360 x 57.8	W610 x 113
Column (F)	305 x 305 x 97	305 x 305 x 118	305 x 305 x 97	W460 x 52	W530 x 66
Column (G)	203 x 203 x 60	203 x 203 x 52	203 x 203 x 60	W310 x 86	W610 x 101
Column (H)	254 x 254 x 73	254 x 254 x 73	203 x 203 x 71	W610 x 101	W1000 x 296
Column (I)	254 x 254 x 73	203 x 203 x 46	203 x 203 x 100	W530 x 66	W610 x 82
Beam (J)	178 x 102 x 19	178 x 102 x 19	152 x 89 x 16	W460 x 89	W610 x 101
Beam (K)	305 x 165 x 46	305 x 165 x 40	356 x 171 x 45	W690 x 170	W1100 x 433
Total optimum weight of the structure (kN)	268.48	274	260.4	265.38	307.38
Number of iteration	200	200	200	5000	5000
Number of iteration for best result	122	70	193	3300	4200

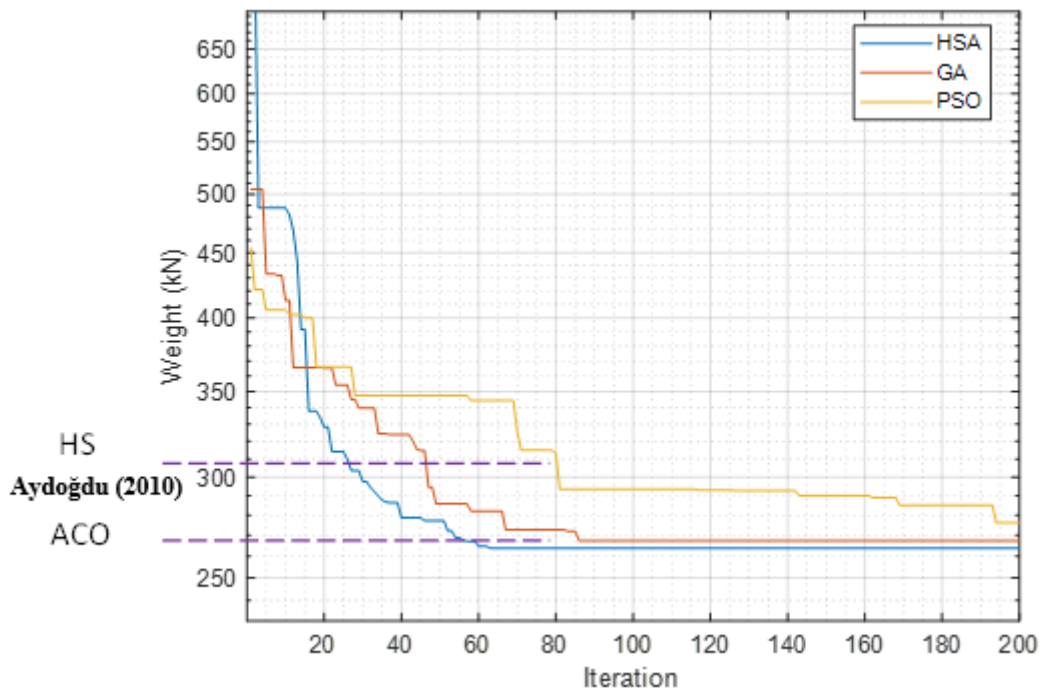


Figure 5-10: Search history for five-storey structure

5.3.4 Three-dimensional, ten-story, two X-bay and three Z-bay steel structure

This example investigates the efficiency of the developed algorithm for a complex structure comprising of many members. It was initially designed by (Kaveh and Talatahari, 2012) and (Talatahari et al., 2015). The structure is a ten-storey frame composed of 290 members connected by 132 joints as shown in Figure 5-11. It has two bays in the X direction and three bays in the Z direction. Each story has its columns divided into three groups; outer, inner and corner columns and the beams are divided into 2 groups; outer and inner beams. The grouping is done in such a way that starting from the ground floor, every three adjacent storeys have the same section for its members while the last floor is a stand-alone i.e. not grouped with other floors. This leads to a total number of 20 groups. The frame is subject to both gravity loads and lateral loads. The gravity load for the inner floor beams is 20kN/m while for outer floor beam is 10 kN/m. That for the roof beams is 15 kN/m and 7.5 kN/m respectively for inner and outer beams. The bases of the frame are fixed.

Kaveh and Talatahari used the Charged System Search (CSS) for the optimisation while Talatahari et al used the Eagle Strategy with Differential Evolution (ES-DE). Only HSA is used in this study due to the high number of groups with lower capacity computer which led to longer time for computation. The parameters used for the HSA algorithm are: 500 iterations and 400 HMS, 0.35 BW, 0.85 HMCR and 0.35 PAR. Both journals based the constraints on AISC-LFRD while this study used EC 2.

The results of the optimisation showing weight, iterations required and set, as well as sections for the members are presented in Table 5.5, where the difference between current solution and the literature is illustrated in Figure 5-12. The lightest frame is observed to be obtained by this study while the heaviest frame was obtained by Kaveh using CSS. The percentage difference is over 2%. It is also observed that the number of iterations required for the HSA in this case is up to 390 as against previous computations of the algorithm in this study. It thus shows that the algorithm can effectively handle problems with varying level of complexity.

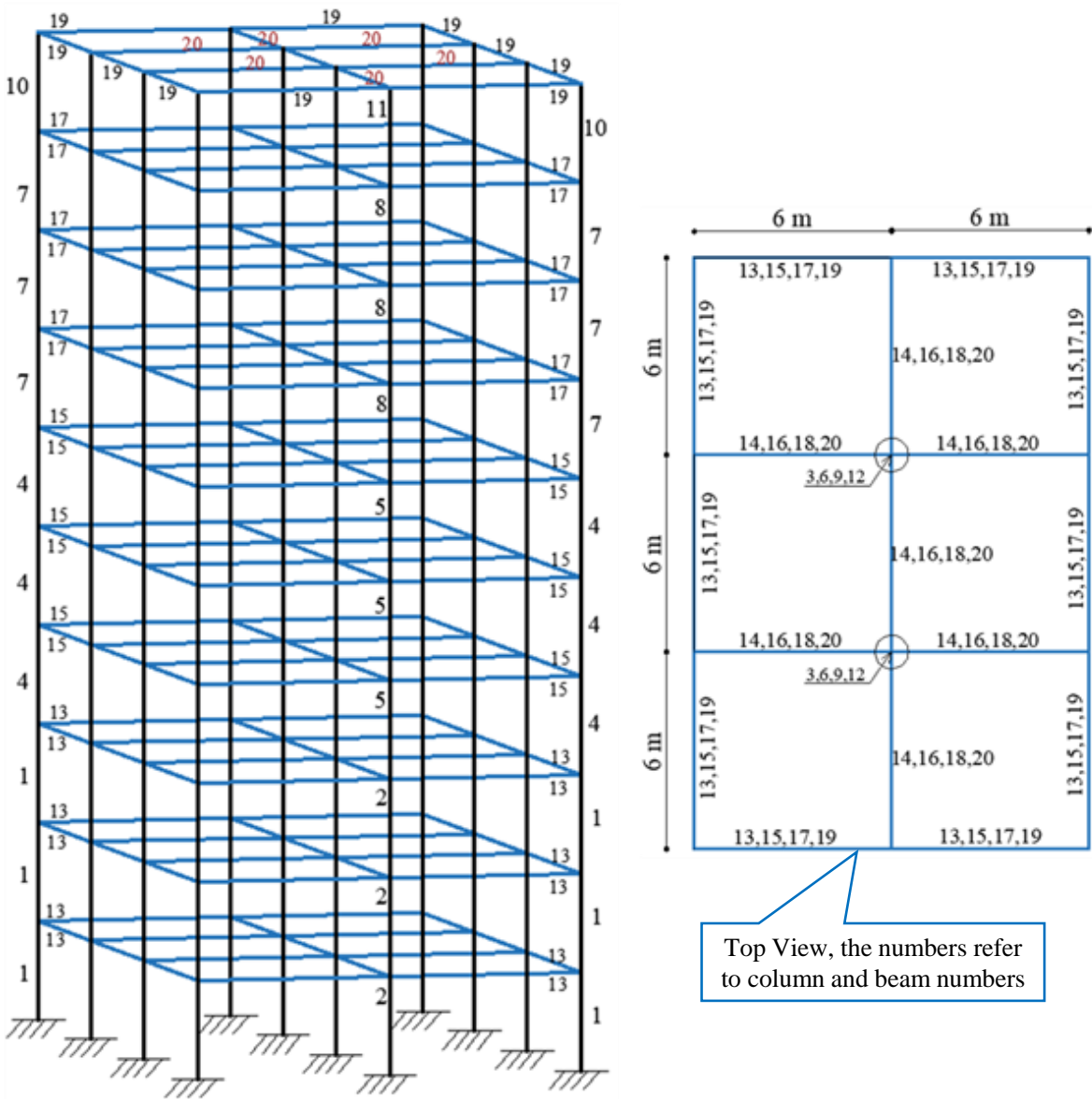


Figure 5-11: Ten-story rigid steel structure

Table 5.5: The optimum solutions obtained by current study, (Kaveh and Talatahari, 2012) and (Talatahari et al, 2015).

Element	Current study	Kaveh and Talatahari, 2012	Talatahari et al, 2015
	HSA	CSS	ES-DE
Column (1)	152 x 152 x 51	W12X58	W12X65
Column (2)	152 x 152 x 44	W18X76	W18X76
Column (3)	254 x 254 x 89	W14X82	W18X86
Column (4)	203 x 203 x 52	W10X54	W16X50
Column (5)	152 x 152 x 37	W18X60	W24X62
Column (6)	305 x 305 x 97	W18X65	W21X68
Column (7)	152 x 152 x 37	W8X24	W14X30
Column (8)	203 x 203 x 60	W16X36	W14X38
Column (9)	254 x 254 x 73	W21X50	W8X48
Column (10)	152 x 152 x 30	W10X15	W8X13
Column (11)	152 x 152 x 30	W8X21	W12X22
Column (12)	152 x 152 x 30	W16X36	W12X53
Beam (13)	305 x 102 x 33	W18X40	W18X40
Beam (14)	457 x 191 x 106	W14X22	W16X19
Beam (15)	305 x 102 x 33	W18X40	W10X36
Beam (16)	457 x 191 x 98	W14X22	W14X26
Beam (17)	152 x 89 x 16	W16X31	W14X26
Beam (18)	406 x 140 x 46	W10X17	W10X17
Beam (19)	305 x 165 x 54	W14X26	W14X22
Beam (20)	305 x 165 x 46	W8X21	W12X26
Total optimum weight of the structure (kN)	738.45	753.88	745.08
Number of iteration	500	-	-
Number of iteration for best result	390	-	-

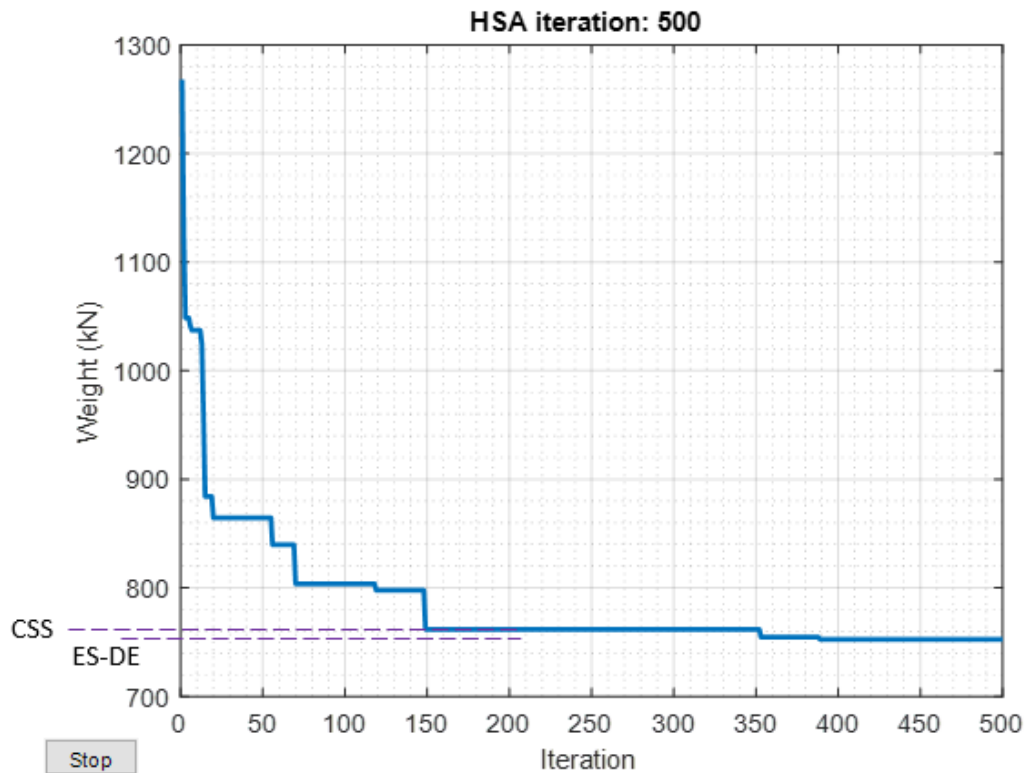


Figure 5-12: Search history for Ten-storey structure

5.4 Summary

This chapter demonstrate the efficiency of the developed algorithms in solving simple-to-complex steel frames. Both two-dimensional and three-dimensional moment resisting steel frames were presented. The frames were checked against previous studies. The results obtained showed the good performance of the algorithms, with HSA almost superseding all other algorithms compared. The others also performed well in providing results that are very close to the global optimum. Although varying design standards were used by the previous authors, the comparability of the result signifies a working algorithm.

CHAPTER SIX. NUMERICAL EXAMPLES

6.1 Introduction

This chapter presents several examples of 2D and 3D modelling of steel structures where single and multi-objective optimisations were carried out. The algorithms were first tested on 2D steel frames due to their simplicity. Three examples were carried out but only one is included in this chapter, while the other two are presented in appendix-B. For the 2D frames, three objective functions were investigated, two as a single function which includes weight and embodied energy and one as a multi-objective (weight + embodied energy).

Furthermore, three examples also presented for three-dimensional frames. In this case, the applied optimisation methods were tested for single and multi-objective problems, where the single objective was set as a minimum embodied energy and the multi-objective was set as a minimum of combined functions (weight + embodied energy). For three-dimensional optimisation of steel structures, the weight objective function was neglected, because of the linearity of weight with embodied energy functions. The optimisation was performed in MATLAB in which UKB and UKC sections are defined as the design variables.

Conventional design for the above-mentioned examples were also done using MATLAB without implementing the optimisation techniques, where a random selection was done until all the design conditions were satisfied. This was carried out for value engineering of steel structures with state-of-the-art optimisation techniques.

6.2 Two-Dimensional Steel Framed Structures

The efficiency of the proposed optimisation methods for two-dimensional frames was verified with three numerical examples. They are three storey, six storey and nine storey moment resisting frames. A similar loading arrangement was adopted for the three examples. The frames were assumed to be for residential purposes and thus have the sum of the variable and permanent action as 28.5kN/m for floor beams and 12.375kN/m for roof beams. The wind action applied in one direction parallel to the frame is 3.75kN/m. The permanent, variable and wind actions were factored by 1.35, 1.5 and 0.75 respectively. A better description of the

loading arrangement will be stated in the example below. The results for the remaining two examples are presented in appendix B.

6.2.1 Numerical examples

6.2.1.1 Three-storey, three-bay frame

The frame represented in Figure 6-1 is a three-storey three-bay steel frame consisting of 21 members and 16 joints. The frame layout, member groupings, dimensions, loading and base supports of the moment frames are also represented in the figure 6-1. The base supports are observed to be rigid. From figure 6-1, it is observed that there are 4 independent groupings of the members; group A represents the exterior columns, B for the interior columns, C for the floor beams and D for roof beams. The grouping gives the columns a length of 9m from ground to roof. The beams are of 10m span each.

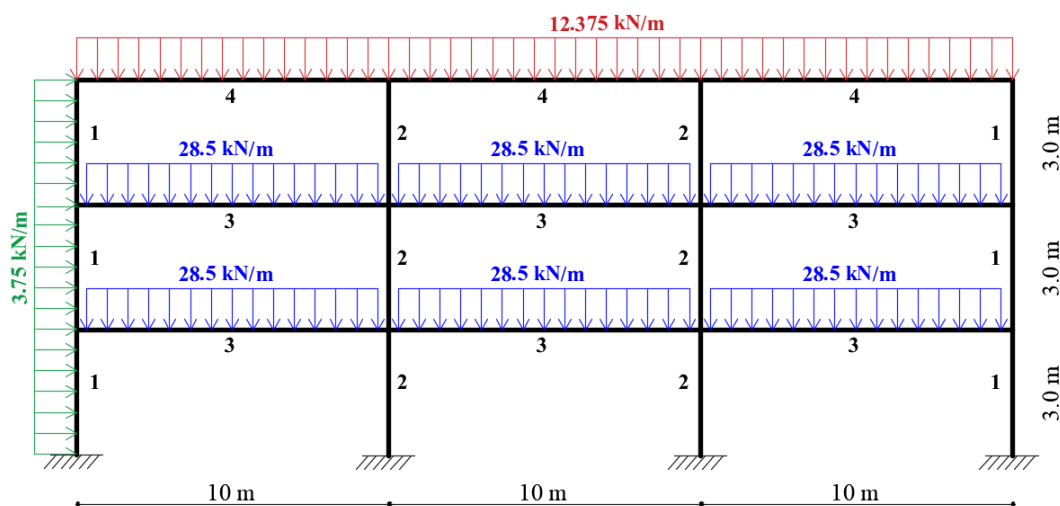


Figure 6-1: Representation of the layout and loading of the three-storey frame

1- Weight function:

For the genetic algorithm adopted for the weight function, the population size was set to 100, number of generations to 100, elite counts also to 100 and the cross over function as 0.7 after many trials. The trial range for both population and generation size was from 50-200. Both were fixed when the minimum weight found became constant. The stochastic nature of GA means slightly different result are expected for each run as randomly generated population are

used. As such, running the program multiple times is expected to build confidence in the result. Therefore, the result shown in the table below were adopted after 4 runs.

The parameters for PSO were set to 100 iterations and 150 swarm size. The algorithm was also run multiple times to obtain the global optimum. The results after 3 runs were adopted.

In addition, parameters for HSA were also tried and set when the best were found. The harmony memory size (HMS) was set to 200 and the number of cycles to 100. Its crossover probability and mutation probability was set to 0.7 and 0.08 respectively after a trial range of 0.5- 0.9. The HMCR, BW and PAR are respectively 0.8, 0.3 and 0.3. The results were obtained after 5 runs.

Figure 6-2 illustrates the search history for the optimum weight for the three algorithms. The optimum search converged at 45 generations for GA. Although, the 100 maximum generations that was set was not reached, this is however better as it ensures that the best combinations of the variables were investigated. The optimum solution for PSO converged at 19 iterations while HSA converged at 15 iterations. The first to become stable was HSA.

Table 6.1 below present the value of the minimum weight and the corresponding values of all other functions considered with the time for finding the best and total time used for the optimisation. The optimum weight obtained by the three algorithms are very close with PSO and GA obtaining the same result. The least weight is found to be 49.7kN by HSA in 8 secs. The optimum weight obtained by GA and PSO is 50.25kN. While PSO obtained the best in 13 secs, GA used 24 secs.

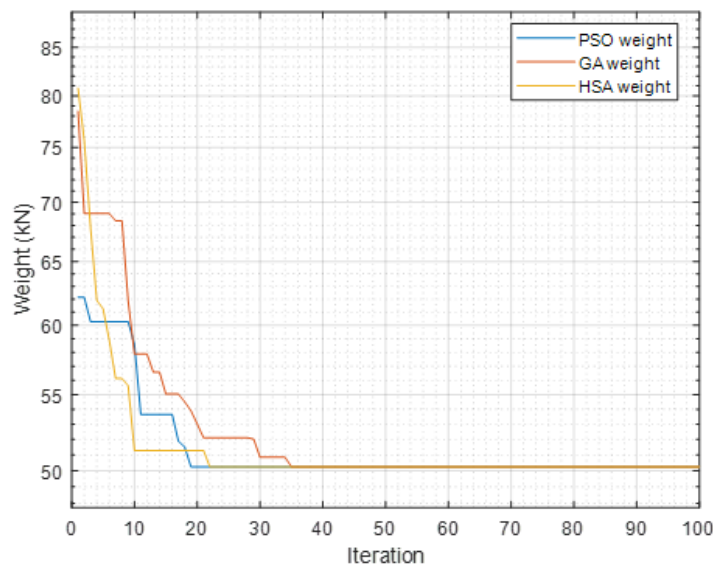


Figure 6-2: Comparison of Search history between GA, HSA and PSO for the three-storey frame (W-objective function)

Table 6.1: Optimisation output for the three-storey frame (W-objective function)

Element	Section		
	GA	PSO	HSA
Column (1)	152 x 152 x 37	152 x 152 x 37	203 x 203 x 46
Column (2)	152 x 152 x 37	152 x 152 x 37	152 x 152 x 30
Beam (3)	406 x 140 x 46	406 x 140 x 46	356 x 171 x 45
Beam (4)	254 x 146 x 31	254 x 146 x 31	203 x 133 x 30
Total optimum weight of the structure (kN)	50.25	50.25	49.70
Total optimum Embodied Energy of the structure (MJ)	119715	119715	108583
Total optimum Embodied Carbon of the structure (kgCO ₂ e)	8734	8734	8641
Total cost of the structure (£)	9606.10	9606.10	9587.50
Number of iteration for best result	45	19	15
Time of finding the best (sec)	24	13	8
Time of running (sec)	51	25	52

2- Embodied Energy objective function:

In the case of the embodied energy function, the same parameters were used as that of the weight function. That is, 100 were set for the population and generations of the genetic algorithm as well as 100 elite counts and 0.7 cross over function. PSO also had 100 and 150 iterations and swarm size respectively. HSA had 200 HMS, 100 iterations, 0.8 HMCR, 0.3 BW and 0.3 PAR. All algorithms were run multiple times and the results that gave the minimum value for each algorithm were picked.

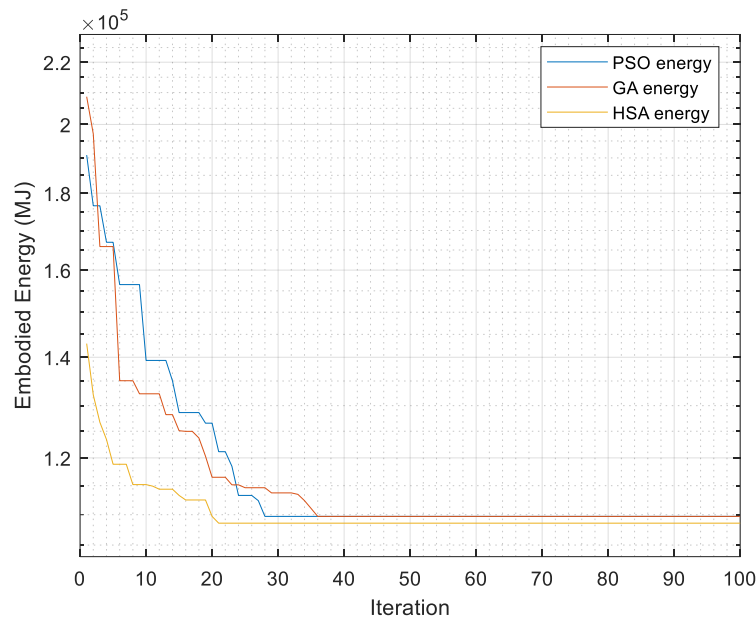


Figure 6-3: Comparison of Search history between GA, HSA and PSO for the three-storey frame (EE-objective function)

Table 6.2: Optimisation output for the three-storey frame (EE-objective function)

Element	Section		
	GA	PSO	HSA
Column (1)	152 x 152 x 37	152 x 152 x 37	203 x 203 x 46
Column (2)	152 x 152 x 37	152 x 152 x 37	152 x 152 x 30
Beam (3)	406 x 140 x 46	406 x 140 x 46	356 x 171 x 45
Beam (4)	254 x 146 x 31	254 x 146 x 31	203 x 133 x 30
Total optimum weight of the structure (kN)	50.25	50.25	49.70
Total optimum Embodied Energy of the structure (MJ)	119715	119715	108583
Total optimum Embodied Carbon of the structure (kgCO ₂ e)	8734	8734	8641
Total cost of the structure (£)	9606.10	9606.10	9587.50
Number of iteration for best result	36	28	21
Time of finding the best (sec)	20	13	10
Time of running (sec)	47	21	46

Figure 6-3 illustrates the search history for the optimum weight for the three algorithms. The optimum search converged at 36 generations for GA, again not reaching the maximum iteration size set. The optimum solution for PSO converged at 28 iterations while HSA converged at 21 iterations. HSA was again the first to become stable.

Table 6.2 below present the value of the minimum weight and the corresponding values of all other functions considered with the time for finding the best and total time used for the optimisation. The optimum weight obtained by the three algorithms are very close, with PSO and GA obtaining the same result. The least weight is found to be 49.7kN by HSA in 10 secs.

The optimum weight obtained by GA and PSO is 50.25 kN. While PSO obtained the best in 13 secs, GA used 20 secs.

3- Multi objective function:

The parameters used for the multi-objective differed from the single objectives functions. The genetic algorithm had 200 iterations and 150 population size. The cross over function and elite count remain 0.7 and 100 respectively. The iteration and swarm size set for PSO were 150 each. HSA had 200 HMS, 200 iterations, 0.85 HMCR, 0.3 BW and 0.3 PAR. Its crossover probability was 0.72 while the mutation probability was 0.08. The trial range for all iterations and generation size was 100-300. All algorithms were run multiple times and the results that gave the minimum value for each algorithm were picked. GA required 5 runs, PSO used 7 runs and HSA was 6 runs.

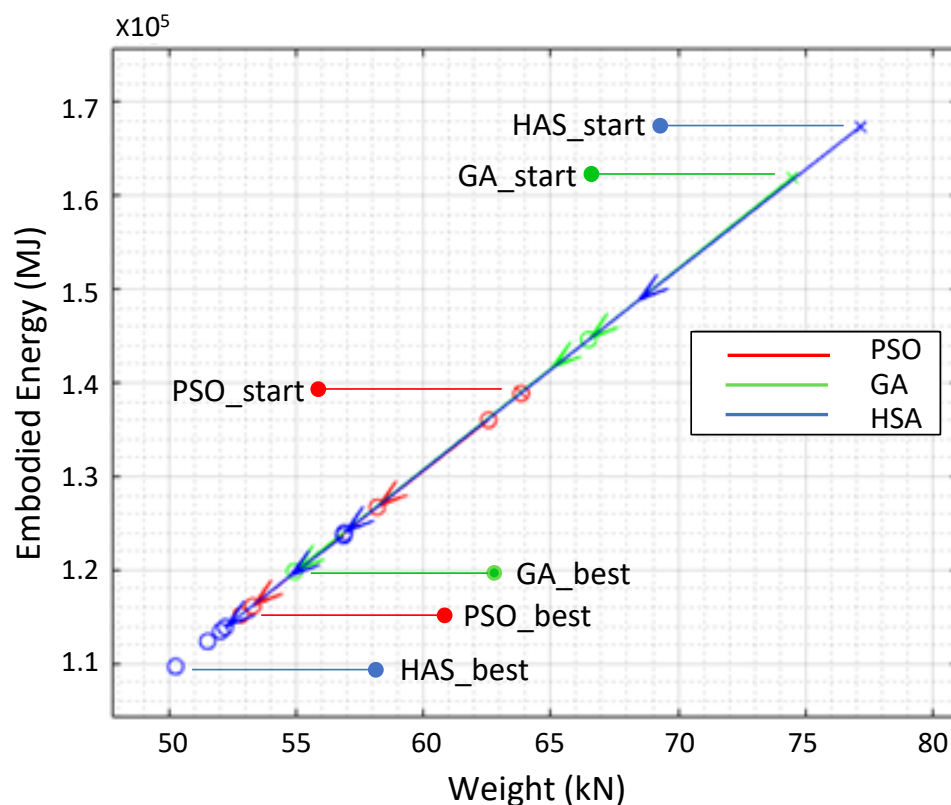


Figure 6-4: Comparison of Search history between GA, HSA and PSO for the three-storey frame (Multi-objective function)

Table 6.3: Optimisation output for the three-storey frame (Multi-objective function)

Element	Section		
	GA	PSO	HSA
Column (1)	254 x 254 x 73	203 x 203 x 46	152 x 152 x 37
Column (2)	203 x 203 x 52	152 x 152 x 44	152 x 152 x 37
Beam (3)	406 x 140 x 39	406 x 140 x 46	406 x 140 x 46
Beam (4)	203 x 133 x 30	203 x 133 x 30	254 x 146 x 31
Total optimum weight of the structure (kN)	54.9	52.8	50.2
Total optimum Embodied Energy of the structure (MJ)	119853	115190	109715
Total optimum Embodied Carbon of the structure (kgCO ₂ e)	9543	9174	8734
Total cost of the structure (£)	10433.9	9940.1	9606.1
Number of iteration for best result	5	41	17
Time of finding the best (sec)	3	35	11
Time of running (sec)	30	94	94

Figure 6-4 illustrates the search history for the optimum weight for the three algorithms. The optimum search converged at 5 generations for GA, which is just a fraction of the maximum set. The optimum solution for PSO converged at 41 iterations while HSA converged at 17 iterations.

Table 6.3 presents the value of the minimum weight and the corresponding values of all other functions considered with the time for finding the best and total time used for the optimisation. The optimum weight obtained by the three algorithms are close. The least weight is found to be 50.2kN by HSA in 11 secs. The optimum weight obtained by PSO is 52.8 kN in 35 secs while GA obtained the highest weight of 54.9 in 3 secs.

4- Discussion:

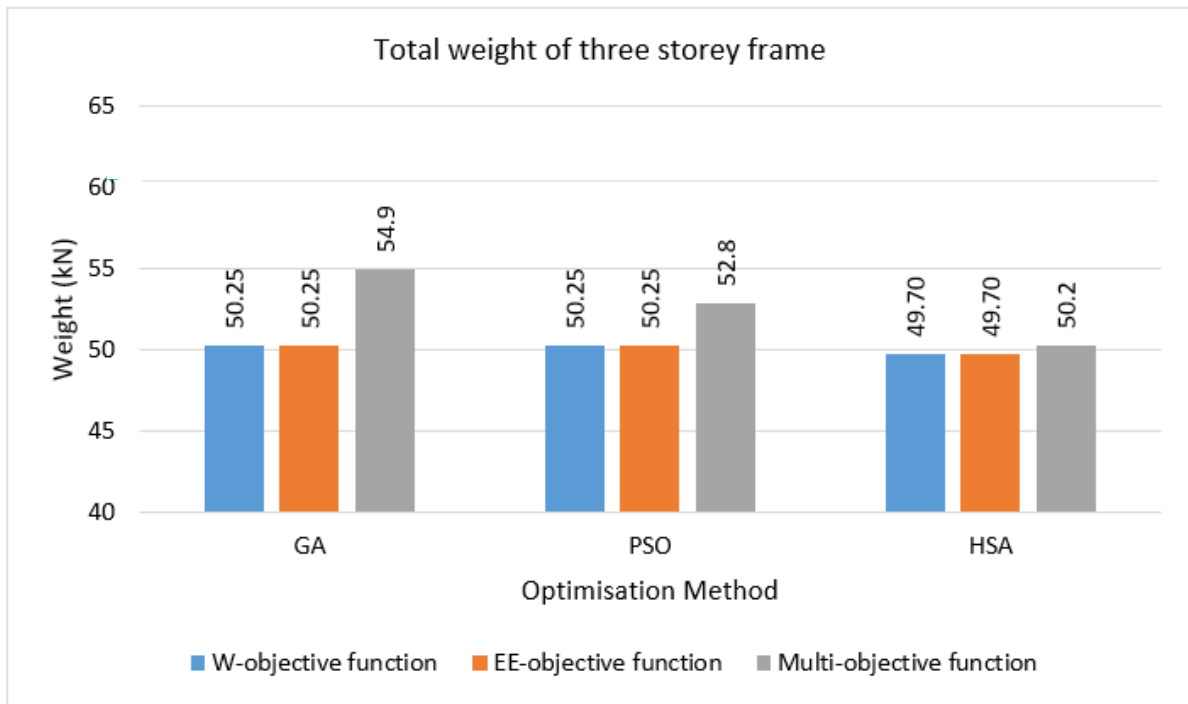


Figure 6-5: Schematic representation of the difference in weight among the three objective functions

The optimum weight obtained by the algorithms for all the three functions investigated are represented in Figure 6-5. It is observed that all three algorithms got the same result for single objectives. HSA was able to perform close to the single objectives but GA and PSO obtained results that are a bit further away. Overall, HSA obtained the lowest result all through. PSO and GA performed in the same manner for single objectives but PSO was better for multiobjective function.

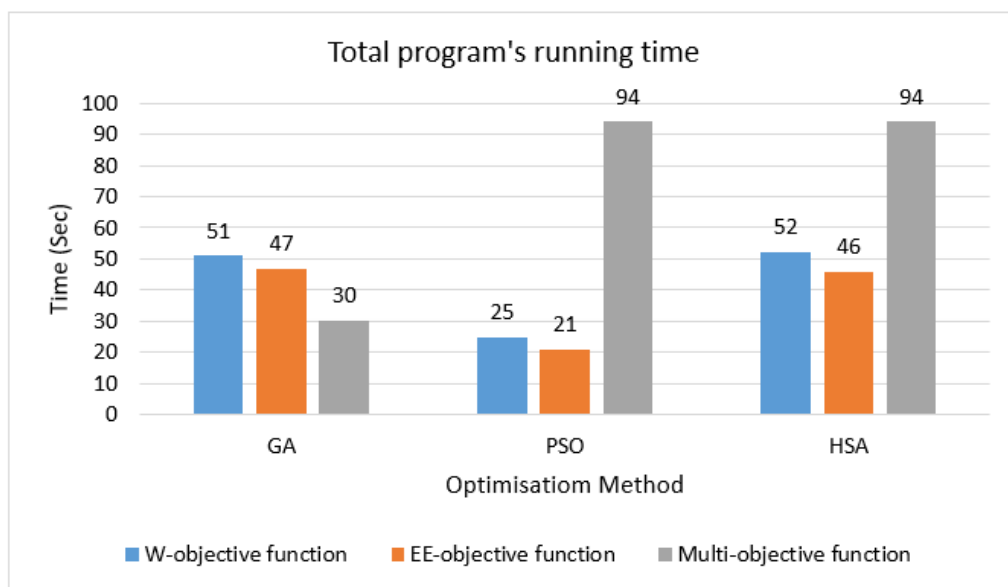


Figure 6-6: Running time average for the three algorithms for the three objective functions.

Figure 6-6 represents the total running time needed by each algorithm for the optimisation. It is observed that embodied energy optimisation was faster for the algorithms than weight optimisation. PSO required the least time for the embodied energy optimisation while HSA used the longest time. While PSO and HSA required more time for multi-objective optimisation, GA was faster than its single objective optimisations. It is also observed that HSA and PSO used the same running time for the multi-objective. It can be concluded from here that GA is the fastest for multi-objective optimisation while PSO is the fastest for single-objective optimisation.

6.3 Three-Dimensional Steel Framed Structures

For the three-dimensional optimisation, three examples were also used to verify the algorithms viz: three-storey, six-storey and nine-storey frames. The plan area of the frames is 30mx30m in x and y direction. The height of the ground floor is 4m while that of the other floors is 3m. Similar loading arrangement was adopted for the three examples. The permanent action for both floor and roof beams were taken as 5kN/m², the imposed action for the floor beams and roof beams respectively is 2kN/m² and 0.6kN/m². The wind action uniformly applied parallel to the frame in one direction is 4.5kN/m. The permanent, variable and wind actions were factored based on EC-3 using 1.35, 1.5 and 0.75 respectively. The base of the frames is assumed to be fixed. A better description of the loading arrangement will be given in each example.

As mentioned in section 6.1, due to the linearity of weight with embodied energy functions, the weight objective function was neglected. The single objective function was set as a minimum embodied energy and the multi-objective was set as a minimum of combined functions (weight + embodied energy).

6.3.1 Three-bay three-story steel structure

Figure 6-7 designates a three-storey space frame consisting of 120 members and 64 joints. The frame layout, member groupings, loading and base support of the moment frame are also represented. From the figure, it is observed that there are 4 independent groupings of the members; group A represents the exterior columns, B for the interior columns, C for the floor beams and D for roof beams. Each column thus has a height of 10m and the beams of 10m span.

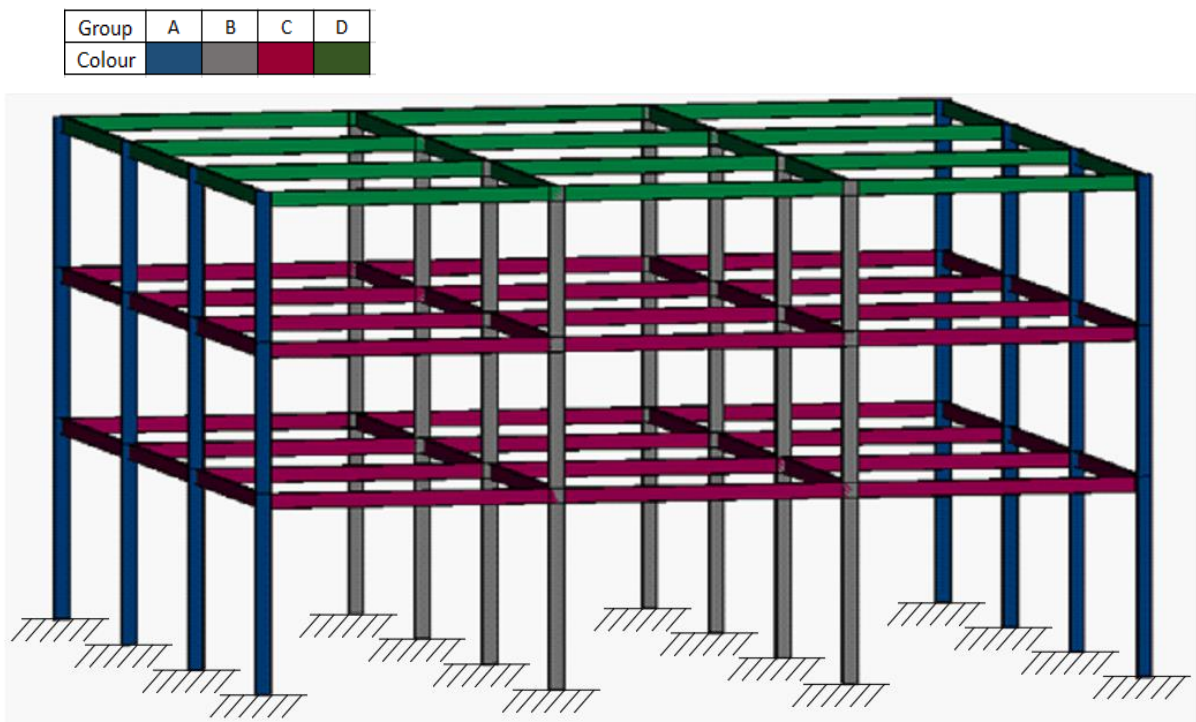


Figure 6-7: Representation of the layout and loading of the three-storey structure

1- Embodied objective function:

Figure 6-8 shows the search history for embodied energy for the three optimisation algorithms. The figure was obtained after a number of runs for each algorithm. The parameters used for GA are: 100 generation size, 150 population, 100 Elite counts and 0.7 cross over function. For

PSO, the iteration size was also 100 while the swarm size was set to 200. Lastly, HSA was set to 100 iterations and 200 harmony memory (HMS). The other parameters for HSA were: 0.85 for HMCR, 0.3 for PAR and 0.3 also for BW. The parameters were set after many trials on the algorithms. The parameters that gave the best result were then selected. From the search history, it can be observed that the first to reach its optimum was PSO while GA and HSA required more iterations to reach the optimum.

Table 6.4 gives the result of the optimum values obtained by each algorithm for the weight, embodied energy, cost and embodied carbon. It also describes the number of iteration required, the time at which the optimum is found and the total time required for the optimisation. The lowest result for EE produced here is 742GJ by HSA while PSO obtained the highest 759GJ when considering the optimisation algorithms alone. The least weight obtained by HSA is 341.57kN. While PSO required the least iterations (21) as well as time for running (56 secs), HSA consumed more time for running and time of finding the best. HSA converged after 33 cycles. GA happens to hold the middle of the spectrum of the three algorithms in terms of the obtained value and time required, however, it required the highest iteration size (46) before becoming stable.

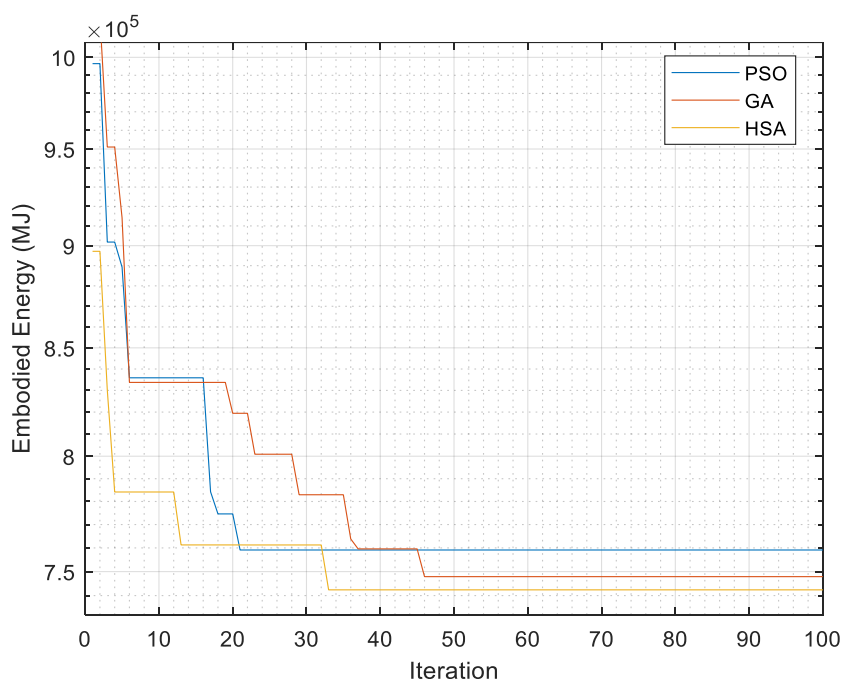


Figure 6-8: Comparison of Search history between GA, PSO and HSA for the three-storey structure (EE-objective function)

Table 6.4: Optimisation output for the three-storey structure (EE-objective function)

Element	Section		
	GA	PSO	HSA
Column (1)	254 x 254 x 73	356 x 368 x 129	305 x 305 x 97
Column (2)	356 x 368 x 129	305 x 305 x 97	356 x 368 x 129
Beam (3)	305 x 165 x 40	305 x 165 x 46	305 x 165 x 40
Beam (4)	305 x 165 x 46	203 x 133 x 25	254 x 146 x 31
Total optimum weight of the structure (kN)	344.13	349.63	341.57
Total optimum Embodied Energy of the structure (MJ)	747.884X10 ³	759.119X10 ³	742.379X10 ³
Total optimum Embodied Carbon of the structure (kgCO _{2,e})	59.666X10 ³	60.590X10 ³	59.224X10 ³
Total cost of the structure (£)	61659.7	60470.6	61270.6
Number of iteration for best result	46	21	33
Time of finding the best (sec)	35.34	30.42	47.53
Time of running (sec)	73	56	144

2- Multi-objective function:

Figure 6-9 gives the comparison of the convergence history for the multi-objective optimisation of the frame using the three algorithms and conventional design. The parameters used by each algorithm for the multi-objective optimisation after many trials are given below:

GA: 160 generation size, 150 population, 100 Elite counts and 0.75 cross over function;

PSO: 120 iteration size and 200 swarm size;

HSA: 180 iteration, 200 harmony memory size, 0.85 for HMCR, 0.3 for PAR and 0.3 for BW.

Since these were the parameters that gave the desirable result, It therefore means that all of the algorithms performed better with an increase in the iteration size as against that used in the single objective functions.

From the search history represented in Figure 6-9, the lowest values were obtained by HSA, while GA obtained the highest values among the three algorithms. All the values obtained were far lesser than that by conventional design. HSA started off from the highest values while PSO started off quite close to the optimum values. However, HSA was still able to adequately search the space to obtain the lowest. GA also started off at the intermediate point but could not obtain the global optimum before terminating.

From Table 6.5, the least EE obtained was 742GJ and was by HSA after 38 iterations. PSO produced 760GJ after 35 iterations and GA obtained 794GJ after 11 iterations. The least weight is 341.57kN. It is observed that GA used the least iteration number while HSA required the

highest. Even though PSO started off close to the optimum values, it still required more time for finding the best and even longer total running time. Furthermore, GA gave the highest values for all the parameters, it however required the lowest time among the algorithms.

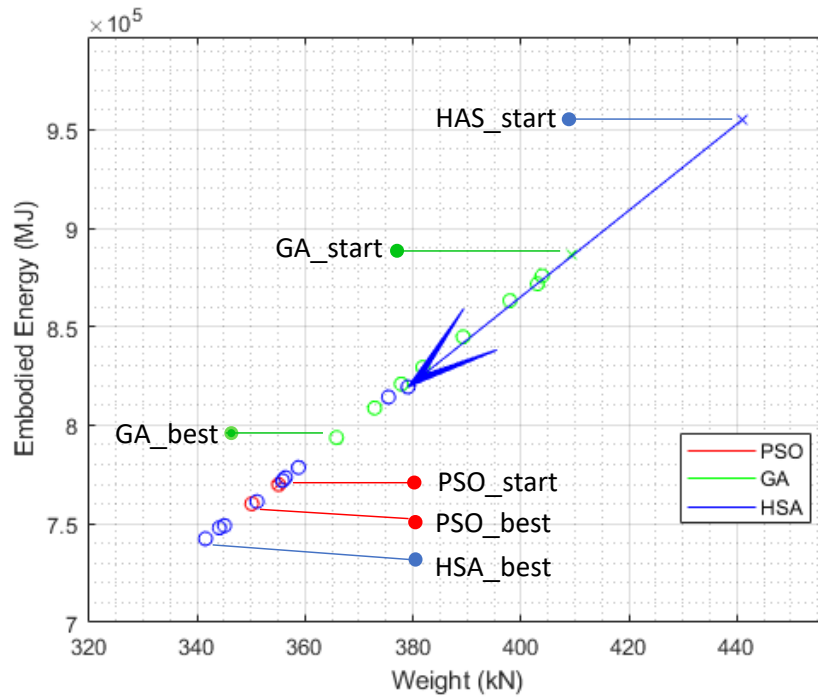


Figure 6-9: Comparison of Search history between GA, PSO and HSA for the three-storey structure (Multi-objective function)

Table 6.5: Optimisation output for the three-storey structure (Multi-objective function)

Element	Section		
	GA	PSO	HSA
Column (1)	356 x 368 x 177	254 x 254 x 89	305 x 305 x 97
Column (2)	254 x 254 x 73	356 x 368 x 129	356 x 368 x 129
Beam (3)	356 x 171 x 45	305 x 165 x 46	305 x 165 x 40
Beam (4)	203 x 133 x 25	203 x 133 x 30	254 x 146 x 31
Total optimum weight of the structure (kN)	365.82	350.29	341.57
Total optimum Embodied Energy of the structure (MJ)	793703	760049	742379
Total optimum Embodied Carbon of the structure (kgCO ₂ e)	63372	60683	59224
Total cost of the structure (£)	63967.2	61313.7	61270.6
Number of iteration for best result	11	35	38
Time of finding the best (sec)	31.42	152.29	56.46
Time of running (sec)	260	839	283

3- Discussion:

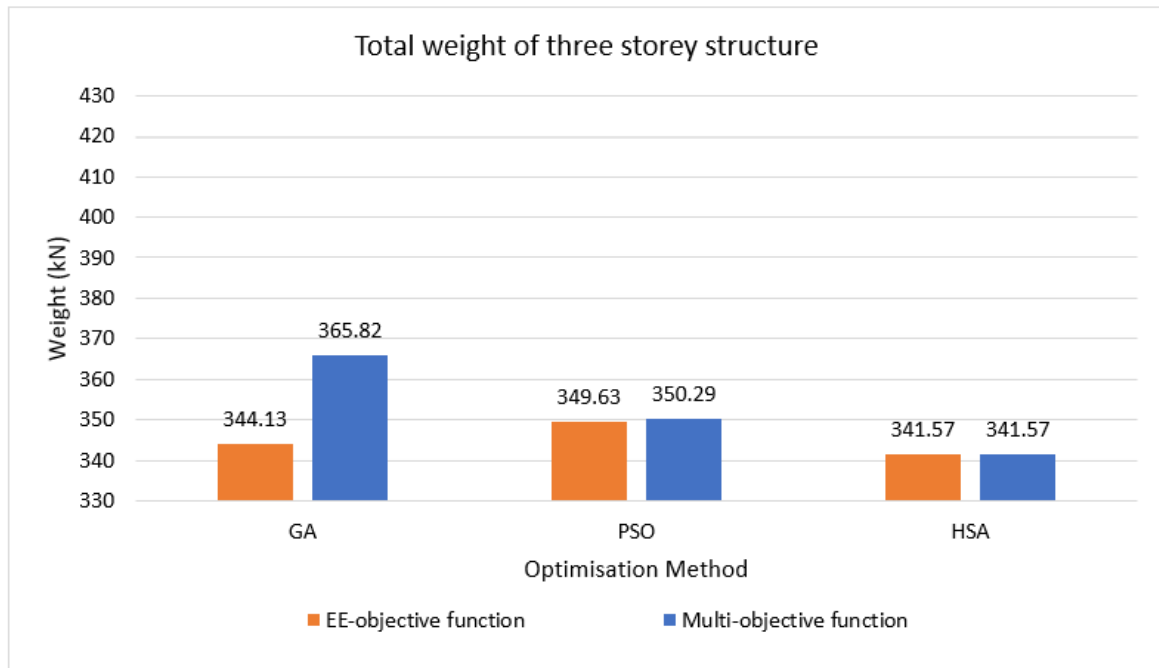


Figure 6-10: Comparison of minimum weight between GA, PSO, HSA and manual for 3 story structure among the two objective functions

Figure 6-10 is a representation of the minimum weight obtained from each objective function minimization using the three algorithms. From the display, only HSA obtained the same result for the multi-objective and EE objective function. PSO also got a very similar result while the result for GA shows a lower value from EE optimisation.

The closest and almost similar results obtained by both HSA and PSO respectively suggest the adequacy of the algorithm in multiobjective optimisation and denote a shortcoming for GA. Although, an even lower value would have been better for the multiobjective optimisation since a trade-off was being envisaged by adopting multiobjective function. However, this result can be linked almost to the total dependence of the embodied energy function on weight.

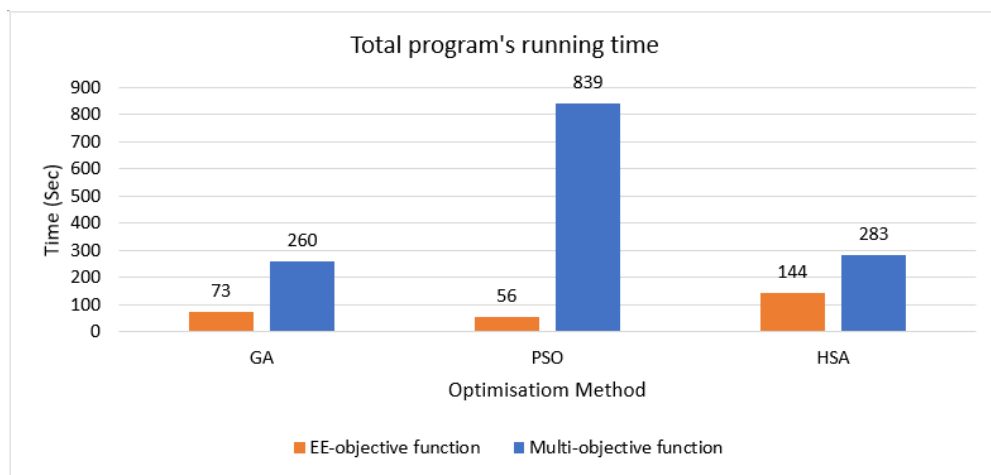


Figure 6-11: Running time average for the three algorithms for the three optimisation methods.

The average running time for the algorithms for the two objective functions optimisation is denoted by Figure 6-11. The longest running time recorded for multi-objective optimisation is 14 minutes which is by PSO. The longest for the single objective is 2.4 mins by HSA. From the figure, it can be deduced that the modus operandi of the algorithms differs between the single and multi-objective optimisation. PSO is faster than the other two when computing single objective optimisation but the slowest for multi-objective optimisation. GA was able to obtain its results in less than 4.5mins although the distance between the single and multi-objective is upto 3mins. HSA on the other hand has its single objective running time just above 2mins and multi-objective function running time above 4.5mins but less than 5mins. However, it has the difference between the single and multi-objective to be about 2mins. The longer time required by each algorithm for multi-objective optimisation confirms that in order for the algorithm to obtain the best trade-off, more rigorous search is required i.e. it needs to properly search the design space, and thus requiring more iteration size and time. Comparing using the population size used by the algorithms too may give another reason as to why some are faster. GA has the lowest population size while the other two are the same. It thus suggests that the size of the population can be a contributory factor to the time used.

6.3.2 Three-bay six-storey steel structure

Figure 6-12 designates a six storey space frame consisting of 240 members and 112 joints. The frame layout, member groupings, loading and base support of the moment frame are also represented. From the figure, it is observed that there are 6 independent groupings of the members; group 1 represents the exterior columns in the first three storey; group 2 represents

the exterior columns for the upper three floors; Group 3 for the interior columns in the first three floors; Group 4 for exterior columns in the upper three floors; 5 for the floor beams and 6 for roof beams.

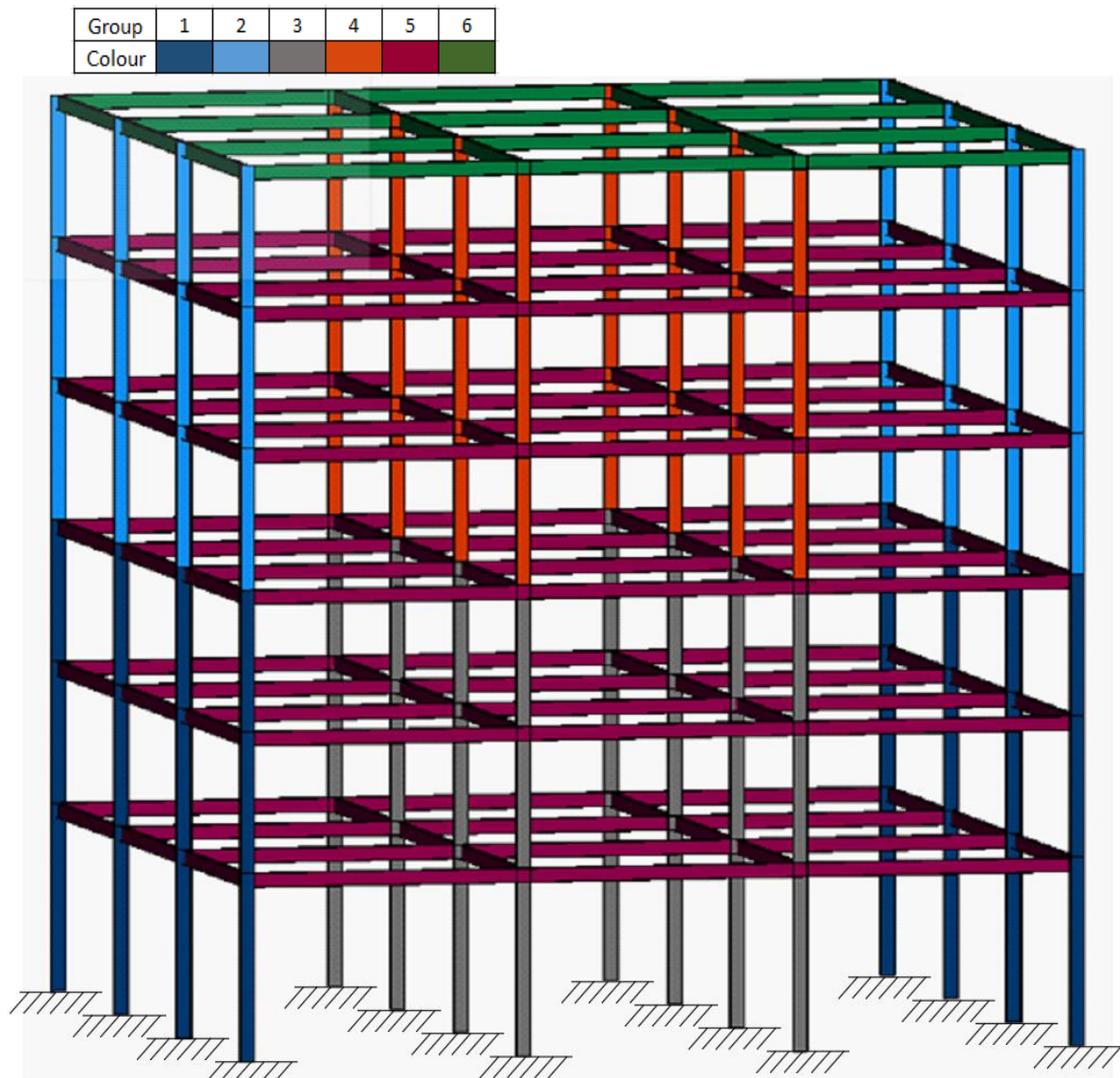


Figure 6-12: Representation of the layout and loading of the six-storey structure

1- Embodied Energy objective function:

Figure 6-13 shows the search history for embodied energy for the three optimisation algorithms. The figure was obtained after a number of runs for each algorithm. The parameters used for GA are: 200 generation size, 200 population, 100 Elite counts and 0.75 cross over function. For PSO, the iteration size was also set to 200 while the swarm size was set to 200. And lastly, HSA also had 200 iteration and 200 population size. The other parameters for HSA were: 0.85

for HMCR, 0.3 for PAR and 0.3 also for BW. From the search history, it can be observed that the first to reach its optimum was PSO while GA and HSA required more iterations to reach the optimum.

Table 6.6 gives the result of the optimum values obtained by each algorithm for the weight, embodied energy, cost and embodied carbon. It also describes the number of iteration required, the time at which the optimum is found and the total time required for the optimisation. The lowest result for EE produced here is 2651GJ by HSA at 44 iterations while PSO obtained the highest value of 2785GJ when considering the optimisation algorithms alone. While PSO required the least iterations (37) as well as time for running (47 secs), HSA consumed more time for running. GA happens to hold the middle of the spectrum of the three algorithms in terms of the obtained value and total time required, however, it required the highest iteration size (46) and used more time in finding the best before becoming stable. The least weight obtained by the EE function is 1226.56 kN.

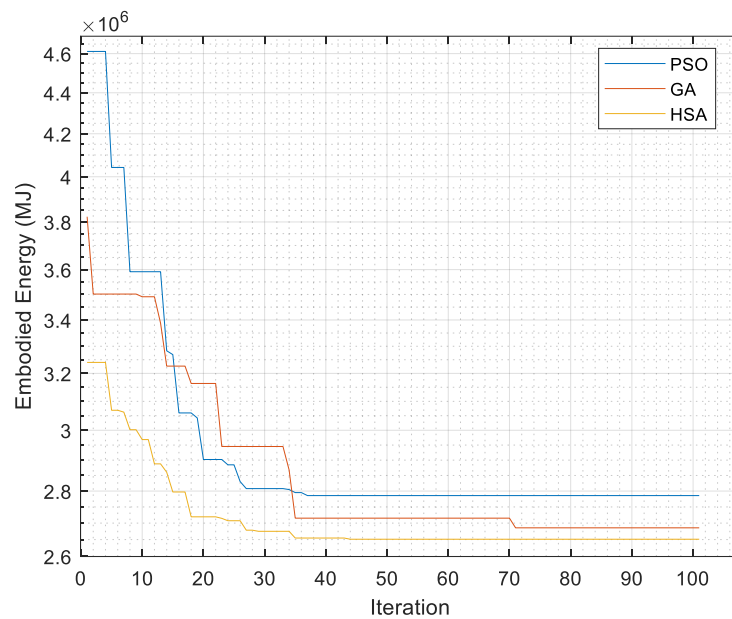


Figure 6-13: Comparison of Search history between GA, PSO and HSA for the six-storey structure (EE-objective function)

Table 6.6: Optimisation output for the six-storey structure (EE-objective function)

Element	Section		
	GA	PSO	HSA
Column (1)	356 x 368 x 153	356 x 406 x 235	305 x 305 x 97
Column (2)	254 x 254 x 73	305 x 305 x 97	203 x 203 x 71
Column (3)	356 x 368 x 202	305 x 305 x 158	356 x 368 x 153
Column (4)	254 x 254 x 73	305 x 305 x 97	203 x 203 x 71
Beam (5)	610 x 229 x 113	610 x 229 x 113	686 x 254 x 125
Beam (6)	254 x 102 x 28	152 x 89 x 16	152 x 89 x 16
Total optimum weight of the structure (kN)	1243.46	1290.74	1226.56
Total optimum Embodied Energy of the structure (MJ)	2685573	2785655	2651043
Total optimum Embodied Carbon of the structure (kgCO _{2,e})	214897	222984	212059
Total cost of the structure (£)	202883.70	208187.90	202452.00
Number of iteration for best result	71	37	44
Time of finding the best (sec)	172.48	31.56	148.48
Time of running (sec)	243	47	337

2- Multi-objective function:

Figure 6-14 gives the comparison of the convergence history for the multi-objective optimisation of the frame using the three algorithms and conventional design. The necessary parameters were set after setting different parameters as trials for each algorithm. The parameters used by each algorithm for the multi objective optimisation are;

GA: 200 generation size, 250 population, 100 Elite counts and 0.8 cross over function;

PSO: 200 iteration size and 200 swarm size;

HSA: had 200 iteration, 300 population size, 0.85 for HMCR, 0.3 for PAR and 0.3 also for BW

It is observed that an increase in the parameters used for the single objective function works better for GA while HSA required an increase in the HMS.

The lowest values as observed from the figure were obtained by HSA, while GA obtained the highest values among the three algorithms. All the values obtained were far lesser than that by conventional design. HSA started off from the highest values while PSO and GA started off at the intermediate values. However, HSA was still able to adequately search the space to obtain the lowest while GA could not obtain the global optimum before terminating.

Table 6.7 depicts all the values of the functions studied as well as the time and iterations used. The highest EE and weight from the table considering the algorithms only are 2961GJ and 1374kN by GA. HSA obtained the lowest EE and weight as 2651GJ and 1226.56kN. From the

table, PSO used the least iteration number of 13 while HSA required the highest (49). Even though PSO started off close to the optimum values, it still required more time for finding the best (153.96secs) and even longer total running time (1015secs). Furthermore, GA gave the highest values for all the functions, it however required the lowest time (108.47secs) among the algorithms.

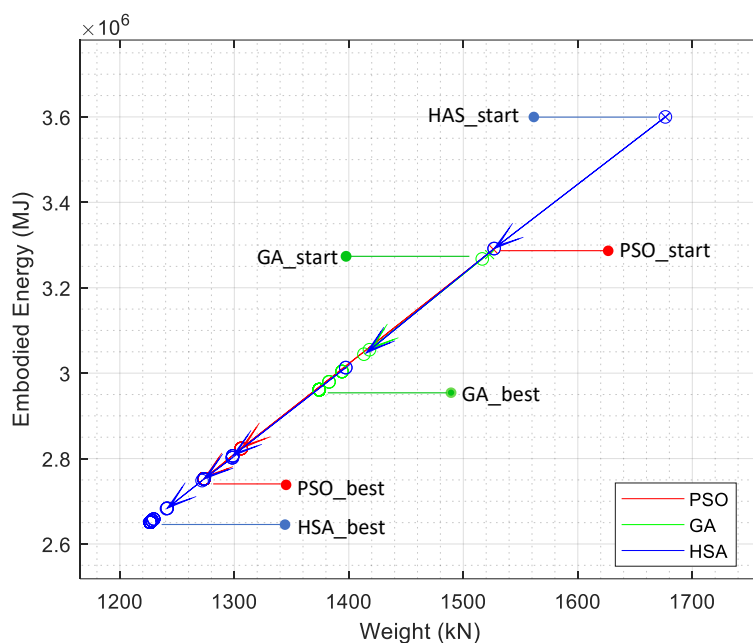


Figure 6-14: Comparison of Search history between GA, PSO and HSA for the six-storey structure (Multi-objective function)

Table 6.7: Optimisation output for the six-storey structure (Multi-objective function)

Element	Section		
	GA	PSO	HSA
Column (1)	356 x 368 x 129	305 x 305 x 97	305 x 305 x 97
Column (2)	254 x 254 x 73	203 x 203 x 71	203 x 203 x 71
Column (3)	356 x 406 x 467	356 x 368 x 153	356 x 368 x 153
Column (4)	305 x 305 x 97	305 x 305 x 118	203 x 203 x 71
Beam (5)	533 x 210 x 101	686 x 254 x 125	686 x 254 x 125
Beam (6)	356 x 127 x 33	254 x 102 x 25	152 x 89 x 16
Total optimum weight of the structure (kN)	1374.06	1273.58	1226.56
Total optimum Embodied Energy of the structure (MJ)	2961260	2752523	2651043
Total optimum Embodied Carbon of the structure (kgCO ₂ e)	237204	220181	212059
Total cost of the structure (£)	216610.30	210047.40	202452.00
Number of iteration for best result	15	13	49
Time of finding the best (sec)	108.47	153.96	171.78
Time of running (sec)	685	1015	343

3- Comparison

Figure 6-15 is a representation of the minimum weight obtained from each objective function minimization using the three algorithms. From the display, only HSA obtained the same result for multiobjective and EE objective functions. PSO also got a close result while the result for GA shows a wide gap between the two objective functions.

While the single objective function in GA obtained a lesser result, PSO obtained a better result in its multiobjective algorithm. This further shows that PSO can be better suited for multiobjective function than single objective function.

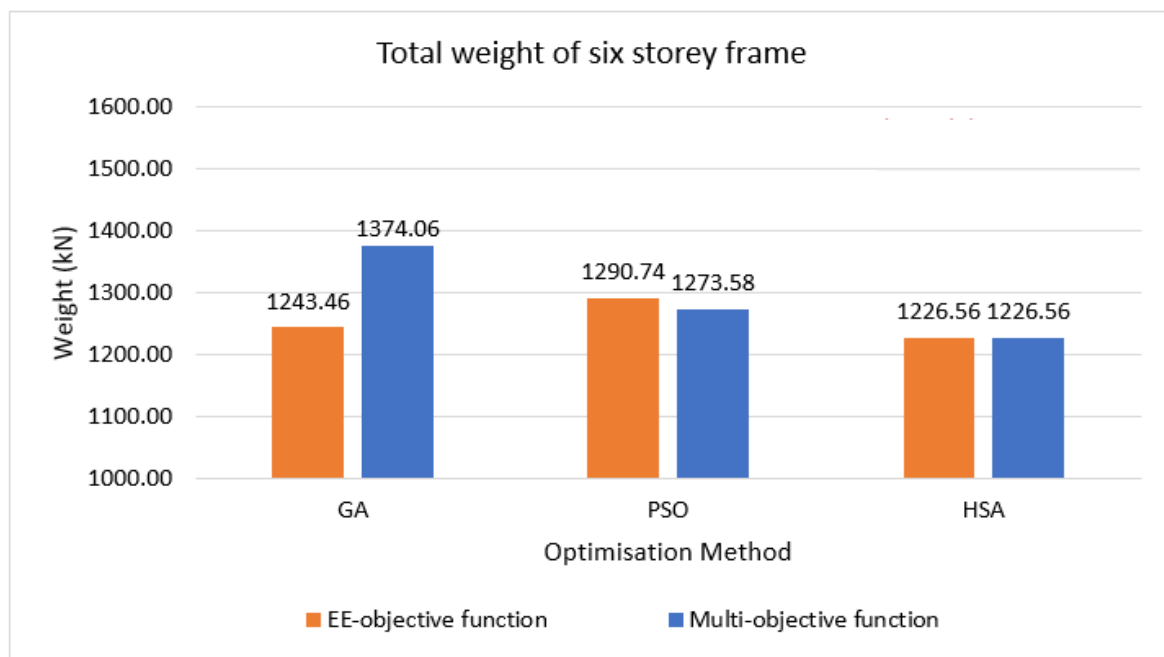


Figure 6-15: Schematic representation of the difference in weight among the three objective functions

The average running time for the algorithms for the two objective functions optimisation is denoted by Figure 6-16. The longest running time recorded for multi-objective optimisation is 17 minutes which is by PSO. The longest for single objective is 5.6 mins by HSA. From the figure, it can be deduced that the modulus operandi of the algorithms differs between the single and multi-objective optimisation. PSO is faster than the other two when computing single objective optimisation but the slowest for multi-objective optimisation. GA required 11 mins for its multi-objective function and 4 mins for single objective denoting a difference of 6mins. However, HSA has both algorithms using almost the same time (about 6mins). The longer time required by each algorithm for multi-objective optimisation confirms that in order for the

algorithm to obtain the best trade-off, more rigorous search is required i.e. it needs to properly search the design space, and thus requiring more iteration size and time. Considering the population size for effect on time, the increase in the population for the multi-objective function might have affected multi objective for GA but no real effect is observed for HSA. However, the same population is used for PSO but a difference of 16mins was observed. Furthermore, PSO had the lowest population size for the multi-objective function across all the three algorithms while the same population was used for the single objective functions. However, PSO was the slowest among the multi-objective function and the fastest for the single. Therefore, population size may be disregarded in relation to time used for running for PSO and HSA but may be a factor for GA.

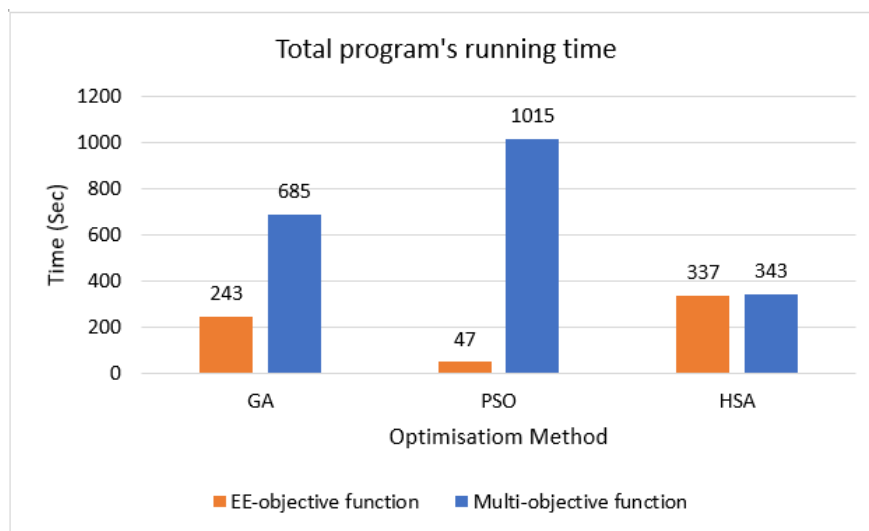


Figure 6-16: Running time average for the three algorithms for the three objective functions.

6.3.3 Three-bay nine-storey steel structure

Figure 6-17 describes the frame layout, member groupings, loadings and supports of a nine-storey three-bay space frame consisting. The frame consists of 360 members and 160 joints. From the figure, it is observed that there are 8 independent groupings of the members; group 1 represents the exterior columns in the first three storeys; Group 2 for the interior columns in the first three floors; group 3 represents the exterior columns for the middle three floors; Group 4 for the interior columns in the middle three storeys; Group 5 for the exterior columns in the upper three floors; Group 6 for the interior columns in the upper three floors; group 7 for the floor beams and group 8 for roof beams.

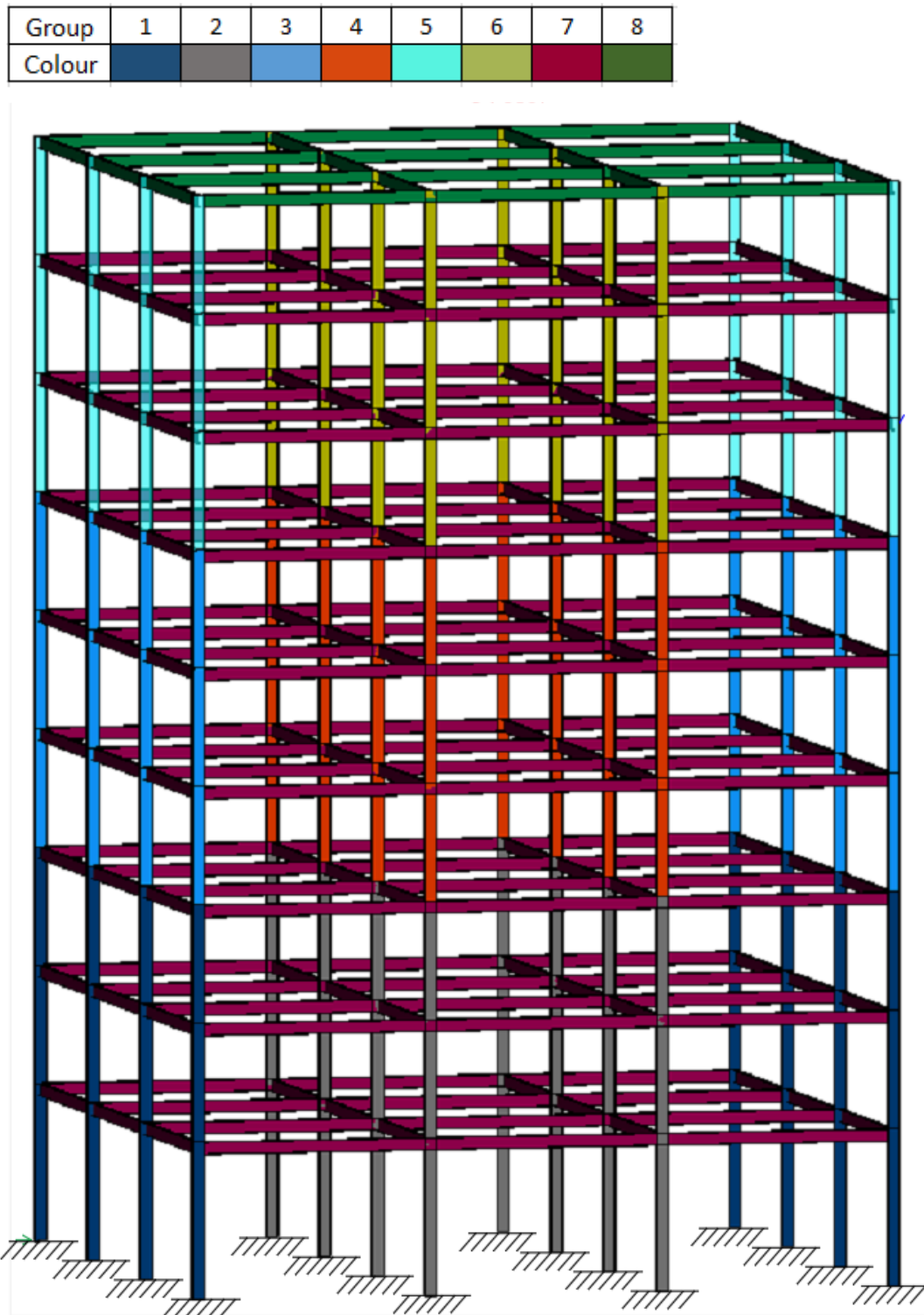


Figure 6-17: Representation of the layout and loading of the six-storey structure

1- Embodied Energy objective function:

Figure 6-18 shows the search history for embodied energy for the three optimisation algorithms and that for the conventional design. The figure was obtained after several runs for each algorithm. The parameters used for the algorithms are selected after many trials. The parameters used for GA are: 250 generation size, 250 population, 100 Elite counts and 0.8 cross over function. For PSO, the iteration size was also 200 while the swarm size was set to 250. And lastly, HSA had 250 iterations and 300 population size. The other parameters for HSA were: 0.85 for HMCR, 0.35 for PAR and 0.35 also for BW. From the search history, it can be observed that the first to reach its optimum was PSO followed by HSA and then GA.

Table 6.8 gives the result of the optimum values obtained by each algorithm for the weight, embodied energy, cost and embodied carbon. It also describes the number of iteration required, the time at which the optimum is found and the total time required for the optimisation. The lowest result for EE produced here is 4674GJ by HSA while PSO obtained the highest value of 4869GJ when considering the optimisation algorithms alone. While PSO required the least iterations (49) as well as time for running (86 secs), HSA consumed more time for running (623secs) at 61 iterations. GA happens to hold the middle of the spectrum of the three algorithms in terms of the obtained value and total time required, however, it required the highest iteration size (46) and used more time in finding the best before becoming stable. The least weight obtained is 2167.9kN.

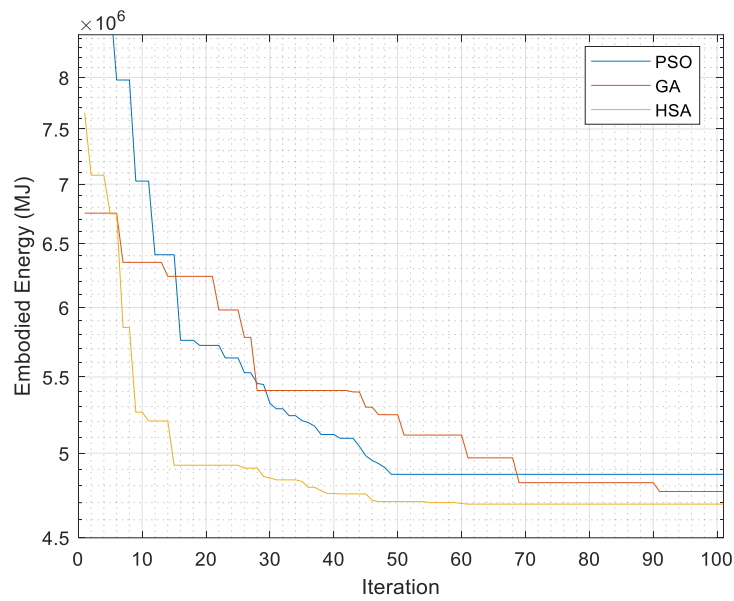


Figure 6-18: Comparison of Search history between GA, PSO and HSA for the nine-storey structure (EE-objective function)

Table 6.8: Optimisation output for the nine-storey structure (EE-objective function)

Element	Section		
	GA	PSO	HSA
Column (1)	254 x 254 x 89	356 x 368 x 129	305 x 305 x 118
Column (2)	356 x 368 x 129	356 x 368 x 129	356 x 368 x 129
Column (3)	305 x 305 x 97	305 x 305 x 97	254 x 254 x 73
Column (4)	305 x 305 x 97	254 x 254 x 89	305 x 305 x 97
Column (5)	254 x 254 x 107	356 x 368 x 129	203 x 203 x 71
Column (6)	254 x 254 x 73	254 x 254 x 73	254 x 254 x 73
Beam (7)	533 x 312 x 151	610 x 305 x 149	610 x 305 x 149
Beam (8)	178 x 102 x 19	356 x 127 x 33	178 x 102 x 19
Total optimum weight of the structure (kN)	2210.39	2256.61	2174.29
Total optimum Embodied Energy of the structure (MJ)	4766014	4869439	4691425
Total optimum Embodied Carbon of the structure (kgCO _{2,e})	381678	389815	375579
Number of iteration for best result	91	49	61
Total cost of the structure (£)	351270.40	363076.10	349381.70
Time of finding the best (sec)	386.37	62.70	379.91
Time of running (sec)	425	86	623

2- Multiobjective function:

Figure 6-19 gives the comparison of the convergence history for the multi-objective optimisation of the frame using the three algorithms and conventional design. The parameters used by each algorithm for the multi-objective optimisation after many trials were;

GA: 250 generation size, 300 population, 100 Elite counts and 0.8 cross over function;

PSO: 200 iteration size and 250 swarm size;

HSA: 300 iteration, 300 population size, 0.85 for HMCR, 0.35 for PAR and 0.35 for BW.

The difference observed from that of single objective is the increase in population size for GA and HSA. The convergence history shows that HSA obtained the lowest values while GA obtained the highest values. All the values obtained were lesser than that by conventional design.

From Table 6.9, the least EE and weight was 4674GJ and 2167.9kN. PSO obtained EE of 5295GJ while GA obtained EE of 5835GJ. While PSO and GA used very little iterations (7 and 10 respectively), HSA used a great deal of iterations (82). It however produced the least values with GA producing the highest values using the minimum time. The higher values obtained by the other two algorithms with small iteration size suggest that the algorithms got trapped in a local minimum and thus could not find the best function value.

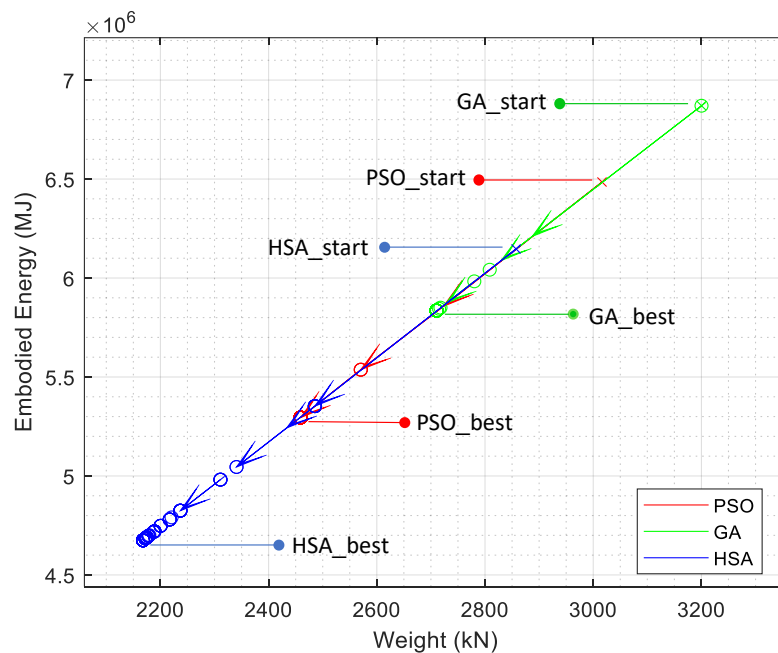


Figure 6-19: Comparison of Search history between GA, PSO and HSA for the nine-storey structure (Multi-objective function)

Table 6.9: Optimisation output for the nine-storey structure (Multi-objective function)

Element	Section		
	GA	PSO	HSA
Column (1)	356 x 368 x 129	356 x 368 x 129	305 x 305 x 97
Column (2)	356 x 406 x 509	305 x 305 x 283	356 x 368 x 129
Column (3)	254 x 254 x 107	203 x 203 x 127	254 x 254 x 73
Column (4)	356 x 368 x 202	254 x 254 x 132	254 x 254 x 89
Column (5)	305 x 305 x 137	203 x 203 x 127	203 x 203 x 71
Column (6)	305 x 305 x 97	254 x 254 x 89	254 x 254 x 73
Beam (7)	610 x 305 x 149	533 x 312 x 151	533 x 312 x 151
Beam (8)	457 x 152 x 60	356 x 127 x 33	178 x 102 x 19
Total optimum weight of the structure (kN)	2710.59	2459.26	2167.9
Total optimum Embodied Energy of the structure (MJ)	5835221	5294877	4674963
Total optimum Embodied Carbon of the structure (kgCO _{2,e})	467663	424331	374365
Number of iteration for best result	10	7	82
Total cost of the structure (£)	419717.5	381629.3	345183.7
Time of finding the best (sec)	58.13	108.66	256.65
Time of running (sec)	641	856	312

3- Comparison

Figure 6-20 is a representation of the minimum weight obtained from each objective function minimization using the three algorithms and conventional design. From the display, only HSA obtained the same result for the multiobjective and EE objective functions. PSO and GA obtained values in the single objective function that are greater than that by the multiobjective function.

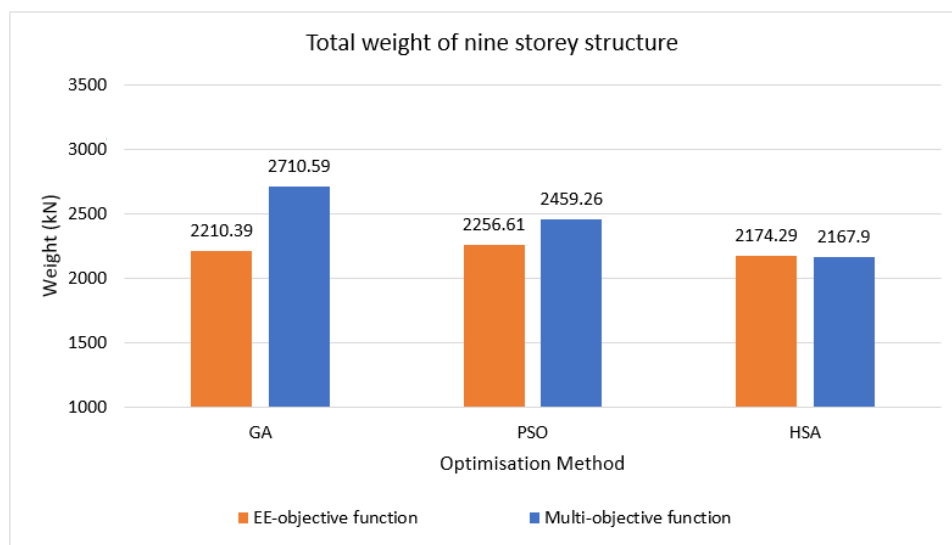


Figure 6-20: Schematic representation of the difference in weight among the three objective functions for nine storey frame

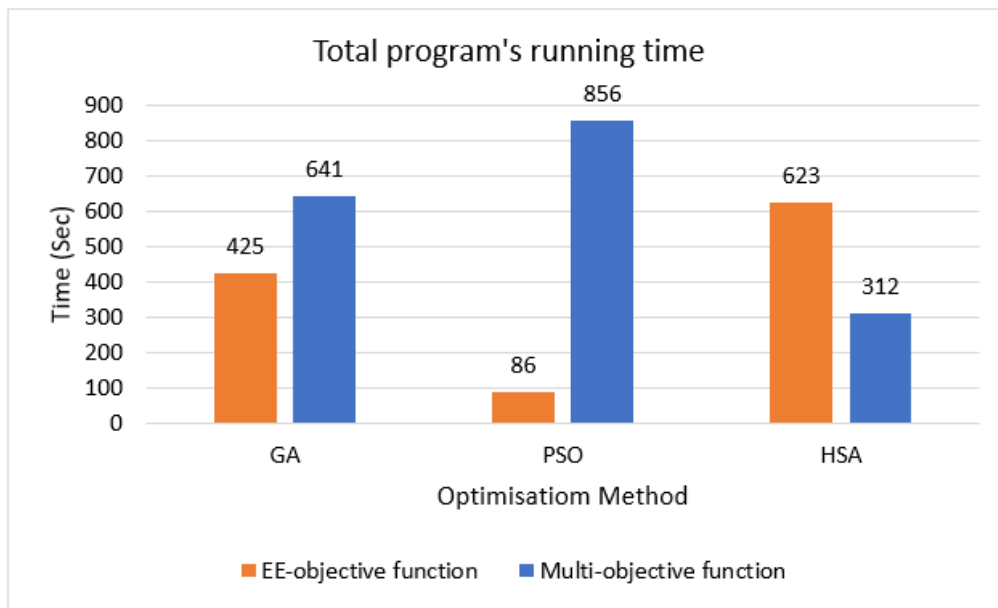


Figure 6-21: Running time average for the three algorithms for the three objective functions for nine storey frames.

The average running time for the algorithms for the two objective functions optimisation is denoted by Figure 6-21. The longest running time recorded for multi-objective optimisation is 14 minutes which is by PSO. The longest for single objective is 10mins by HSA. From the figure, it can be deduced that the modus operandi of the algorithms differs between the single and multi-objective optimisation. PSO is faster than the other two when computing single objective optimisation but the slowest for multi-objective optimisation. GA required 11mins for its multi-objective function and 7mins for single objective denoting a difference of 4mins. However, HSA in this case has the single objective using more time than the multi-objective with a difference of 5mins. The longer time required by GA and PSO algorithms for multi-objective optimisation suggests that in order for the algorithm to obtain the best trade-off, more rigorous search is required. However, the multi-objective functions here used lesser iterations for convergence. Considering the population size for effect on time, the increase in the population for the multi-objective function might have affected multi for GA but the reverse is the case for HSA. However, the same population is used for PSO but a difference of 13mins was observed. Furthermore, PSO had the lowest population size set for the multi-objective function across all the three algorithms while the same population was used for the single objective functions. However, PSO was the slowest for the multi and fastest for the single. Therefore, the correlation between population and time may only affect GA.

4- Discussion:

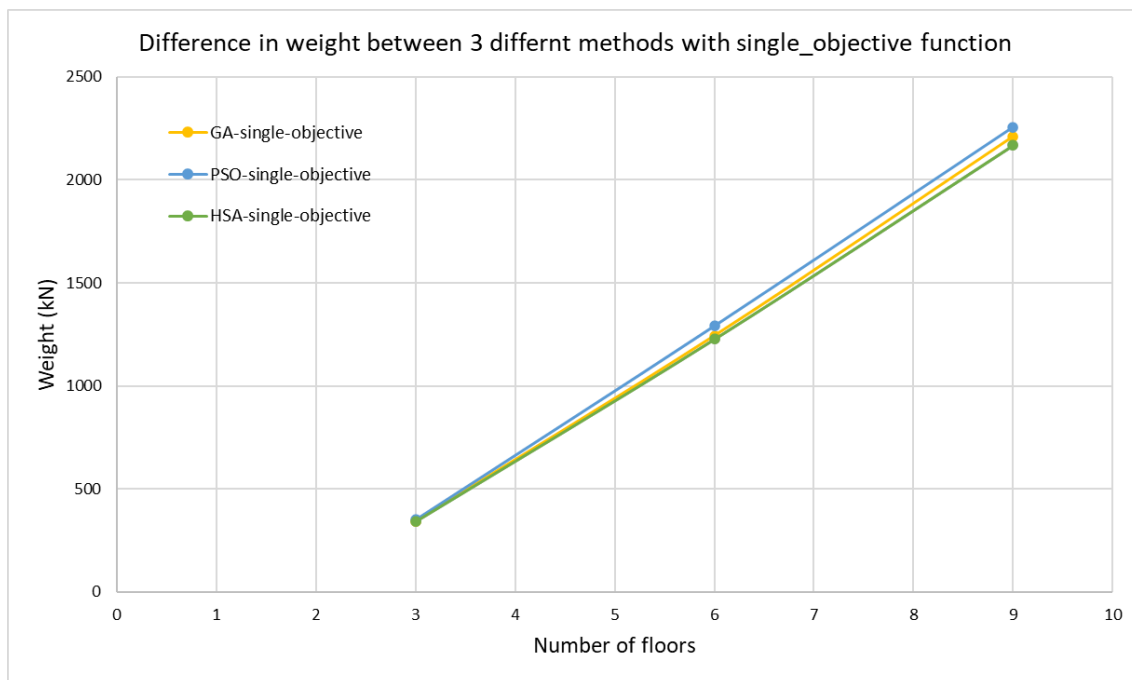


Figure 6-22: Comparison of weight obtained by three algorithms across the examples

Figure 6-22 depicts the difference in the weight obtained by each algorithm across the three examples. It shows the rate of increase in the values as we increase the problem size. From the foregoing discussion, it is evident that HSA is the most efficient of the three algorithms used in the study. Starting the comparison with the three storeys frame, GA and PSO algorithms respectively obtained results that are 1% and 2.4% above that of HSA. While in the 6 storeys frame the percentage difference increased to 1.4% and 5.2%. And lastly, that of 9 storey frame is 2% and 4.1% respectively for GA and PSO. It is observed that the difference between PSO and HSA is higher and much more pronounced in the 6 storey frame. However, the increase rate for GA is close together and the final increase is just 2%. This validates that the efficiency for GA is close to HSA for single objective optimisation.

Comparing this result with that of past studies, a similar trend is observed. That is, HSA obtained values that were higher than GA. According to the study by Degertekin and Hayalioglu (2010), 8.6%, 18.3% and 11.2% difference was observed between GA and HSA for 10 storey, 7 storeys and 9 storeys frames respectively. Although the no of bays differs between the three frames. Saka (2008) also obtained 0.11% and 14% difference between GA

and HSA for 6 storey and 15 storey frames respectively. The result thus obtained here only further validates the efficiency of HSA over GA.

In contrast, most studies comparing GA and PSO found PSO to be better in optimisation than GA. For instance, Mohammadi et al. (2019) concluded in their study through three examples that PSO was faster and better in convergence than GA in optimisation. This research thus discovered otherwise and found GA to be better in single optimisation. There are however not much studies on comparison of HSA and PSO for steel-framed structures.

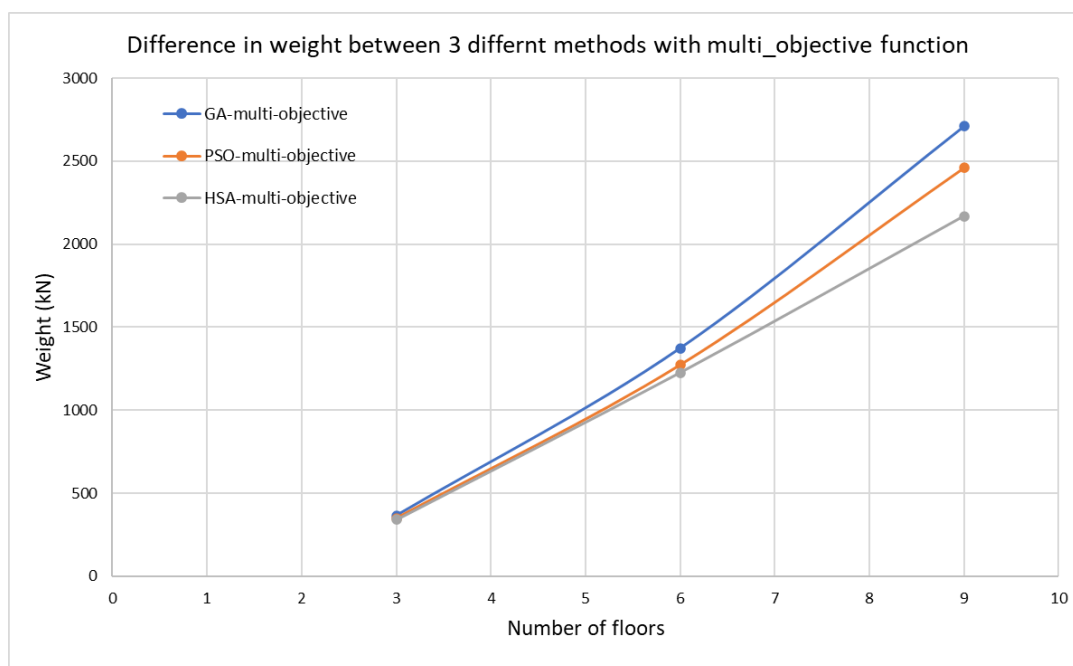


Figure 6-23: Comparison of weight obtained from multiobjective optimisation by the algorithms for the three examples

In the case of the multiobjective optimisation Figure 6-23, the reverse is the case for GA and PSO. PSO obtained values that are closer to HSA with percentages of 2.6%, 3.8% and 13.4% respectively for the three frames. However, an obvious gap is observed for GA with percentage difference of 7.1%, 12% and 25% respectively. The increase rate for PSO between the three and 6 storey frame is quite close together while a very sharp increase is observed for the 9 storey frame. The increase rate for GA are wide apart and even more for the 9 storey frame. It can be concluded that while PSO has a good efficiency when compared to HSA, the efficiency starts to reduce at more complex problems. It also follows that the efficiency of GA is reduced when using it for multiobjective optimisation with even less capacity for larger problems.

The efficiency of PSO for multiobjective optimisation matches that of Plevris et al. (2018) where both single and multi-objective optimisation were compared. They found the multi-objective to give better results.

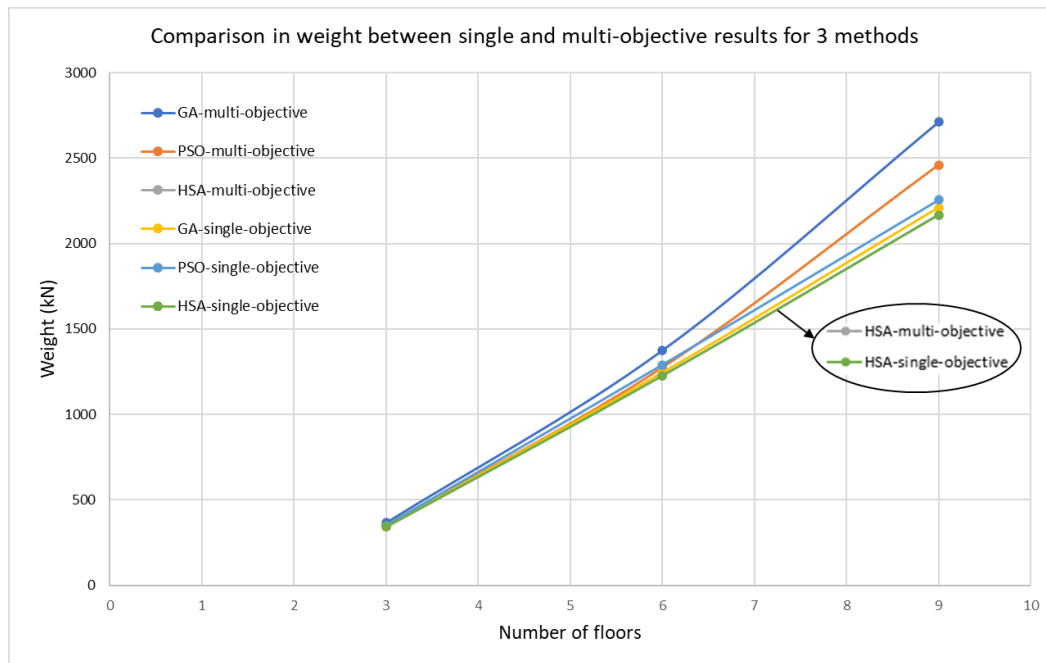


Figure 6-24: Comparison in weight between single and multi-objective optimisation for the three algorithms across the three examples.

Figure 6-24 describes the contrast in the optimum weight obtained in the two objective functions optimisation across the three frames by each algorithm. For HSA, no difference was observed across the three frames. That is, the algorithm obtained the same result for both single and multi-objective functions. PSO, on the other hand recorded, 0.2% difference for both 3 and 6 storey frames while 9% difference was observed for the 9-storey frame. Finally, the difference recorded for GA was high starting from the simplest of the problems (6%) and kept up doubling until the most complex (23%). This result further reinstates what has been discussed earlier as regards the efficiency of GA and PSO for multi-objective optimisation.

6.4 Analysis of life cycle assessment

The results of the cost optimization and the corresponding cost distribution of the specified cost contributions, obtained for the selected parameters, are shown in Figure 6-25. The main contribution to the total cost of the steel structure is represented by the steel consumption

(61%), followed by the erection (21%) and painting costs (14%). The transportation costs (4%), where the distance travelled was considered as 100 km. The calculated overall costs per mass of the structure is 1.65 £/kg.

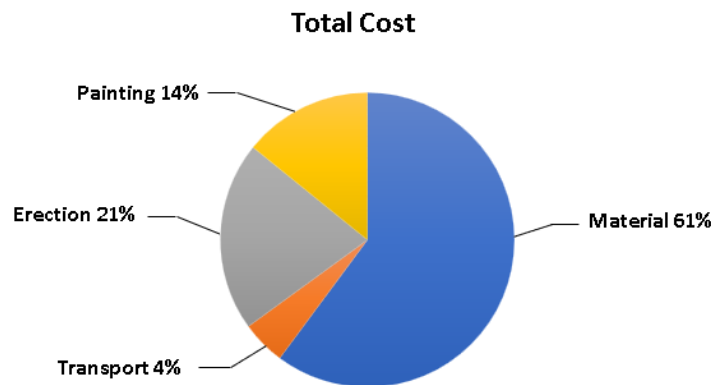


Figure 6-25: Cost distribution as a result of cost optimization of six-storey, three-bay steel structure

The life cycle assessment of embodied energy and carbon emissions, during the material manufacturing and construction phase, are illustrated in Figure 6-26. The results indicate that the energy consumption and carbon emissions in the material manufacturing stage accounts for 91.5% and 93% of total embodied energy and carbon emissions, respectively. The proportion of the energy consumption and carbon emissions of painting (fire protection + corrosion protection) accounts for 2.5% and 1.6% of the total consumption and emissions, respectively. The erection phase was responsible for 4.8% and 4.2%, followed by the transportation phase, which only accounts for 1.2% of the total embodied energy and carbon emissions.

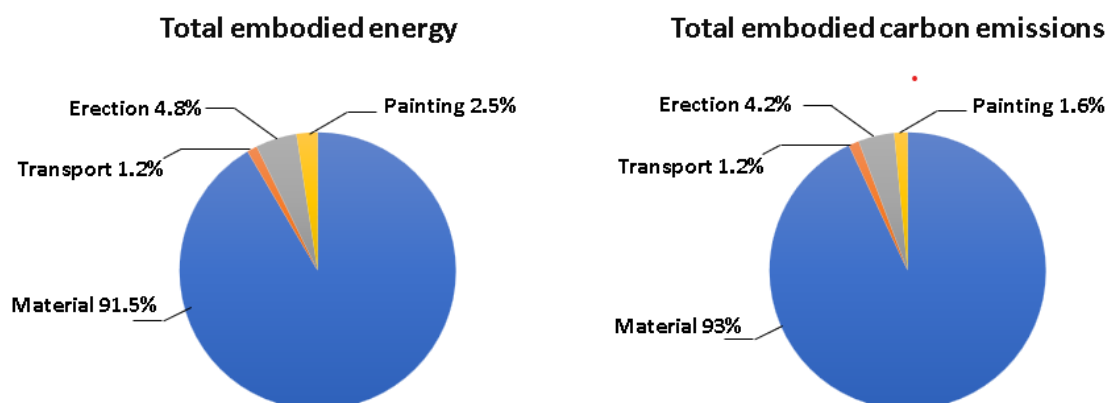


Figure 6-26: Distribution of embodied energy and carbon sources, during the production and construction stages.

6.5 Summary:

The optimum design of complex steel space frames can be a daunting task due to the presence of many non-linear constraints and discrete design variables. This chapter therefore proposed suitable algorithms that can effectively handle this task. Three common algorithms were used to investigate three numerical examples of steel space frames. These algorithms were initially validated for two-dimensional frames before proceeding to the 3-D frames. Comparisons of the algorithms for both single and multi-objective optimisation were carried out. The optimisation results included the time required, the minimum weight, embodied energy, embodied carbon and cost. The results obtained in all cases demonstrated that HSA was the most suitable for producing lighter designs as well as frames with low embodied energy. It was able to give the same result for its multi-objective and single objective optimisation. GA was closer to its efficiency for single objective optimisation while PSO works better for multi-objective optimisation. HSA is also fast enough for multi-objective optimisation.

CHAPTER SEVEN. PARAMETRIC STUDY

7.1 Introduction

This chapter involves an extensive analysis and discussion of the results obtained from a parametric study conducted considering only one objective function (embodied energy) and calculating the weight, cost and embodied carbon for three-dimensional 6 storey frames meant for different uses.

The parametric analysis carried out in this research investigated the effect of imposed loading intensity, steel grade, and different bay spacing in both X and Z directions. The structure considered, has a fixed length of 30 m in both X and Z directions to give a plan area of 900 m². The plan of the building for one of the bays arrangement is represented by Figure 7-1. The fixed and variable parameters set for the cases are given below in Table 7.1 and Table 7.2.

The validation chapter preceding this chapter established the efficiency and robustness of HSA method above GA and PSO. As such, it was the only algorithm adopted for the parametric analysis. The parameters adopted for the optimal cases for each group are the same after proper checks. The iteration size and harmony memory size are respectively 200 and 300. The HMCR, PAR and BW are 0.8, 0.3 and 0.3 respectively.

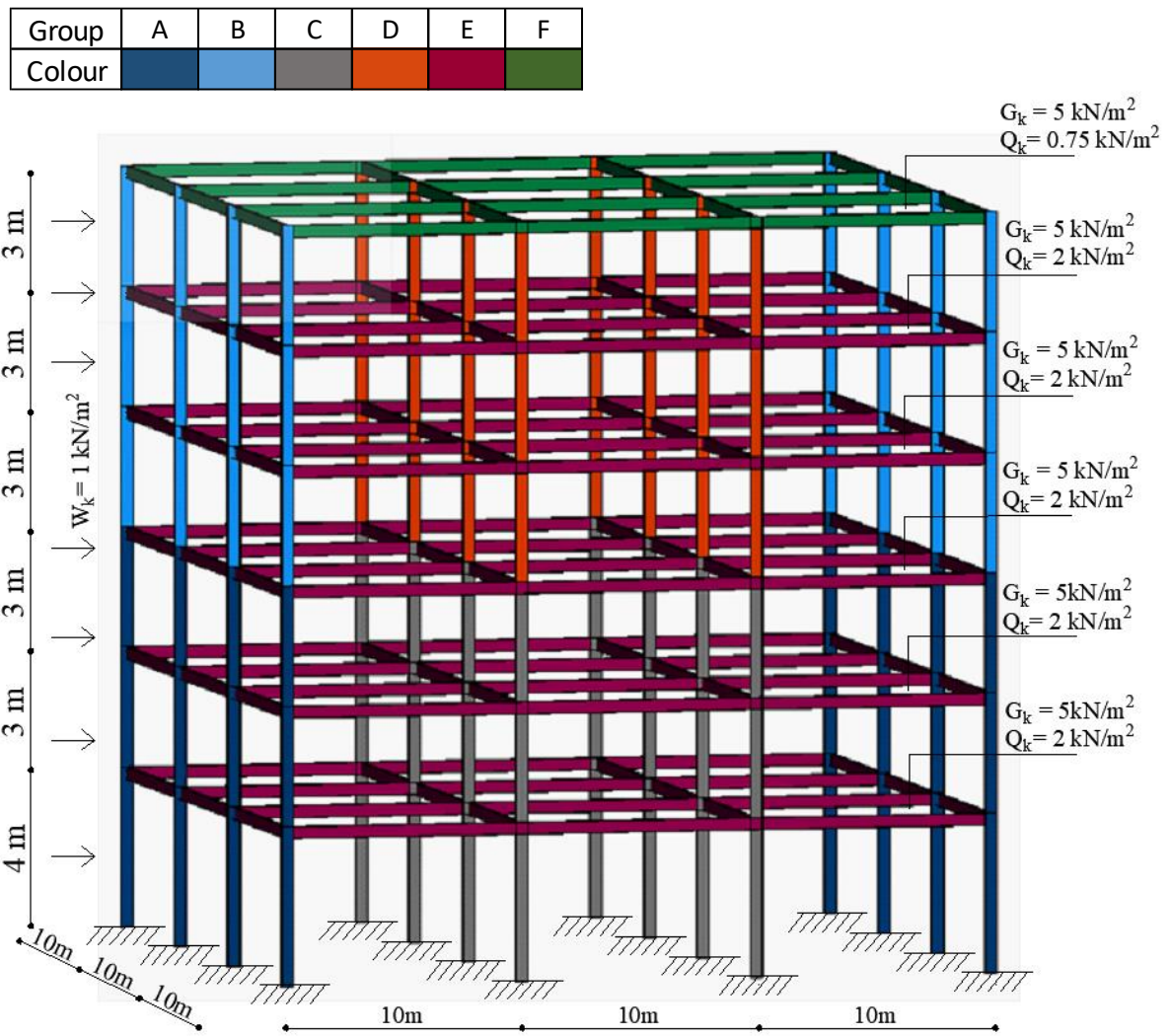


Figure 7-1: Layout and loading of the six-storey structure with (3X3) bay

Table 7.1: Fixed parameters

Fixed parameters		
Section	Value	Unit
Structure number of stories	6	stories
Height of ground floor	4	m
Floor height	3	m
Total height of the structure	19	m
Floor permanent action	5	kN/m ²
Roof permanent action	5	kN/m ²
Roof variable action	0.75	kN/m ²
Wind load	1	kN/m ²
Structure plan area	900	m ²
Structure length in X-direction	30	m
Structure length in Z-direction	30	m

Table 7.2: Variable parameters

Variable parameters	
Section	Value
Floor variable action	2, 4 and 6 kN/m ²
Steel grade	S355 & S275
No of bays in X-direction	3,4,5 and 6
No of bays in Z-direction	3,4,5 and 6
Cases studied <i>No of bays in X × No of bays in Z</i>	3 X 3
	3 X 4
	3 X 5
	3 X 6
	4 X 4
	4 X 5
	4 X 6
	5 X 5
	5 X 6
	6 X 6

The total number of cases investigated is 60 and thus were divided into 6 groups. A brief explanation of the groupings is given as follows:

Group 1: This group describes all the bay variations mentioned in Table 7.2 when the variable action is 2 kN/m² and the steel grade is high strength steel (S355).

Group 2: This group describe the bay variations when an increase in load to 4 kN/m^2 is used with a steel grade of S355.

Group 3: A representation for all the bay variations subjected to 6 kN/m^2 loading and steel grade of S355.

Group 4: From here, a change in steel grade is investigated. The group describes all the bay variations mentioned in Table 7.2 when the variable action is 2 kN/m^2 and the steel grade is mild steel (S275).

Group 5: This group describes the bay variations when an increase in load to 4 kN/m^2 is used with a steel grade of S275.

Group 6: The group is a combination of all the bay variations subjected to 6 kN/m^2 loading and steel grade of S275.

Therefore, it is observed that there are three groups representing high strength steel and three groups for mild steel. Table 7.3 further highlights this.

The bay variation mentioned in Table 7.2 is further explained as follows. The first number represents a number of bays in the X direction while the second number represents a number of bays in the Z direction. It should be noted that the span length for each direction is respectively 10m, 7.5m, 6m and 5m for 3, 4, 5 and 6 no of bays. For example:

4X4 means 4 bays in both X and Z directions with 7.5 m span length of each bay.

3X6 means 3 bays in the X direction and 6 bays in the Z direction with 10 m and 5 m span lengths respectively.

Table 7.3: Details for the groups classification

Steel grade, S355			Steel grade, S275		
Group 1	Group 2	Group 3	Group 4	Group 5	Group 6
$Q_k=2$	$Q_k=4$	$Q_k=6$	$Q_k=2$	$Q_k=4$	$Q_k=6$
3 X 3	3 X 3	3 X 3	3 X 3	3 X 3	3 X 3
3 X 4	3 X 4	3 X 4	3 X 4	3 X 4	3 X 4
3 X 5	3 X 5	3 X 5	3 X 5	3 X 5	3 X 5
3 X 6	3 X 6	3 X 6	3 X 6	3 X 6	3 X 6
4 X 4	4 X 4	4 X 4	4 X 4	4 X 4	4 X 4
4 X 5	4 X 5	4 X 5	4 X 5	4 X 5	4 X 5
4 X 6	4 X 6	4 X 6	4 X 6	4 X 6	4 X 6
5 X 5	5 X 5	5 X 5	5 X 5	5 X 5	5 X 5
5 X 6	5 X 6	5 X 6	5 X 6	5 X 6	5 X 6
6 X 6	6 X 6	6 X 6	6 X 6	6 X 6	6 X 6
<p>Note:</p> <p>3X3 means 3 bays in both X and Z directions with 10 m span length of each bay.</p> <p>4X6 means 4 bays in X direction and 6 bays in Z direction with 7.5 m and 5 m span lengths respectively</p>					

7.2 Parametric results

The following section describes the result obtained for each group with figures of the search history and optimum values. For each group, the weight, embodied energy, embodied carbon and cost of the frames are given as well as the parameters per square metre (sqm).

7.2.1 Group 1: Steel structures with grade (S355) subjected to variable action of 2 kN/m²

This section presents the optimum values for the low-intensity loading for the frame using steel grade S355. The low load intensity represents buildings intended for residential or light living. Table 7.4 highlights the optimum result obtained for weight, embodied energy, embodied carbon and cost. The time required for running the program and the parameters per square metre are also represented.

Table 7.4: Case details and optimum results for group 1

Imposed action, $Q_k = 2\text{kN/m}^2$												
Steel grade S355												
Group 1	Case No	No of Bays in X-dir	No of Bays in Z-dir	Weight (kN)	Embodied Energy (MJ)	Embodied carbon (kgCO ₂ e)	Cost (£)	Time of running (sec)	Weight per sqm (kN/m ²)	Embodied Energy per sqm (MJ/m ²)	Embodied carbon per sqm (kgCO ₂ e/m ²)	Cost per sqm (£/m ²)
	1	3	3	2766.1	5952479	477152	425545	332	0.512	1102.311	88.361	78.805
	2	3	4	2776.6	5986871	479447	441278	398	0.514	1108.680	88.786	81.718
	3	3	5	2900.7	6259607	501092	467029	500	0.537	1159.186	92.795	86.487
	4	3	6	2975.8	6426773	514277	485181	648	0.551	1190.143	95.236	89.848
	5	4	4	2631.5	5679272	454612	424413	552	0.487	1051.717	84.187	78.595
	6	4	5	2550.0	5513556	440953	423357	700	0.472	1021.029	81.658	78.399
	7	4	6	2550.6	5517388	441164	426383	936	0.472	1021.739	81.697	78.960
	8	5	5	2374.4	5134001	410589	394456	970	0.440	950.741	76.035	73.047
	9	5	6	2223.2	4822691.7	385091.4	387814.7	950	0.412	893.091	71.313	71.818
10	6	6	2070.0	4490696	358574	361340	1156	0.383	831.610	66.403	66.915	

Figure 7-2 gives the schematic representation of the optimum embodied energy for each case studied under group 1. The optimal case which is the 6 bays in both X and Z direction was also represented. From the figure, the optimum case which is the 6X6 case has a value of 4490GJ while the highest (3X6) has 6426GJ as the optimum embodied energy. A percentage difference of 30% in energy savings can be observed.

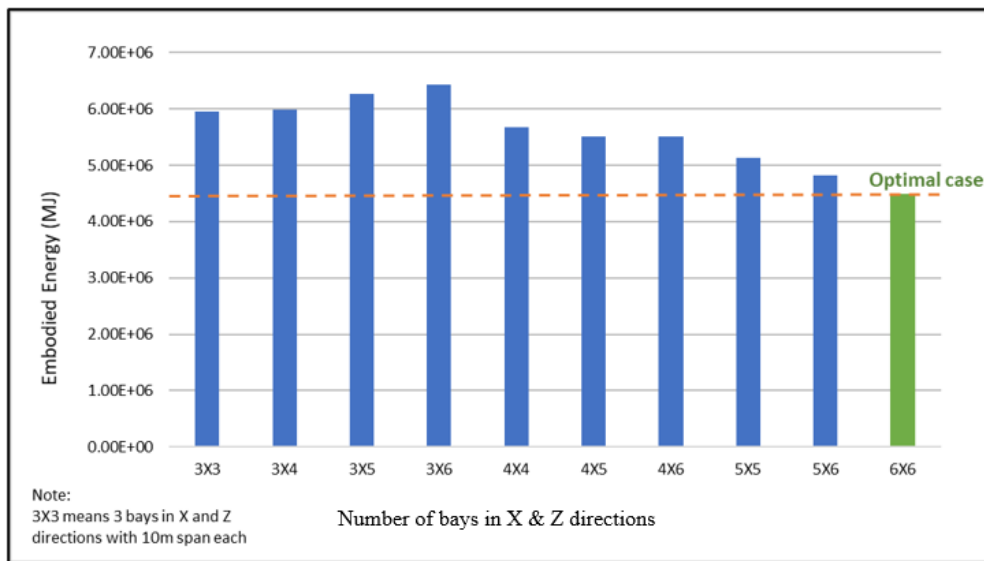


Figure 7-2: Optimum embodied energy for group 1

The search history for the optimum case in group 4 achieved in 4 runs is given by Figure 7-3. It can be observed that the search finally stabilized at 188 iteration size.

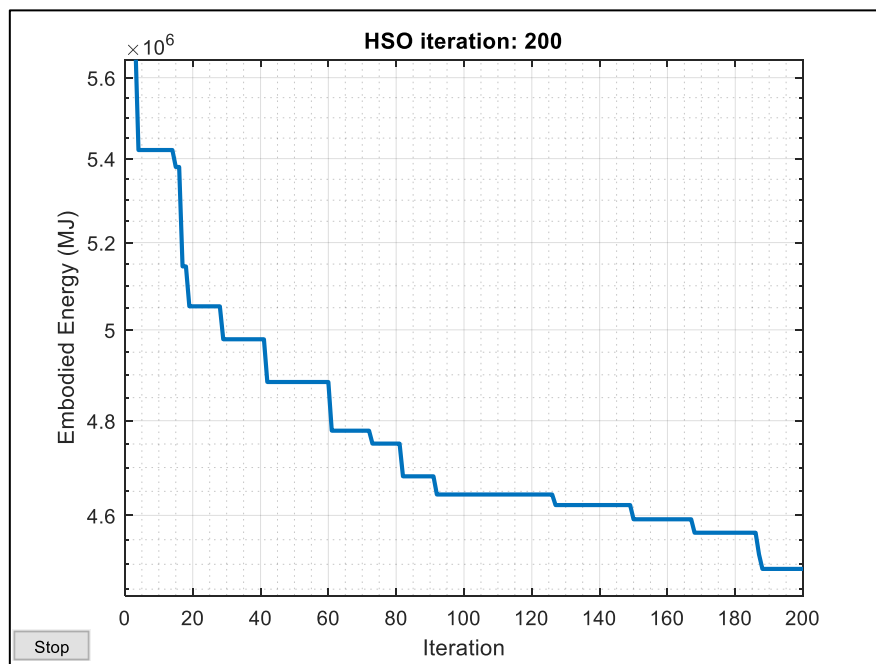


Figure 7-3: Search history for the optimal case in group 1

Table 7.5 shows the optimum UKB and UKC sections for the optimal case in group 1. This table also gives the expected total weight, cost, embodied energy and carbon of the structure. That is, considering all the frames that are expected in the structure altogether. The time for finding the best value as well as the total time taken to complete the algorithm is also given.

Table 7.5: Optimisation output for the optimal case in group 1

Element	Optimum Result
Column (A)	254 x 254 x 73
Column (B)	203 x 203 x 60
Column (C)	356 x 368 x 129
Column (D)	203 x 203 x 60
Beam (E)	356 x 171 x 57
Beam (F)	127 x 76 x 13
Total optimum weight of the structure (kN)	2070.04
Total optimum Embodied Energy of the structure (MJ)	4490696
Total optimum Embodied Carbon of the structure (kgCO ₂ e)	358574
Total cost of the structure (£)	361340.10
Number of iteration for best result	188
Time of finding the best (sec)	1098
Time of running (sec)	1156

7.2.2 Group 2: Steel structures with grade (S355) subjected to variable action of 4 kN/m²

This section presents the optimum values for the medium intensity loading for the frame using steel grade S355. The medium load intensity represents buildings intended for office use and the likes. Table 7.6 highlights the optimum result obtained for weight, embodied energy, embodied carbon and cost. The time required for running the program and the parameters per square metre are also represented.

Table 7.6: Case details and optimum results for group 2

Imposed action, $Q_k = 4\text{kN/m}^2$												Steel grade S355	
Group 2	Case No	No of Bays in X-dir	No of Bays in Z-dir	Weight (kN)	Embodied Energy (MJ)	Embodied carbon (kgCO ₂ e)	Cost (£)	Time of running (sec)	Weight per sqm (kN/m ²)	Embodied Energy per sqm (MJ/m ²)	Embodied carbon per sqm (kgCO ₂ e/m ²)	Cost per sqm (£/m ²)	
	1	3	3	3099.7	6662222	534360	467224	329	0.512	1233.745	98.956	86.523	
	2	3	4	2776.6	5986871	479447	441278	398	0.514	1108.680	88.786	81.718	
	3	3	5	2987.1	6444554	515954	479219	503	0.553	1193.436	95.547	88.744	
	4	3	6	2982.5	6441744	515454	486922	651	0.552	1192.916	95.454	90.171	
	5	4	4	2658.5	5737640	459280	428882	540	0.492	1062.526	85.052	79.423	
	6	4	5	2620.3	5658348	452806	426605	722	0.485	1047.842	83.853	79.001	
	7	4	6	2578.4	5578252	446003	431904	905	0.477	1033.010	82.593	79.982	
	8	5	5	2415.5	5226182	417836	405153	968	0.447	967.811	77.377	75.028	
	9	5	6	2303.3	4984952	398494.6	387662	924	0.427	923.139	73.795	71.789	
10	6	6	2123.3	4604218	367719	368168	1120	0.393	852.633	68.096	68.179		

Figure 7-4 gives the schematic representation of the optimum embodied energy for each case studied under group 2. The optimal case which is the 6 bays in both directions was also represented. From the figure, the optimal case has a value of 4604GJ while the highest (3X6) has 6662GJ as the optimum embodied energy. A percentage difference of 31% in energy savings can be observed.

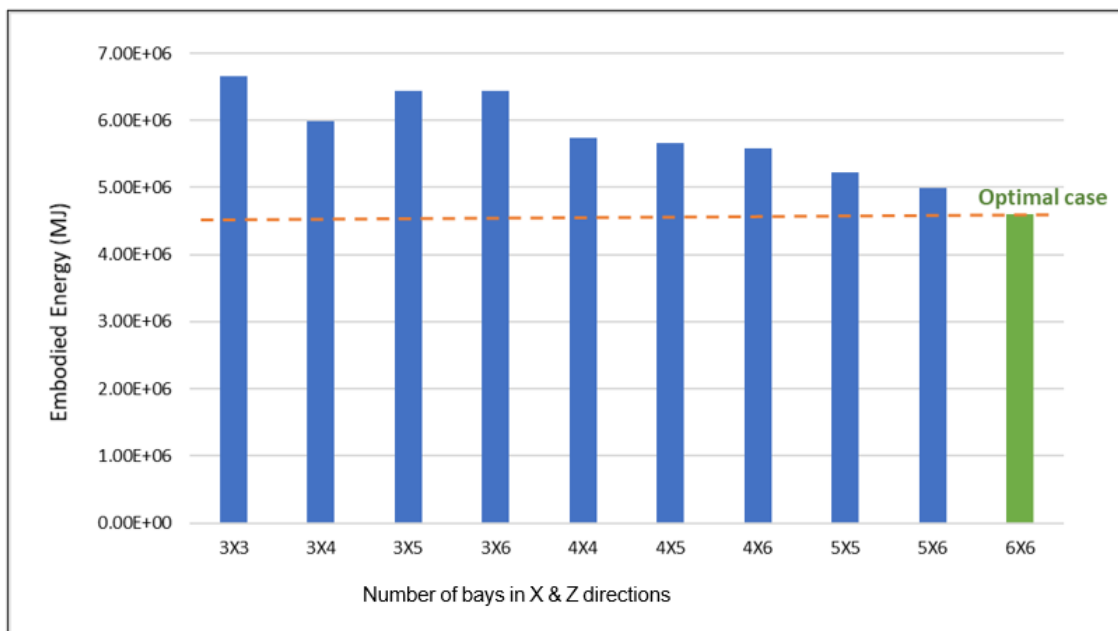


Figure 7-4: Optimum embodied energy for group 2

The search history for the optimum case in group 2 achieved in 5 runs is given by Figure 7-5. A stable search is observed at 179 iterations.

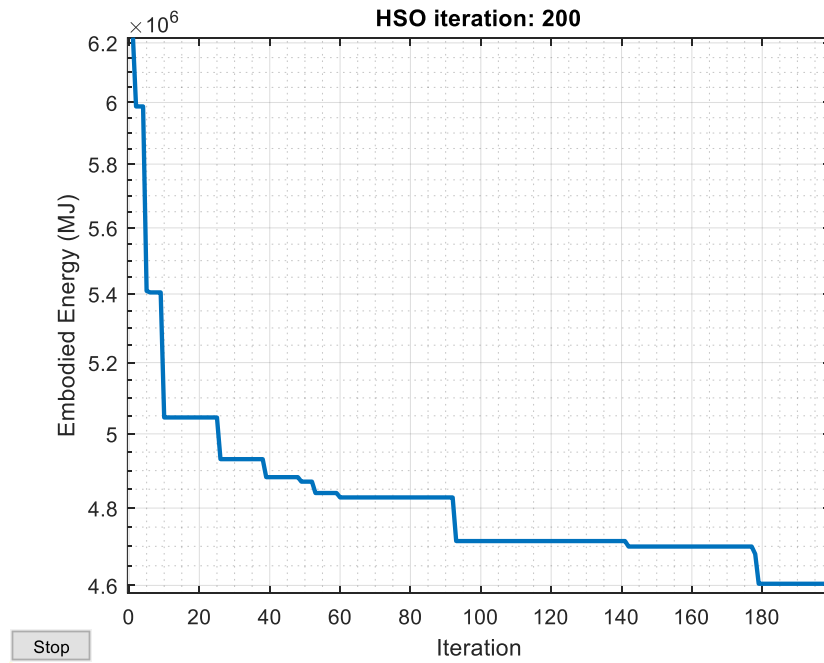


Figure 7-5: Search history for the optimal case in group 2

Table 7.7 shows the optimum UKB and UKC sections for the optimal case for group 2. Other parameters represented also include the total weight of the structure, total embodied energy total embodied carbon and cost. The time for both running and finding the best was also stated.

Table 7.7: Optimisation output for the optimal case in group 2

Element	Optimum Result
Column (A)	305 x 305 x 97
Column (B)	254 x 254 x 73
Column (C)	356 x 368 x 129
Column (D)	254 x 254 x 73
Beam (E)	305 x 165 x 54
Beam (F)	178 x 102 x 19
Total optimum weight of the structure (kN)	2123.33
Total optimum Embodied Energy of the structure (MJ)	4604218
Total optimum Embodied Carbon of the structure (kgCO ₂ e)	367719
Total cost of the structure (£)	368168
Number of iteration for best result	179
Time of finding the best (sec)	1008
Time of running (sec)	1120

7.2.3 Group 3: Steel structures with grade (S355) subjected to variable action of 6 kN/m²

This section presents the optimum values for the high-intensity loading for the frame using steel grade S355. The high load intensity represents buildings intended for commercial purposes like event halls or containing large crowds like library. Table 7.8 highlights the optimum result obtained for weight, embodied energy, embodied carbon and cost of the frame. The time required for running the program and the parameters per square metre are also represented.

Table 7.8: Case details and optimum results for group 3

Imposed action, $Q_k = 6\text{kN/m}^2$												Steel grade S355	
Group 3	Case No	No of Bays in X-dir	No of Bays in Z-dir	Weight (kN)	Embodied Energy (MJ)	Embodied carbon (kgCO ₂ e)	Cost (£)	Time of running (sec)	Weight per sqm (kN/m ²)	Embodied Energy per sqm (MJ/m ²)	Embodied carbon per sqm (kgCO ₂ e/m ²)	Cost per sqm (£/m ²)	
	1	3	3	3241.4	6964542	558689	486097	248	0.512	1289.730	103.461	90.018	
	2	3	4	2959.7	6373241	510715	460383	400	0.548	1180.230	94.577	85.256	
	3	3	5	2992.0	6454998	516796	479806	1298	0.554	1195.370	95.703	88.853	
	4	3	6	3027.0	6529914	522812	484874	737	0.561	1209.243	96.817	89.791	
	5	4	4	2770.3	5976338	478486	443894	534	0.513	1106.729	88.609	82.203	
	6	4	5	2658.2	5739967	459353	432332	701	0.492	1062.957	85.065	80.061	
	7	4	6	2590.0	5603842	448023	434620	897	0.480	1037.749	82.967	80.485	
	8	5	5	2441.9	5282248	422359	408332	990	0.452	978.194	78.215	75.617	
	9	5	6	2339.8	5064776	404840.9	394857	954	0.433	937.921	74.971	73.122	
	10	6	6	2149.4	4660898	372237	372938	1092	0.398	863.129	68.933	69.063	

Figure 7-6 gives the schematic representation of the optimum embodied energy for each case studied under group 3. The optimal case which is the 6 bays in both X and Z direction has been highlighted. From the figure, the optimum case has a value of 4661GJ while the highest (3X3) has 6965GJ as the optimum embodied energy. A percentage difference of 33% in energy savings can be observed by just changing the bay arrangement.

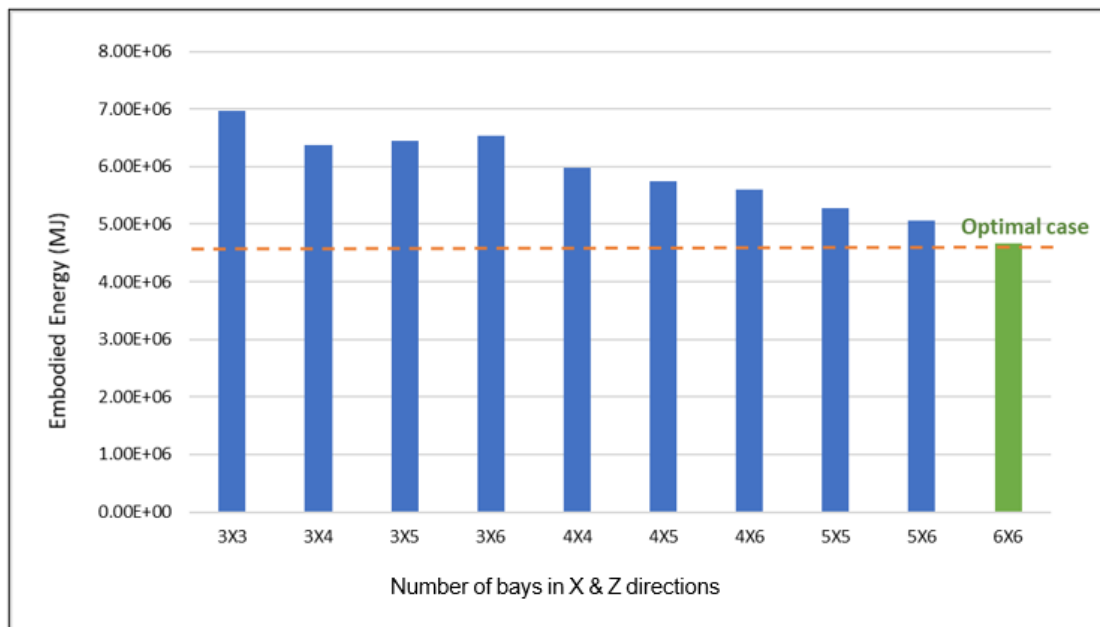


Figure 7-6: Optimum embodied energy for group 3

The search history for the optimum case in group 3 achieved in 6 runs is given by Figure 7-7. The algorithm found the best at 189 iterations.

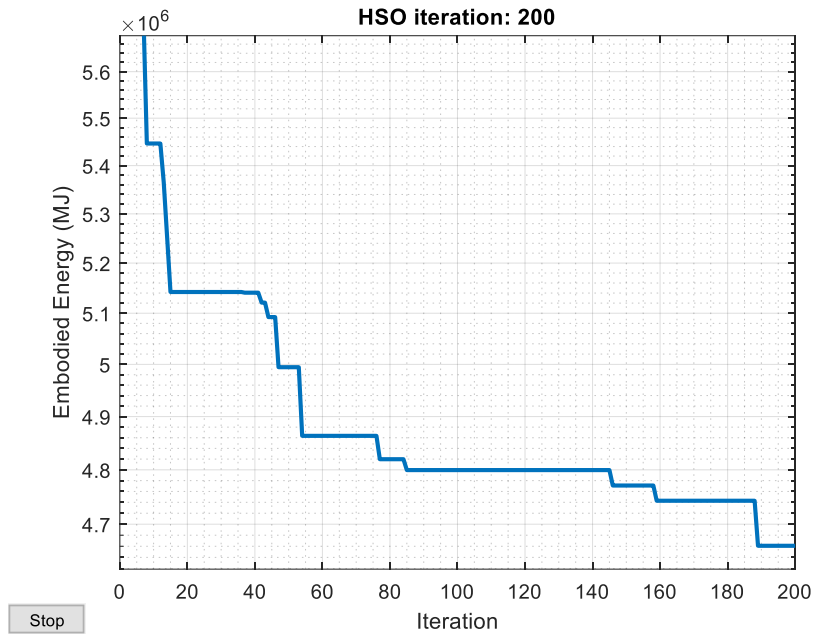


Figure 7-7: Search history for the optimal case in group 3

Table 7.9 shows the optimum UKB and UKC sections for the optimal case for group 1. Other parameters represented include the total weight of the structure, embodied energy, embodied carbon and cost. The required iterations, time of running and time of finding the best value was also given.

Table 7.9: Optimisation output for the optimal case in group 3

Element	Optimum Result
Column (A)	305 x 305 x 97
Column (B)	254 x 254 x 73
Column (C)	356 x 368 x 129
Column (D)	254 x 254 x 73
Beam (E)	305 x 165 x 54
Beam (F)	254 x 102 x 25
Total optimum weight of the structure (kN)	2149.37
Total optimum Embodied Energy of the structure (MJ)	4460898
Total optimum Embodied Carbon of the structure (kgCO ₂ e)	372237
Total cost of the structure (£)	372938
Number of iteration for best result	189
Time of finding the best (sec)	994
Time of running (sec)	1092

7.2.4 Group 4: Steel structures with grade (S275) subjected to variable action of 2 kN/m²

This section presents the optimum values for the low-intensity loading for the frame using steel grade S275. The low load intensity represents buildings intended for residential or light living. Table 7.10 highlights the optimum result obtained for weight, embodied energy, embodied carbon and cost. The time required for running the program and the parameters per square metre are also represented.

Table 7.10: Case details and optimum results for group 4

Imposed action, $Q_k = 2\text{kN/m}^2$			Steel grade S275									
Group 4	Case No	No of Bays in X-dir	No of Bays in Z-dir	Weight (kN)	Embodied Energy (MJ)	Embodied carbon (kgCO ₂ e)	Cost (£)	Time of running (sec)	Weight per sqm (kN/m ²)	Embodied Energy per sqm (MJ/m ²)	Embodied carbon per sqm (kgCO ₂ e/m ²)	Cost per sqm (£/m ²)
	1	3	3	3049.4	6557968	525848	464223	228	0.565	1214.439	97.379	85.967
	2	3	4	2904.4	6258453	501349	456894	360	0.538	1158.973	92.842	84.610
	3	3	5	3010.2	6498139	520101	487311	499	0.557	1203.359	96.315	90.243
	4	3	6	3091.0	6659680	533525	485200	844	0.572	1233.274	98.801	89.852
	5	4	4	2754.4	5934908	475447	432809	593	0.510	1099.057	88.046	80.150
	6	4	5	2599.0	5613079	449163	423785	704	0.481	1039.459	83.178	78.479
	7	4	6	2560.6	5538531	442870	427586	936	0.474	1025.654	82.013	79.183
	8	5	5	2429.2	5255816	420204	407462	1444	0.450	973.299	77.816	75.456
	9	5	6	2246.2	4868255.3	388901.8	386540.6	950	0.416	901.529	72.019	71.582
10	6	6	2091.5	4533123	362122	360153	1087	0.387	839.467	67.060	66.695	

Figure 7-8 gives the schematic representation of the optimum embodied energy for each case studied under group 4. The optimal case which is the 6 bays in both x and z direction has been highlighted. From the figure, the optimum case has a value of 4533GJ while the highest (3X6) has 6558GJ as the optimum embodied energy. A percentage difference of 31% in energy savings can be observed by just changing the bay arrangement.

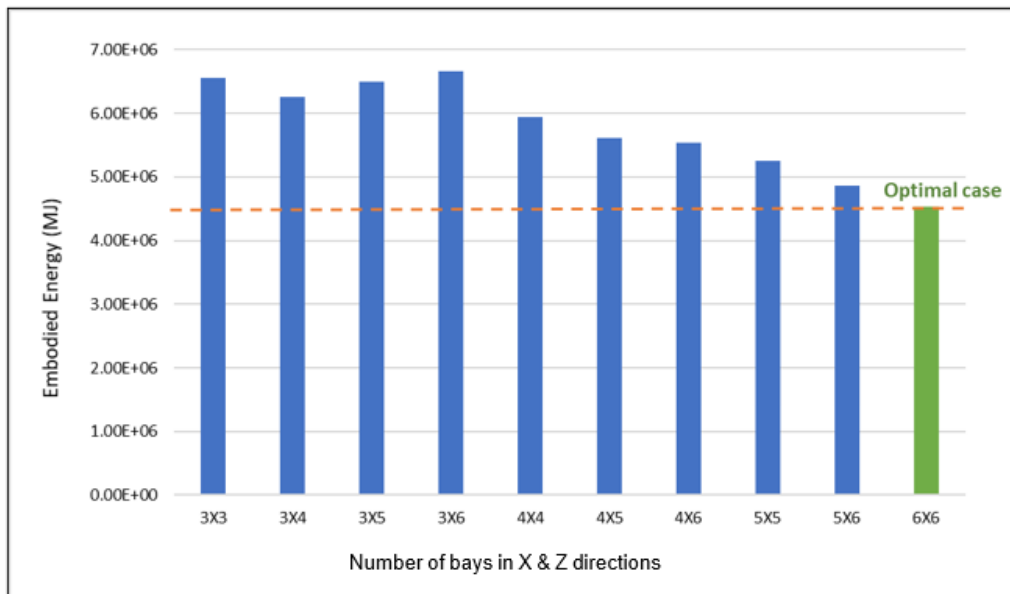


Figure 7-8: Optimum embodied energy for group 4

The search history for the optimum case in group 4 achieved in 4 runs is given by Figure 7-9. It is observed from the figure that the best value was arrived at after 140 iterations.

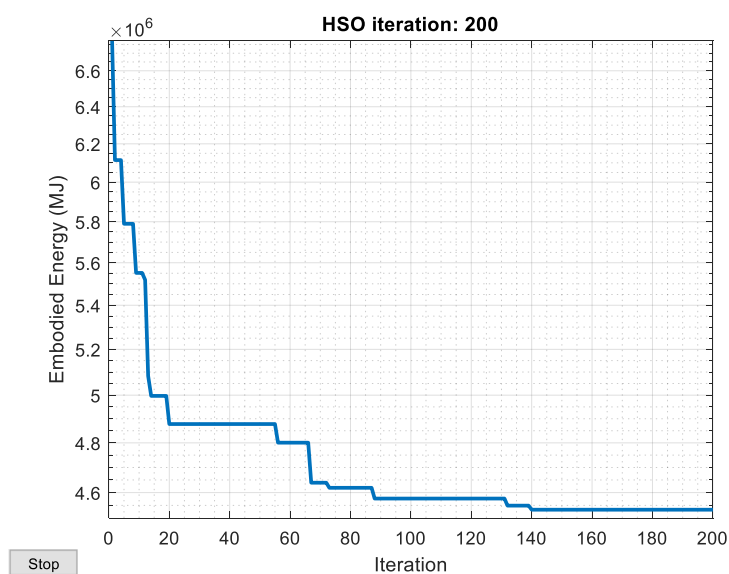


Figure 7-9: Search history for the optimal case in group 4

Table 7.11 presents the UKB and UKC sections for the optimal case for group 4. Other parameters stated include the total weight of the structure, embodied energy cost and embodied carbon. The minimum iteration required as well as time for the algorithm was also represented.

Table 7.11: Optimisation output for the optimal case in group 4

Element	Optimum Result
Column (A)	305 x 305 x 97
Column (B)	254 x 254 x 73
Column (C)	356 x 368 x 129
Column (D)	203 x 203 x 71
Beam (E)	305 x 165 x 54
Beam (F)	127 x 76 x 13
Total optimum weight of the structure (kN)	2091.52
Total optimum Embodied Energy of the structure (MJ)	4533123
Total optimum Embodied Carbon of the structure (kgCO ₂ e)	362122
Total cost of the structure (£)	360153
Number of iteration for best result	140
Time of finding the best (sec)	782
Time of running (sec)	1087

7.2.5 Group 5: Steel structures with grade (S275) subjected to variable action of 4 kN/m²

This section presents the optimum values for the low-intensity loading for the frame using steel grade S275. The medium load intensity represents buildings intended for offices and the likes. Table 7.12 highlights the optimum result obtained for weight, embodied energy, embodied carbon and cost. The time required for running the program and the parameters per square metre are also represented.

Table 7.12: Case details and optimum results for group 5

Imposed action, $Q_k = 4\text{kN/m}^2$										Steel grade S275		
Group 5	Case No	No of Bays in X-dir	No of Bays in Z-dir	Weight (kN)	Embodied Energy (MJ)	Embodied carbon (kgCO ₂ e)	Cost (£)	Time of running (sec)	Weight per sqm (kN/m ²)	Embodied Energy per sqm (MJ/m ²)	Embodied carbon per sqm (kgCO ₂ e/m ²)	Cost per sqm (£/m ²)
	1	3	3	3226.3	6932492	556107	484176	248	0.597	1283.795	102.983	89.662
	2	3	4	2965.7	6387256	511795	462615	362	0.549	1182.825	94.777	85.669
	3	3	5	3069.8	6613035	529830	480604	511	0.568	1224.636	98.117	89.001
	4	3	6	3048.2	6575897	526490	488373	679	0.564	1217.759	97.498	90.439
	5	4	4	2793.8	6018606	482198	437606	536	0.517	1114.557	89.296	81.038
	6	4	5	2672.3	5769799	461764	433883	708	0.495	1068.481	85.512	80.349
	7	4	6	2578.5	5577674	445982	431133	911	0.477	1032.903	82.589	79.839
	8	5	5	2426.3	5250257	419733	407804	1004	0.449	972.270	77.728	75.519
	9	5	6	2342.7	5077861	405624.2	403185	924	0.434	940.345	75.116	74.664
10	6	6	2159.6	4690031	374298	382910	1105	0.400	868.524	69.314	70.909	

Figure 7-10 describes the optimum embodied energy for each case studied under group 5. The optimal case which is the 6 bays in both x and z directionS has been highlighted. From the figure, the optimum case has a value of 4690GJ while the highest (3X3) has 6932GJ as the optimum embodied energy. A percentage difference of 32% in energy savings can be observed in just changing the bay arrangement.

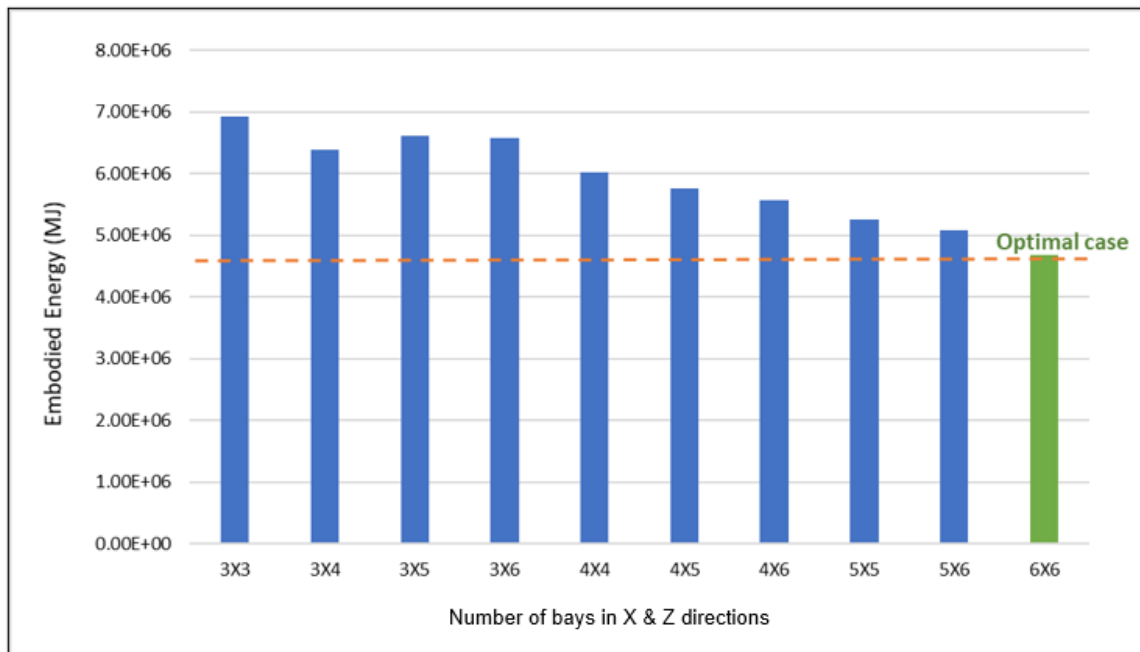


Figure 7-10: Optimum embodied energy for group 5

The search history for the optimal case in group 5 achieved in 6 runs is given by Figure 7-11. It can be observed that the search for the optimum became stable at 188 iterations.

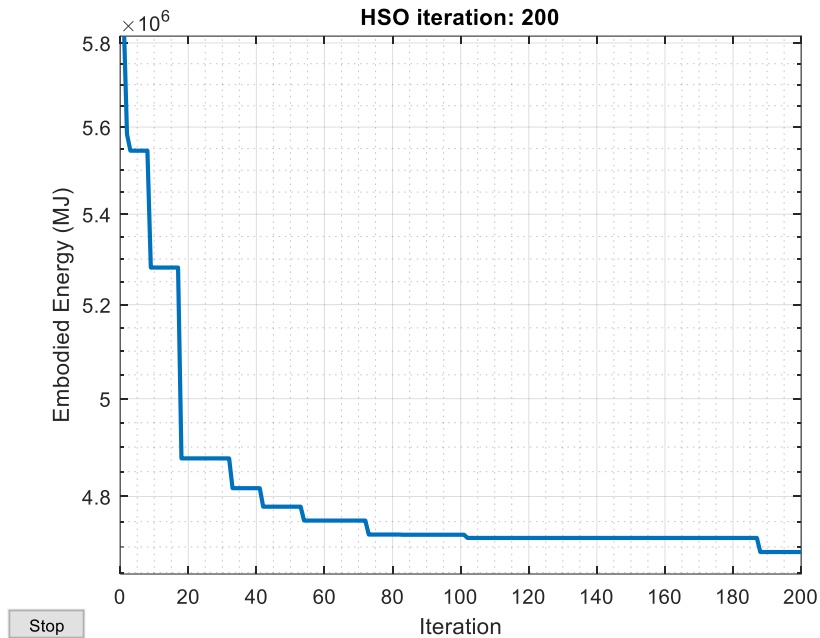


Figure 7-11: Search history for the optimal case in group 5

Table 7.13 states the optimum UKB and UKC sections for the optimal case for group 5. The total value for the structure of each parameters considered is also given as well as the time of running, time to find the best and the number of iterations.

Table 7.13: Optimisation output for the optimal case in group 5

Element	Optimum Result
Column (A)	305 x 305 x 118
Column (B)	305 x 305 x 97
Column (C)	356 x 368 x 129
Column (D)	254 x 254 x 73
Beam (E)	356 x 171 x 57
Beam (F)	127 x 76 x 13
Total optimum weight of the structure (kN)	2159.62
Total optimum Embodied Energy of the structure (MJ)	4690031
Total optimum Embodied Carbon of the structure (kgCO ₂ e)	374298
Total cost of the structure (£)	382910
Number of iteration for best result	188
Time of finding the best (sec)	1067
Time of running (sec)	1105

7.2.6 Group 6: Steel structures with grade (S275) subjected to variable action of 6 kN/m^2

The optimum values for the high-intensity loading for the frame using steel grade S275 are described in this section. The high load intensity represents buildings intended for commercial purposes. Table 7.14 highlights the optimum result obtained for weight, embodied energy, embodied carbon and cost. The time required for running the program and the parameters per square metre are also represented.

Table 7.14: Case details and optimum results for group 6

Imposed action, $Q_k = 6 \text{ kN/m}^2$										Steel grade S275		
Group 6	Case No	No of Bays in X-dir	No of Bays in Z-dir	Weight (kN)	Embodied Energy (MJ)	Embodied carbon (kgCO ₂ e)	Cost (£)	Time of running (sec)	Weight per sqm (kN/m ²)	Embodied Energy per sqm (MJ/m ²)	Embodied carbon per sqm (kgCO ₂ e/m ²)	Cost per sqm (£/m ²)
	1	3	3	3485.9	7484628	600617	516426	233	0.646	1386.042	111.225	95.635
	2	3	4	3121.4	6718051	538474	481592	374	0.578	1244.084	99.717	89.184
	3	3	5	3033.9	6543278	523945	484035	1300	0.562	1211.718	97.027	89.636
	4	3	6	3036.9	6551692	524539	486959	637	0.562	1213.276	97.137	90.178
	5	4	4	2859.7	6157958	493462	444853	534	0.530	1140.363	91.382	82.380
	6	4	5	2723.4	5878261	470514	440031	722	0.504	1088.567	87.132	81.487
	7	4	6	2695.4	5819324	465741	437225	958	0.499	1077.653	86.248	80.968
	8	5	5	2500.4	5407805	432441	416798	1024	0.463	1001.445	80.082	77.185
	9	5	6	2392.3	5182539	414098.4	408401	954	0.443	959.730	76.685	75.630
10	6	6	2197.6	4769271	380749	385730	1124	0.407	883.198	70.509	71.431	

Figure 7-12 describes the optimum embodied energy for each case studied in group 6.

The optimal case which is the 6 bays in both x and z directions has been highlighted. From the figure, the optimum case has a value of 4769GJ while the highest (3X3) has 7485GJ as the optimum embodied energy. A percentage difference of 36% in energy savings can be observed in just changing the bay arrangement.

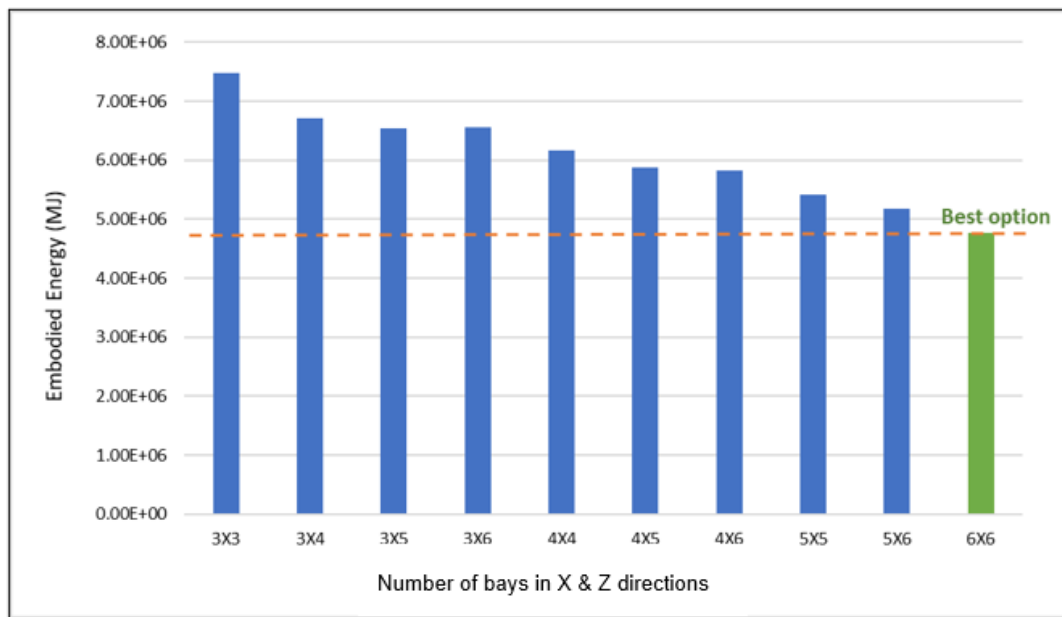


Figure 7-12: Optimum embodied energy for group 6

Figure 7-13 gives a schematic representation of the search history for the optimal case in group 6. The value achieved in 4 runs required 166 iterations before becoming stable.

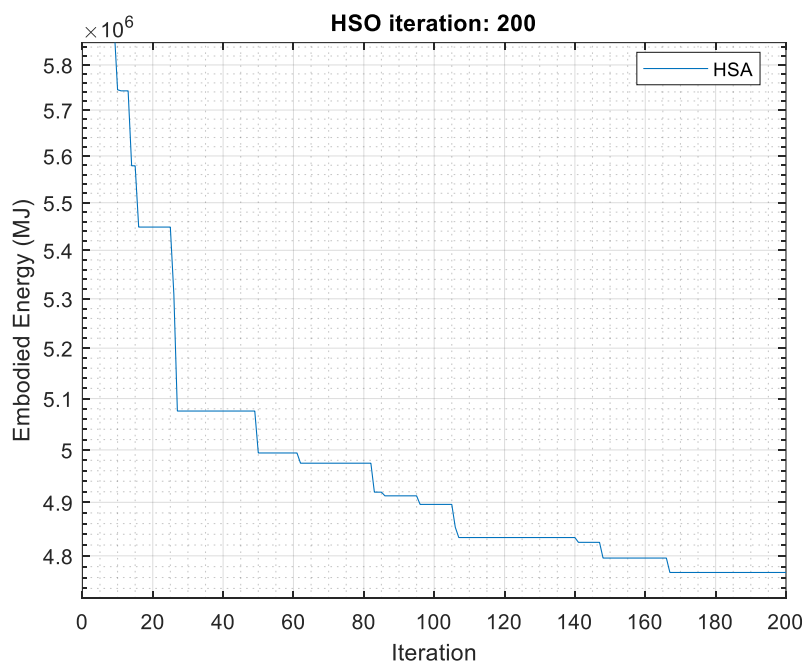


Figure 7-13: Search history for the optimal case in group 6

Table 7.15 presents the optimum UKB and UKC sections for the beams and columns of the optimal case for group 6. The total value for the structure of each parameter considered is also given as well as the time of running, time to find the best and the number of iterations.

Table 7.15: Optimisation output for the optimal case in group 6

Element	Optimum Result
Column (A)	356 x 368 x 129
Column (B)	305 x 305 x 97
Column (C)	356 x 368 x 129
Column (D)	305 x 305 x 97
Beam (E)	356 x 171 x 51
Beam (F)	152 x 89 x 16
Total optimum weight of the structure (kN)	2070.04
Total optimum Embodied Energy of the structure (MJ)	4490696
Total optimum Embodied Carbon of the structure (kgCO ₂ e)	358574
Total cost of the structure (£)	361340
Number of iteration for best result	188
Time of finding the best (sec)	895
Time of running (sec)	1124

7.3 Discussion:

7.3.1 Effect of Steel Grade

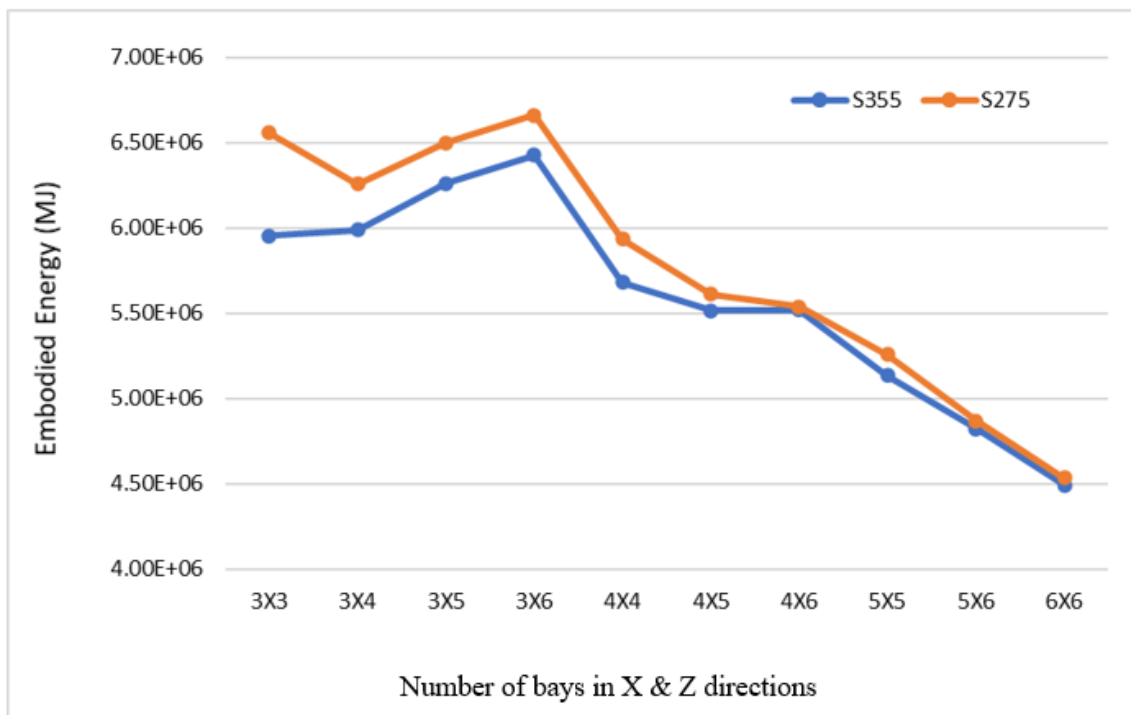


Figure 7-14: Comparison of optimum embodied energy between group 1 and 4

The difference in energy usage with different steel grades is described in Figure 7-14. This figure compares the groups with 2kN load using S355 and S275 steel. It is evident from the figure that S355 led to more energy savings. The highest percentage difference observed is 10.2% while the lowest is 0.4%. It is also observed that the percentage difference is more evident for cases with less number of bays in X and Z directions (especially X-direction) and the difference become lower with increasing number of bays. A good explanation for this is the lower number of steel sections required by the high strength steel grade (S355) as compared to the mild steel. That is, the sections with high strength steel are able to carry more load than those of mild steel. However, the lower contribution to the difference for frames with more number of bays means that increased number of frames required for the structure will lead to increased number of S355 steel used. Since it naturally has more embodied energy than S275 steel, it tends to cancel out the savings made by using it in lower quantity.

7.3.2 Effect of the variable actions

Figure 7-15 displays the percentage increase in embodied energy as the loading on the frames increased. It is expected that as the load increases, the required sections will also increase and consequently raise the embodied energy used. The percentage difference between 2kN/m² and 4kN/m² ranges from 0%-11.9% while for 2kN/m² and 6kN/m² is 1.6% -17%. That between 4kN/m² and 6kN/m² is 0.2%-6.5%. However, due to optimisation, some savings with the increasing load were still achieved.

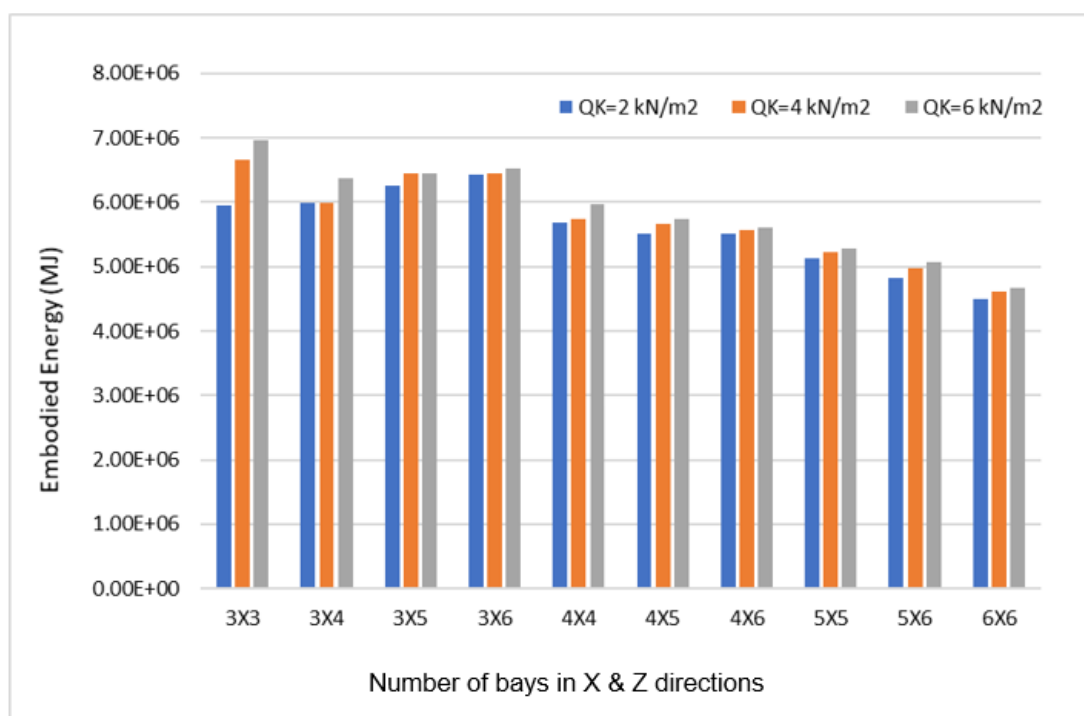


Figure 7-15: Comparison of optimum embodied energy for all cases in group 1, 2 and 3 when steel grade S355.

For example, case 3X4 has the same value of embodied energy for 2kN/m^2 and 4kN/m^2 . Case 3X5 almost has the same embodied energy for 4kN/m^2 and 6kN/m^2 (0.2% difference). This can be because the optimisation of the lesser load intensity used sections that were not at their full capacity but a lesser combination could not be obtained while the higher load intensity only used up the remaining capacity of the sections or added a little to it.

Furthermore, looking into the mode of increase across the cases, a varying trend is observed. For example, for all cases with 3 number of bays in the X-direction, an initial increase in percentage was observed to be 11.9 followed by 0, then 3 and finally 0.2% for 2kN/m^2 and 4kN/m^2 . The same is observed for those with 4 number of bays in the X-direction. A trend cannot be formed with the remaining two as they are not up to three. Considering the cases altogether, the trend also varies. Therefore, the embodied energy increase with increased loading intensity does not correlate with the bay arrangement in both directions.

It should however be noted that certain bay arrangements with a lower load intensity led to increase in embodied energy that is equal or almost equal to another bay arrangement with a higher load intensity. For example, the embodied energy for 3X5 case with 4kN/m^2 and 6kN/m^2 is the same as that for 3X6 case with 2kN/m^2 and 4kN/m^2 . Also, 4X4 case with 2kN/m^2 has a

higher embodied energy than that of 4X6 case with 6kN/m^2 and many more. It is obvious that reducing span length in the X-direction leads to decreasing values of embodied energy. However, due to the varying trend, a conclusion as to how others affect the embodied energy could not be made.

7.3.3 Optimum design graphs for the structure for different variable actions.

This section presents the parameters per square meter for the optimal case in groups 1, 2 and 3. That is, the values for increasing load intensities of S355 steel. In all the cases, the optimal case was found to be the 6X6 case. The optimum EE per square meter for the structure as shown in Figure 7-16 ranges from $831.6\text{--}863.13\text{ MJ/m}^2$ while the EC varies from $66.403\text{--}68.93\text{ KgCO}_2\text{e/m}^2$. These values are expected to guide designers and stakeholders on how much embodied energy and embodied carbon savings they should look out for when considering frames with 6 storeys or lower which falls within the loading intensities considered.

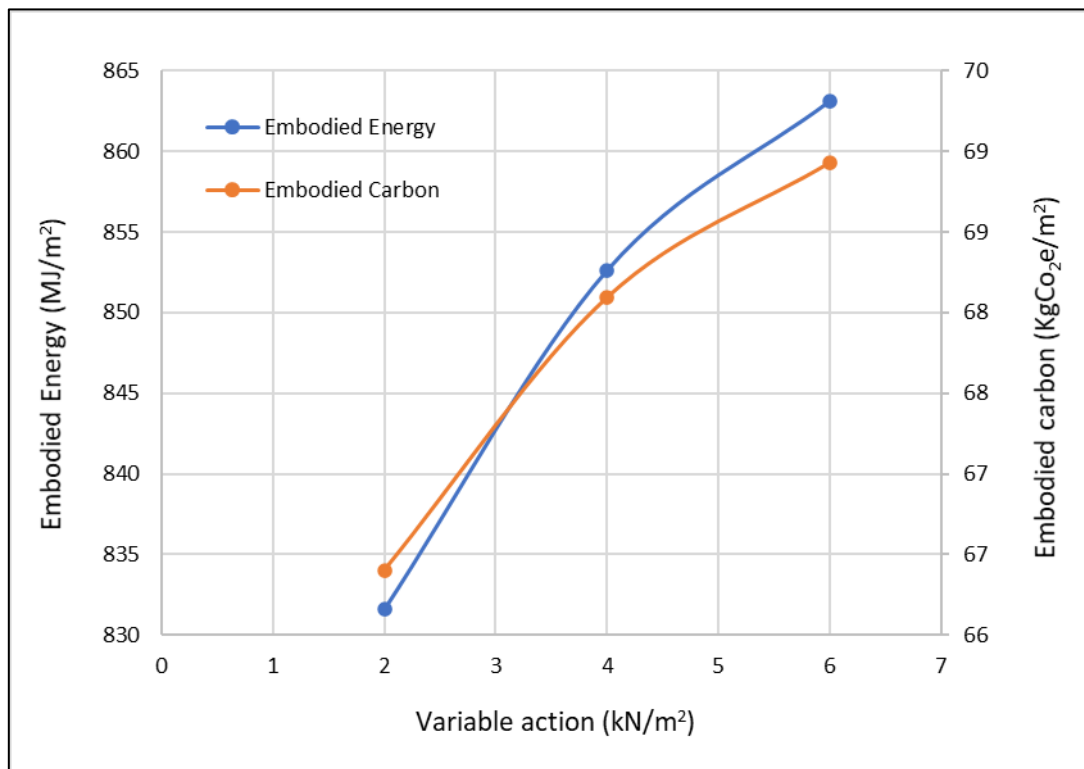


Figure 7-16: Embodied energy and Embodied carbon per sqm for the optimal cases 6X6 of group 1, 2 and 3 when steel grade is S355

Furthermore, as shown in Figure 7-17 the optimum weight per square meter for the structure with 6X6 number of bays in both directions range from 0.383 - 0.398 kN/m² while the cost varies from 66.92- 69.063 £/m². These also give an idea of the expected cost of similar structures and the weight to be passed to the foundation.

In both graphs 7-16 and 7-17, embodied energy and weight is observed to have a higher percentage increase than embodied carbon and cost respectively at the higher load intensities.

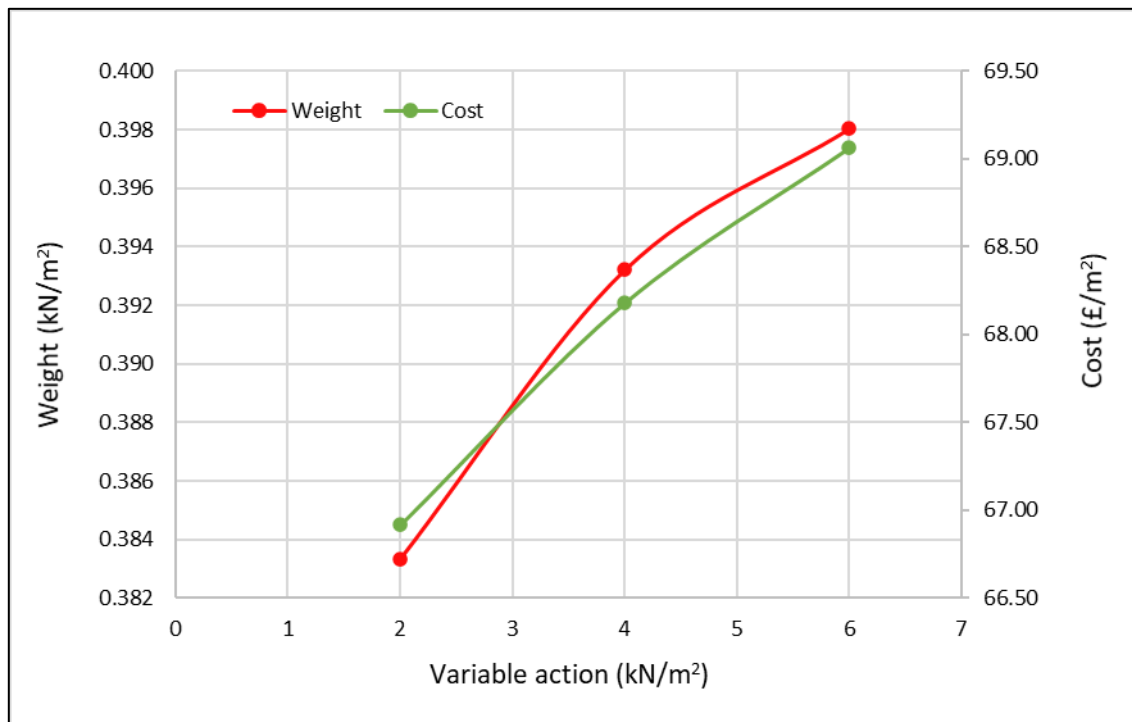


Figure 7-17: Weight and Cost per sqm for the optimal cases 6X6 of group 1, 2 and 3 when steel grade is S355

7.3.4 Running time for optimisation

The running time is related using only group 1 as a similar trend is observed across the groups. Figure 7-18 gives a representation of the running time required to give the optimum by each case in the group. It is observed that a progressive increase is depicted in the figure. That is, as the number of bays increases, the time required increases. This means, since the number of variables increased by increasing the number of cross-sections, the time to find the optimum will also increase. 3x3 (332 secs) has the lowest time while 6x6 (1156 secs) used the highest. Moreover, the program is expected to analyse, design and optimise the sections, it is expected that it would need more time for larger elements and combinations. Conclusively, for the

optimisation of a 6 storey frame, the minimum time required is almost 6mins while the maximum is 20mins. Therefore, a complete cross sections specification with varying options can be concluded within the time frame of discussing a structural engineering project.

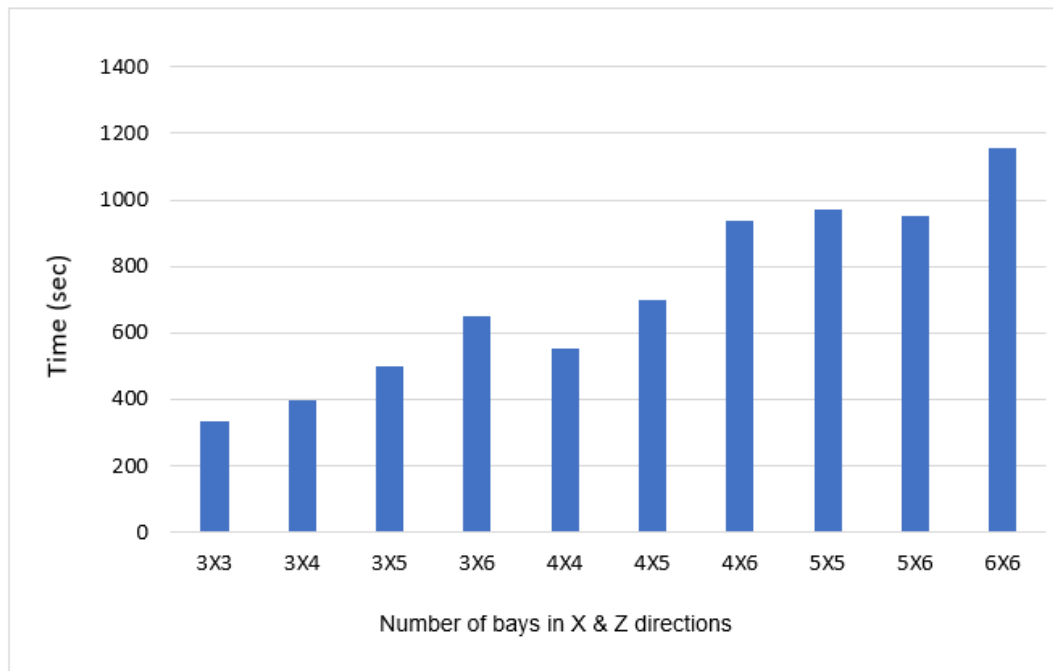


Figure 7-18: Computational time for the optimisation process of all the cases in group 1.

7.4 Summary

This chapter conducted a parametric study by investigating the effects of varying span lengths, variable actions and steel grades on the embodied energy of a three-dimensional 6 storey frame. The results presented included the weight, embodied energy, embodied carbon and cost of the individual frames and the whole structure. The result showed that lower span lengths in both directions can lead to 36% savings than using larger spans. It further revealed that using S355 steel grade can lead to more energy savings but the savings may not be evident with lower span lengths. The changing variable actions led to higher embodied energy for the frames but those with larger spans tend to exhibit more response. Figures and values provided for the frame as well as the parameters per sqm calculated are expected to guide designers on which bay arrangement to adopt for similar structures or how much embodied energy, cost, weight and embodied carbon they are going to use.

CHAPTER EIGHT. Conclusion and Suggestions for Future Work

8.1 Introduction

This research has presented a comprehensive study on the lifecycle optimisation of two and three-dimensional rigid steel structures using GA, PSO and HSA optimisation methods developed in MATLAB. Embodied energy, embodied carbon and cost models were developed for the implementation of the single and multi-objective functions optimisation of weight and embodied energy. The steel structures were designed to satisfy constraints based on EC3. The first section of this chapter summarises the main achievements in this research. The conclusions that have been drawn from the design optimisation approaches that have been developed are outlined in the second section. Finally, in the third section, certain recommendations and suggestions were made for possible directions of further works.

8.2 Contribution to knowledge

The significant achievements of this study can be summarised as follows:

- 1- Developing a comprehensive and thoroughly integrated program that combined MATLAB and excel to carry out design optimisation of 2d and 3d steel structures using three different optimisation techniques.
- 2- The developed code has the capability to carry out single and multi-objective optimisation, specifically the weight, the cost, and the embodied energy/embodied carbon.
- 3- The procedure of optimisation techniques, Genetic Algorithm (GA), PSO and HSA have been modified in order to carry out the solution of the optimisation problem in faster and more reliable outcome.
- 4- Two life cycle stages of the structure have been considered: production and construction stages, which include three boundaries: materials, transportation, and erection.
- 5- A parametric study has been carried out to investigate the effect of imposed loading intensity, steel grade, and different bay spacing in both direction of the structure.

- 6- The designed charts and tables produced as a result of extensive parametric study will help assisting civil and structural engineering practitioners in evaluating the decisions against the energy consumption and carbon impact as well as the cost of the structures during the initial design stages and tendering requirement.

8.3 Conclusions

This study leads to some significant conclusions which are summarised as follows:

- 1- The research concluded that the integration of the analysis, design and optimisation methods, employed in MATLAB, can be effective in obtaining prompt optimum results during the decision-making stage, where the solutions are obtained in less than 12 minutes for up to 9 storeys three-dimensional design problems.
- 2- General speaking, all three optimisation techniques adopted in this study are reliable and efficient structural optimisation tools. They are capable in finding optimum or near optimum I and H-section designations for practical steel frames under the real design code provisions. However, occasionally, the optimum results were trapped at local minima in some of the runs. Therefore, their performance varies according to the size of the problem and the type of objective function.
- 3- HSA is found to be the most efficient of the three methods studied in optimizing single and multi-objective problems of two and three-dimensional steel structures. The results show that optimum results obtained from HSA is 3% and 5% lesser than that of GA and PSO respectively for single-objective problems. For multi-objective problems, HSA produced optimum result that is 16% and 7% lesser than that of GA and PSO. The study also revealed that the algorithms perform better in single objective problems than in multi-objective problems.
- 4- The design search history graphs, generated in this study, show that the initial starting value of the algorithm has no influence on the resulting value. Although, GA and PSO started at a low value, they were unable to improvise the vectors effectively, compared to HAS, which started at a high value. However, they were able to improvise the vectors effectively, until the minimal weight and embodied energy were obtained.

- 5- After modifying all the used algorithms to help improve the computational time, it was found that PSO requires less time than others in single objective problems, whereas HAS provides a faster convergence in multi-objective problems, compared to GA and PSO.
- 6- The comparison of the results, obtained by each technique, showed a directly proportional relationship between weight and embodied energy.
- 7- GA requires an adequate generation and iteration size to ensure that the optimiser researches the space effectively. Whereas, PSO doesn't require high size of iteration, though, it requires an adequate swarm size. Particle Swarm method requires a higher computational time for multi-objective problems, compared to the others in all cases.
- 8- Iteration and population size have a significant effect on the values of the optimum results. An increase in one of these parameters can cause a high increase in the computational time and a very small decrease in the optimum solution.
- 9- A parametric study has been carried out to investigate the effect of imposed loading intensity, steel grade, and different bay spacing in both direction of the structure.
- 10- From the parametric results, it can be concluded that lower span lengths, in both directions, can lead to a 43% savings in embodied energy, compared to that when using larger spans. This might be due to the non-presence of the bracing system.
- 11- It has been revealed that using S355 steel grade can lead to more savings in the objective functions, compared to S275 steel grade. The highest percentage of saving observed was 10% for structures with a lower number of bays in X and Z directions (long spans), and the difference reduces with an increase in the number of bays (short spans) as shown in Figure 7-14.
- 12- Figures and values provided for the structures, as well as the parameters per square meter calculated, are expected to guide designers on which bay arrangement to adopt for similar structures and indicate how much embodied energy, cost, weight and embodied carbon they are going to use.
- 13- Increasing the variable actions value led to a minor increase in the weight and embodied energy of the structures. However, those with larger spans tend to show a significant increase in the objective functions.

- 14- The effect of grouping system for the members of the structure on the optimal results were investigated in the validation chapter and compared with the literature. The savings in the objective functions observed were up to 16%.
- 15- The developed optimisation models have proven their effectiveness in evaluating the embodied energy, embodied carbon, and cost for steel structures, and can be used to assist designers, during the initial stages, in evaluating the design decisions against their energy consumption and carbon impacts.

8.4 Recommendations for future work

The work undertaken in this research has indicated some intriguing areas for further structure optimisation research. Some recommendations for future research are as follows:

- 1- Future research can further investigate the efficiency of multi-objective functions using other multi-objective stochastic methods such as Simulated Annealing, Ant Colony Optimisation, Differential Evolution and Tabu Search.
- 2- Further develop the embodied energy, embodied carbon and cost models in such a way that they include the whole life cycle stages of the structure such as operation, maintenance, and demolition stages.
- 3- Further research can be carried out including dynamic loads such as earthquake load in addition to static permanent action (dead load) and variable action (live load) in order to investigate the optimum design of steel structures using these optimisation techniques.
- 4- Further research can be carried out by incorporating the design of connections and its effect on the optimum cost, embodied energy. Using semi-rigid connections and Simple construction method (simple joints) with bracing system types and arrangement
- 5- Further research might be implemented to investigate the difference in embodied energy and carbon savings between the use of Europe code 3 and the American code of practice known as AISC-LRFD or any other international code of practice.
- 6- Other section types such as IPE, HEA and HEB can be included to investigate their influence on the optimal solutions.

- 7- Since the parameters of the metaheuristic methods are sensitive, comprehensive research is worthwhile to find a formula that defines the relationship between the appropriate design variables, population size, iteration size, minimum and maximum mutation probabilities, elite count rate, crossover fraction, mutation rate, inflation rate, acceleration coefficients, and maximum number of generations so that the metaheuristic parameters can be directly specified for a particular problem.

References

- Adeli, H., Sarma, K.C., 2006. *Cost Optimization of Structures: Fuzzy Logic, Genetic Algorithms, and Parallel Computing*. John Wiley & Sons, Ltd, Chichester, England.
- Aghoury, M. el, Tawfic, M., Amoush, E., 2020. Compressive Strength of Axially Loaded Built-up Sigma Cold Formed Sections Columns. *Future Engineering Journal* 1, 2314–7237.
- Akbarnezhad, A., Xiao, J., 2017. Estimation and minimization of embodied carbon of buildings: A review. *Buildings* 7, 1–24. <https://doi.org/10.3390/buildings7010005>
- Akin, J.E., Arjona-baez, J., 2001. Enhancing structural topology optimization. *Engineering Computations (Swansea, Wales)* 18, 663–675. <https://doi.org/10.1108/02644400110387640>
- Ali, D., Seyan, O., 2014. *Optimum Design of Reinforced Concrete Columns According To Aci and Ec2 Codes*.
- Alkhadashi, A., Mohammad, F., Zubayr, R.O., Aoun Klalib, H., Balik, P., 2022. Multi-objective design optimisation of steel framed structures using three different methods. *International Journal of Structural Integrity* 13, 92–111. <https://doi.org/10.1108/IJSI-07-2021-0080>
- Alkhadashi, A.E.M., 2016. *OPTIMUM DESIGN OF CIRCULAR REINFORCED CONCRETE COLUMNS ACCORDING TO ACI 318-11 AND EC2 CODES*. Nottingham Trent University.
- Arora, J.S., 2016. *Introduction to optimum design*, 4th Edition. ed. Academic Press, Iowa City, IA, USA.
- Aydoğdu, İ., 2010. *OPTIMUM DESIGN OF 3-D IRREGULAR STEEL FRAMES USING ANT COLONY OPTIMIZATION AND HARMONY SEARCH ALGORITHMS*. MIDDLE EAST TECHNICAL UNIVERSITY.
- Barraza, M., Bojórquez, E., Fernández-González, E., Reyes-Salazar, A., 2017. Multi-objective optimization of structural steel buildings under earthquake loads using NSGA-II and PSO. *KSCE Journal of Civil Engineering* 21, 488–500. <https://doi.org/10.1007/s12205-017-1488-7>
- Belegundu, A.D., Chandrupatla, T.R., 2011. *Optimization Concepts and Applications in Engineering*, 2nd ed, Kybernetes. Cambridge University, Cambridge. <https://doi.org/10.1108/03684920510595436>

- Brütting, J., Senatore, G., Fivet, C., 2018. Optimization formulations for the design of low embodied energy structures made from reused elements. *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)* 10863 LNCS, 139–163. https://doi.org/10.1007/978-3-319-91635-4_8
- Brütting, J., Senatore, G., Schevenels, M., Fivet, C., 2020. Optimum Design of Frame Structures From a Stock of Reclaimed Elements. *Frontiers in Built Environment* 6. <https://doi.org/10.3389/fbuil.2020.00057>
- Burgan, B.A., Rackham, J.W., Labory, F.L., Klósak, M., Potjes, B., 2012. Economics of steel-framed buildings in Europe (ESE).
- Camp, C. v., Assadollahi, A., 2013. CO2 and cost optimization of reinforced concrete footings using a hybrid big bang-big crunch algorithm. *Structural and Multidisciplinary Optimization* 48, 411–426. <https://doi.org/10.1007/s00158-013-0897-6>
- Camp, C. v., Huq, F., 2013. CO2 and cost optimization of reinforced concrete frames using a big bang-big crunch algorithm. *Engineering Structures* 48, 363–372. <https://doi.org/10.1016/j.engstruct.2012.09.004>
- Cazacu, R., Grama, L., 2014. Steel Truss Optimization Using Genetic Algorithms and FEA. *Procedia Technology* 12, 339–346. <https://doi.org/10.1016/j.protcy.2013.12.496>
- Chae, C.-U., Kim, S., 2016. Evaluation of Embodied Energy and CO2 eq for Building Construction (Annex 57), Institute for Building Environment and Energy Conservation.
- Charalampakis, A.E., Tsiatas, G.C., 2019. Critical Evaluation of Metaheuristic Algorithms for Weight Minimization of Truss Structures. *Frontiers in Built Environment* 5. <https://doi.org/10.3389/fbuil.2019.00113>
- Chen, I.H., Chen, W.F., 1999. Practical advanced analysis for seismic frame design. *Advances in Structural Engineering* 2, 237–263. <https://doi.org/10.1177/136943329900200401>
- Cicconi, P., Germani, M., Bondi, S., Zuliani, A., Cagnacci, E., 2016. A Design Methodology to Support the Optimization of Steel Structures. *Procedia CIRP* 50, 58–64. <https://doi.org/10.1016/j.procir.2016.05.030>
- Coello, C.A.C., Lamont, G.B., Veldhuizen, D.A. Van, 2007. Evolutionary Algorithms for Solving Multi-Objective Problems, *Evolutionary Algorithms for Solving Multi-Objective Problems*. <https://doi.org/10.1007/978-0-387-36797-2>

- Cohn, M.Z., Dinovitzer, A.S., 1994. Application of Structural Optimization. *Journal of Structural Engineering* 120. [https://doi.org/10.1061/\(ASCE\)0733-9445\(1994\)120:2\(617\)](https://doi.org/10.1061/(ASCE)0733-9445(1994)120:2(617))
- Cullen, J.M., Allwood, J.M., Bambach, M.D., 2012. Mapping the global flow of steel: From steelmaking to end-use goods. *Environmental Science and Technology* 46, 13048–13055. <https://doi.org/10.1021/es302433p>
- Degertekin, S.O., 2007. A comparison of simulated annealing and genetic algorithm for optimum design of nonlinear steel space frames. *Structural and Multidisciplinary Optimization* 34, 347–359. <https://doi.org/10.1007/s00158-007-0096-4>
- Degertekin, S.O., Hayalioglu, M.S., 2010. Harmony search algorithm for minimum cost design of steel frames with semi-rigid connections and column bases. *Structural and Multidisciplinary Optimization* 42, 755–768. <https://doi.org/10.1007/s00158-010-0533-7>
- Dixit, M.K., 2017. Life cycle embodied energy analysis of residential buildings: A review of literature to investigate embodied energy parameters. *Renewable and Sustainable Energy Reviews* 79, 390–413. <https://doi.org/10.1016/j.rser.2017.05.051>
- Dogan, E., 2010. Optimim design of Rigid and Semi-rigid steel sway frames including soil-structure interacton. Middle east technical university.
- Dokeroglu, T., Sevinc, E., Kucukyilmaz, T., Cosar, A., 2019. A survey on new generation metaheuristic algorithms. *Computers and Industrial Engineering* 137. <https://doi.org/10.1016/j.cie.2019.106040>
- Domingos, A., Ferreira, D., Mainier, F.B., 2015. Application of Life Cycle Assessment (LCA) in Construction Industry 3–7.
- Eleftheriadis, S., Duffour, P., Greening, P., James, J., Stephenson, B., Mumovic, D., 2018. Investigating relationships between cost and CO2 emissions in reinforced concrete structures using a BIM-based design optimisation approach. *Energy and Buildings* 166, 330–346. <https://doi.org/10.1016/j.enbuild.2018.01.059>
- Eleftheriadis, Stathis, Dunant, C.F., Drewniok, M.P., Rogers-Tizard, W., Kyprianou, C., 2018. Comparative numerical analysis for cost and embodied carbon optimisation of steel building structures. *Advances in Computational Design* 3, 385–404. <https://doi.org/10.12989/acd.2018.3.4.385>

- Erdal, Ferhat, Saka, M.P., Erdal, F, 2008. Innovative design of green buildings for harsh environment View project Optimal design of civil engineering structures by aid of contemporary metaheuristic algorithms View project EFFECT OF BEAM SPACING IN THE HARMONY SEARCH BASED OPTIMUM DESIGN OF GRILLAGES, ASIAN JOURNAL OF CIVIL ENGINEERING (BUILDING AND HOUSING).
- Esfandiary, M.J., Sheikholarefin, S., Bondarabadi, H.A.R., 2016. a Combination of Particle Swarm Optimization and Multi - Criterion Decision - Making for Optimum Design of Reinforced Concrete Frames. INTERNATIONAL JOURNAL OF OPTIMIZATION IN CIVIL ENGINEERING Int . J . Optim . Civil Eng 6, 245–268.
- Fathizadeh, S.F., Vosoughi, A.R., Banan, M.R., 2021. Considering soil–structure interaction effects on performance-based design optimization of moment-resisting steel frames by an engineered cluster-based genetic algorithm. Engineering Optimization 53, 440–460. <https://doi.org/10.1080/0305215X.2020.1739278>
- Foraboschi, P., Mercanzin, M., Trabucco, D., 2014. Sustainable structural design of tall buildings based on embodied energy. Energy and Buildings 68, 254–269. <https://doi.org/10.1016/j.enbuild.2013.09.003>
- Galambos, T. v, 2006. STRUCTURAL DESIGN CODES: THE BRIDGE BETWEEN RESEARCH AND PRACTICE. Minneapolis, USA. <https://doi.org/10.2749/222137806796168868>
- Hammond, G., Jones, C., 2011. A BSRIA Guide. Embodied Carbon: The Inventory of Carbon and Energy. University of Bath, UK. Ice 136. <https://doi.org/10.1680/ener.2011.164.4.206>
- Hasançebi, Çarbaş, Doğan, Erdal, Saka, 2009. Performance evaluation of metaheuristic search techniques in the optimum design of real size pin jointed structures. Computers and Structures 87.
- Hasançebi, O., Ferhat Erdal, ;, Saka, M.P., 2010. Adaptive Harmony Search Method for Structural Optimization. <https://doi.org/10.1061/ASCEST.1943-541X.0000128>
- Haynes, R., 2010. Embodied Energy Calculations within Life Cycle Analysis of Residential Buildings. Etool.Net.Au 2010, 1–15.

- Heinisuo, M., Mela, K., Tiainen, T., Jokinen, T., 2016. Cost optimization of high strength steel structure. *Proceedings of the 16th International Conference on Computing in Civil and Building Engineering* 700, 32.
- Issa, H., Mohammad, F., 2008. Investigating the Optimum Design of Steel Portal Frame Using Genetic Algorithms Investigating the Optimum Design of Steel Portal Frame Using Genetic Algorithms. *BuHu 8th International Postgraduate Research Conference, Vol. 2 Investigating 2*, 35–47.
- Issa, H.K., 2010. Design Optimisation of Steel Portal Frames Using Modified Distributed Genetic Algorithms 272.
- IstructE, 2000. *Manual for the design of steelwork building structures*, Foundryman. SETO, London.
- Jin, S.-W., Ohmori, H., Lee, S.-J., 2017. Optimal design of steel structures considering welding cost and constructability of beam-column connections. *Journal of Constructional Steel Research* 135, 292–301. <https://doi.org/10.1016/j.jcsr.2017.03.020>
- Kameshki, E.S., 2003. Discrete Optimum Design of Steel Frames by Genetic Algorithm. *Journal of King Saud University - Engineering Sciences* 15, 217–233. [https://doi.org/10.1016/S1018-3639\(18\)30772-4](https://doi.org/10.1016/S1018-3639(18)30772-4)
- Karlaftis, M.G., 2015. *Engineering and Applied Sciences Optimization*, 1st ed. Springer International Publishing, Cham, Switzerland. <https://doi.org/https://doi.org/10.1007/978-3-319-18320-6>
- Kaveh, A., Ghazaan, M.I., 2018. *Meta-heuristic Algorithms for Optimal Design of Real-Size Structures*, 1st ed. Springer, Cham, Switzerland. <https://doi.org/https://doi.org/10.1007/978-3-319-78780-0>
- Kaveh, A., Talatahari, S., 2012. Charged system search for optimal design of frame structures. *Applied Soft Computing Journal* 12, 382–393. <https://doi.org/10.1016/j.asoc.2011.08.034>
- Khalifa, A., 2011. Design Optimization of semi rigid steel framed Structures to AISC- LRFD using Harmony search algorithm. *The Islamic Univeristy of Gaza*.
- Lee, K.S., Geem, Z.W., 2004. A new structural optimization method based on the harmony search algorithm. *Computers and Structures* 82, 781–798. <https://doi.org/10.1016/j.compstruc.2004.01.002>

- Lim, J.B.P., Nethercot, D.A., 2002. Design and development of a general cold-formed steel portal framing system. *Structural Engineer* 80, 31–40.
- Luthra, I., Krishna, S., Upahyay, D., Gupta, R., 2017. Proceedings of the International Conference on Electronics, Communication and Aerospace Technology (ICECA 2017) : date: 20,21,22, April 2017. International Conference on Electronics, Communication and Aerospace Technology (ICECA).
- Mao, C., Shen, Q., Shen, L., Tang, L., 2013. Comparative study of greenhouse gas emissions between off-site prefabrication and conventional construction methods: Two case studies of residential projects. *Energy and Buildings* 66, 165–176. <https://doi.org/10.1016/j.enbuild.2013.07.033>
- Martin, L., Purkiss, J., 2008. Structural design of steelwork to EN 1993 and EN 1994, 3rd ed. Butterworth-Heinemann, Oxford.
- Marwaha, A., 2017. DEPARTMENT OF CIVIL ENGINEERING H24A04 DESIGN OPTIMISATION OF THREE DIMENSIONAL STEEL FRAME STRUCTURES USING THE GENETIC ALGORITHM TECHNIQUE.
- McKenzie, W., 1998. Design of structural steelwork, 1st ed. Red Globe Press, England.
- Mckhenzie, W.M.C., 2004. Design of Structural Elements, Design of Structural Elements. Palgrave Macmillan, Hampshire. <https://doi.org/10.1201/b18121>
- Megson, T.H.G., 2019. Deflection of Beams, Structural and Stress Analysis. <https://doi.org/10.1016/b978-0-08-102586-4.00013-5>
- Mela, K., Heinisuo, M., 2014. Weight and cost optimization of welded high strength steel beams. *Engineering Structures* 79, 354–364. <https://doi.org/10.1016/j.engstruct.2014.08.028>
- Messac, A., 2015. Optimization in Practice with MATLAB®, Optimization in Practice with MATLAB®. Cambridge University Press. <https://doi.org/10.1017/cbo9781316271391>
- Milne, G., 2013. Materials | YourHome [WWW Document]. URL <https://www.yourhome.gov.au/materials> (accessed 7.27.20).
- Mohammad, F.A., Hemin, G.A., 2018. Optimum design of reinforced concrete cantilever retaining walls. *Pamukkale University Journal of Engineering Sciences* 24, 1043–1050. <https://doi.org/10.5505/pajes.2018.57873>

- Moynihan, M.C., Allwood, J.M., 2014. Utilization of structural steel in buildings. Proceedings of the Royal Society A: Mathematical, Physical and Engineering Sciences 470. <https://doi.org/10.1098/rspa.2014.0170>
- Msabawy, A.A., 2017. Minimum Weight of Cold-Formed Steel Portal Frames with Semi-Rigid Joints Using GRG Algorithm. Nottingham Trent University.
- Nagarajan, P., 2018. Matrix Methods of Structural Analysis, 1st ed. Boca Raton. <https://doi.org/https://doi.org/10.1201/9781351210324>
- Nässén, J., Holmberg, J., Wadeskog, A., Nyman, M., 2007. Direct and indirect energy use and carbon emissions in the production phase of buildings: An input-output analysis. Energy 32, 1593–1602. <https://doi.org/10.1016/j.energy.2007.01.002>
- Oh, B.K., Glisic, B., Lee, S.H., Cho, T., Park, H.S., 2019. Comprehensive investigation of embodied carbon emissions, costs, design parameters, and serviceability in optimum green construction of two-way slabs in buildings. Journal of Cleaner Production 222, 111–128. <https://doi.org/10.1016/j.jclepro.2019.03.003>
- Pavlovčič, L., Krajnc, A., Beg, D., 2004. Cost function analysis in the structural optimization of steel frames. Structural and Multidisciplinary Optimization 28, 286–295. <https://doi.org/10.1007/s00158-004-0430-z>
- Phan, D.T., Lim, J.B.P., Sha, W., Siew, C.Y.M., Tanyimboh, T.T., Issa, H.K., Mohammad, F.A., 2012. Design optimization of cold-formed steel portal frames taking 00, 1–19.
- Phan, T., Lim, J., Selowara Joo, M., Lau, H.-H., 2017. Design Optimization of Long-Span Cold-Formed Steel Portal Frames Accounting for Effect of Knee Brace Joint Configuration. Technologies (Basel) 5, 81. <https://doi.org/10.3390/technologies5040081>
- Plevris, V., Batavanis, A., Papadrakakis, M., 2011. Optimum Design of Steel Structures With the Particle 25–28.
- Prakash, A., Agarwala, S.K., Singh, K.K., 1988. Optimum design of reinforced concrete sections. Computers and Structures 30, 1009–1011. [https://doi.org/10.1016/0045-7949\(88\)90142-3](https://doi.org/10.1016/0045-7949(88)90142-3)
- Rao, S.S., 2020. Engineering optimization : theory and practice, 5th ed. John Wiley & Sons, Hoboken, NJ, USA.

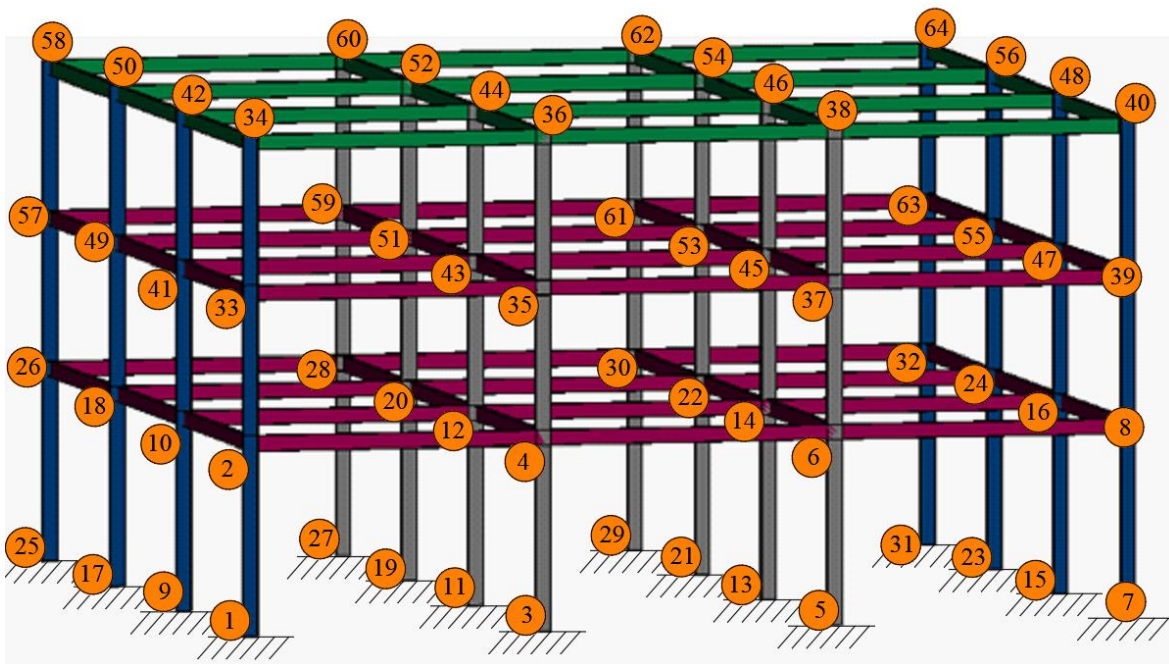
- Rao, S.S., 2009. Engineering Optimization, Fourth. ed, Operational Research Quarterly (1970-1977). John Wiley & Sons, Canada. <https://doi.org/10.2307/3009231>
- Ravat, A., Dhawan, A., Tiwari, M., 2021. Advances in VLSI, Communication, and Signal Processing, in: Harvey, D., Kar, H., Verma, S., Bhadauria, V. (Eds.), LMI and YALMIP: Modeling and Optimization Toolbox in MATLAB. Springer, Singapore, pp. 507–515.
- Ravindran, A., Reklaitis, G. v., Ragsdell, K.M., 2006. Engineering optimization : methods and applications, 2nd ed. John Wiley & Sons, Hoboken,.
- Robb, I., 1972. Steel Frame Design Examples, Third. ed. Palgrave Macmillan, London. <https://doi.org/10.1007/978-1-349-81762-7>
- Saka, M.P., 2009. Optimum design of steel sway frames to BS5950 using harmony search algorithm. Journal of Constructional Steel Research 65, 36–43. <https://doi.org/10.1016/j.jcsr.2008.02.005>
- Saka, M.P., Erdal, F., 2009. Harmony search based algorithm for the optimum design of grillage systems to LRFD-AISC. Structural and Multidisciplinary Optimization 38, 25–41. <https://doi.org/10.1007/s00158-008-0263-2>
- Sarma, K.C., Adeli, H., 2002. Life-cycle cost optimization of steel structures. International Journal for Numerical Methods in Engineering 55, 1451–1462. <https://doi.org/10.1002/nme.549>
- Scheuer, C., Keoleian, G.A., Reppe, P., 2003. Life cycle energy and environmental performance of a new university building: modeling challenges and design implications. Energy Build 35, 1049–1064.
- SCI, 2008. Steel buildings in Europe: Multi-storey steel buildings Part 2: Concept Design. Steel Construction Institute.
- Shadram, F., Johansson, T.D., Lu, W., Schade, J., Olofsson, T., 2016. An integrated BIM-based framework for minimizing embodied energy during building design. Energy and Buildings 128, 592–604. <https://doi.org/10.1016/j.enbuild.2016.07.007>
- Soori, S., Salajegheh, E., 2019. Optimum Design of Steel Moment Frame by using Particle Swarm Optimization. Optimum Design of Steel Moment Frame by using Particle Swarm Optimization Algorithm.

- Talatahari, S., Gandomi, A.H., Yang, X.S., Deb, S., 2015. Optimum design of frame structures using the Eagle Strategy with Differential Evolution. *Engineering Structures* 91, 16–25. <https://doi.org/10.1016/j.engstruct.2015.02.026>
- Torregosa, R.F., Kanok-Nukulchai, W., 2002. Weight optimization of steel trusses using genetic algorithm. *Advances in Structural Engineering* 5, 99–110.
- Trahair, N.S., Bradford, M.A., Nethercot, D.A., Gardner, L., 2008. *The Behaviour and Design of Steel Structures to EC3*, 4th ed. Taylor and Francis Group, London.
- UKGBC, 2018. *Embodied Carbon: Developing a Client Brief*. UK Green Building Council 1–59.
- UKGBC, 2014. *Tackling embodied carbon in buildings - INCLUDES CASE STUDIES*. UK Green Building Council 1–16.
- Victoria, 2016. *C K 2016 B.C. BEST PRACTICES METHODOLOGY FOR QUANTIFYING GREENHOUSE GAS EMISSIONS INCLUDING GUIDANCE FOR PUBLIC SECTOR ORGANIZATIONS, LOCAL GOVERNMENTS AND COMMUNITY EMISSIONS* Ministry of Environment.
- Vukotic, L., Fenner, R.A., Symons, K., 2010. Assessing embodied energy of building structural elements. *Proceedings of the Institution of Civil Engineers: Engineering Sustainability* 163, 147–158. <https://doi.org/10.1680/ensu.2010.163.3.147>
- Williams, A., 2009. *Structural Analysis: In Theory and Practice*.
- Yassami, M., Ashtari, P., 2015. Using fuzzy genetic algorithm for the weight optimization of steel frames with semi-rigid connections. *International Journal of Steel Structures* 15, 63–73. <https://doi.org/10.1007/s13296-014-1105-2>
- Yenjay, O., 2005. A comparative study on optimisation methods for the constrained nonlinear programming problems'. *Mathematical Problems in Engineering* 165–173. <https://doi.org/doi:10.1155/mpe.2005.165>
- Zhou, H., Azar, E.R., 2018. Embodied energy assessment of building structural systems using building information modeling. *ISARC 2018 - 35th International Symposium on Automation and Robotics in Construction and International AEC/FM Hackathon: The Future of Building Things*. <https://doi.org/10.22260/isarc2018/0100>

APPENDIX-A

Validation of programmed structural analysis with PROKON version (W3.1.08)

A-1. Results of the 3-dimensional structure with 3x3



A-1.1 Below are the structural analysis results of the self-coded program.

	1	2	3	4	5	6	7	8
	Members	Joint	X-Axial_kN	Y-Shear_kN	X-Shear_kN	Tors_kNm	Myy_kNm	Mxx_kNm
1	'M-1'	1	429.8500	6.5000	2.7200	0	-3.6400	11.7700
2	"	2	-429.8500	11.5000	-2.7200	0	-7.2500	-21.7700
3	'M-2'	3	690.0300	9.9800	2.7200	0	-3.6400	22.5800
4	"	4	-690.0300	-9.9800	-2.7200	0	-7.2500	17.3400
5	'M-3'	5	688.8200	9.3200	2.7200	0	-3.6400	21.9300
6	"	6	-688.8200	-9.3200	-2.7200	0	-7.2500	15.3600
7	'M-4'	7	445.0700	19.2000	2.7200	0	-3.6400	35.4000
8	"	8	-445.0700	-19.2000	-2.7200	0	-7.2500	41.4000
9	'M-5'	9	742.9600	6.5000	-0.6600	0	0.8800	11.7700
10	"	10	-742.9600	11.5000	0.6600	0	1.7700	-21.7700
11	'M-6'	11	1.0032e+03	9.9800	-0.6600	0	0.8800	22.5800
12	"	12	-1.0032e+03	-9.9800	0.6600	0	1.7700	17.3400
13	'M-7'	13	1.0019e+03	9.3200	-0.6600	0	0.8800	21.9300
14	"	14	-1.0019e+03	-9.3200	0.6600	0	1.7700	15.3600
15	'M-8'	15	758.1800	19.2000	-0.6600	0	0.8800	35.4000
16	"	16	-758.1800	-19.2000	0.6600	0	1.7700	41.4000
17	'M-9'	17	742.9600	6.5000	0.6600	0	-0.8800	11.7700
18	"	18	-742.9600	11.5000	-0.6600	0	-1.7700	-21.7700
19	'M-10'	19	1.0032e+03	9.9800	0.6600	0	-0.8800	22.5800
20	"	20	-1.0032e+03	-9.9800	-0.6600	0	-1.7700	17.3400
21	'M-11'	21	1.0019e+03	9.3200	0.6600	0	-0.8800	21.9300
22	"	22	-1.0019e+03	-9.3200	-0.6600	0	-1.7700	15.3600
23	'M-12'	23	758.1800	19.2000	0.6600	0	-0.8800	35.4000
24	"	24	-758.1800	-19.2000	-0.6600	0	-1.7700	41.4000
25	'M-13'	25	429.8500	6.5000	-2.7200	0	3.6400	11.7700
26	"	26	-429.8500	11.5000	2.7200	0	7.2500	-21.7700
27	'M-14'	27	690.0300	9.9800	-2.7200	0	3.6400	22.5800
28	"	28	-690.0300	-9.9800	2.7200	0	7.2500	17.3400
29	'M-15'	29	688.8200	9.3200	-2.7200	0	3.6400	21.9300
30	"	30	-688.8200	-9.3200	2.7200	0	7.2500	15.3600
31	'M-16'	31	445.0700	19.2000	-2.7200	0	3.6400	35.4000
32	"	32	-445.0700	-19.2000	2.7200	0	7.2500	41.4000
33	'M-17'	2	271.1500	-19.5200	9.3400	0	-14.1300	-41.0100
34	"	33	-271.1500	33.0200	-9.3400	0	-13.8800	-37.7900
35	'M-18'	4	430.7500	7.5300	9.3400	0	-14.1300	10.4000

	1	2	3	4	5	6	7	8
	Members	Joint	X-Axial_kN	Y-Shear_kN	X-Shear_kN	Tors_kNm	Myy_kNm	Mxx_kNm
36	"	35	-430.7500	-7.5300	-9.3400	0	-13.8800	12.2000
37	'M-19'	6	430.6600	5.9100	9.3400	0	-14.1300	6.3400
38	"	37	-430.6600	-5.9100	-9.3400	0	-13.8800	11.3800
39	'M-20'	8	277.3600	33.0800	9.3400	0	-14.1300	49.6500
40	"	39	-277.3600	-33.0800	-9.3400	0	-13.8800	49.5800
41	'M-21'	10	466.1900	-19.5200	-2.1700	0	3.3600	-41.0100
42	"	41	-466.1900	33.0200	2.1700	0	3.1400	-37.7900
43	'M-22'	12	625.7900	7.5300	-2.1700	0	3.3600	10.4000
44	"	43	-625.7900	-7.5300	2.1700	0	3.1400	12.2000
45	'M-23'	14	625.7000	5.9100	-2.1700	0	3.3600	6.3400
46	"	45	-625.7000	-5.9100	2.1700	0	3.1400	11.3800
47	'M-24'	16	472.4000	33.0800	-2.1700	0	3.3600	49.6500
48	"	47	-472.4000	-33.0800	2.1700	0	3.1400	49.5800
49	'M-25'	18	466.1900	-19.5200	2.1700	0	-3.3600	-41.0100
50	"	49	-466.1900	33.0200	-2.1700	0	-3.1400	-37.7900
51	'M-26'	20	625.7900	7.5300	2.1700	0	-3.3600	10.4000
52	"	51	-625.7900	-7.5300	-2.1700	0	-3.1400	12.2000
53	'M-27'	22	625.7000	5.9100	2.1700	0	-3.3600	6.3400
54	"	53	-625.7000	-5.9100	-2.1700	0	-3.1400	11.3800
55	'M-28'	24	472.4000	33.0800	2.1700	0	-3.3600	49.6500
56	"	55	-472.4000	-33.0800	-2.1700	0	-3.1400	49.5800
57	'M-29'	26	271.1500	-19.5200	-9.3400	0	14.1300	-41.0100
58	"	57	-271.1500	33.0200	9.3400	0	13.8800	-37.7900
59	'M-30'	28	430.7500	7.5300	-9.3400	0	14.1300	10.4000
60	"	59	-430.7500	-7.5300	9.3400	0	13.8800	12.2000
61	'M-31'	30	430.6600	5.9100	-9.3400	0	14.1300	6.3400
62	"	61	-430.6600	-5.9100	9.3400	0	13.8800	11.3800
63	'M-32'	32	277.3600	33.0800	-9.3400	0	14.1300	49.6500
64	"	63	-277.3600	-33.0800	9.3400	0	13.8800	49.5800
65	'M-33'	33	107.6700	-21.5900	8.1900	0	-12.7400	-38.1700
66	"	34	-107.6700	35.0900	-8.1900	0	-11.8300	-46.8400
67	'M-34'	35	172.4000	2.6700	8.1900	0	-12.7400	2.1200
68	"	36	-172.4000	-2.6700	-8.1900	0	-11.8300	5.8900
69	'M-35'	37	172.6400	1.6700	8.1900	0	-12.7400	1.5800
70	"	38	-172.6400	-1.6700	-8.1900	0	-11.8300	3.4200

	1	2	3	4	5	6	7	8
	Members	Joint	X-Axial_kN	Y-Shear_kN	X-Shear_kN	Tors_kNm	Myy_kNm	Mxx_kNm
71	'M-36'	39	109.0900	30.7500	8.1900	0	-12.7400	41.2200
72	"	40	-109.0900	-30.7500	-8.1900	0	-11.8300	51.0100
73	'M-37'	41	186.7700	-21.5900	-1.7300	0	2.7100	-38.1700
74	"	42	-186.7700	35.0900	1.7300	0	2.4900	-46.8400
75	'M-38'	43	251.5000	2.6700	-1.7300	0	2.7100	2.1200
76	"	44	-251.5000	-2.6700	1.7300	0	2.4900	5.8900
77	'M-39'	45	251.7400	1.6700	-1.7300	0	2.7100	1.5800
78	"	46	-251.7400	-1.6700	1.7300	0	2.4900	3.4200
79	'M-40'	47	188.1900	30.7500	-1.7300	0	2.7100	41.2200
80	"	48	-188.1900	-30.7500	1.7300	0	2.4900	51.0100
81	'M-41'	49	186.7700	-21.5900	1.7300	0	-2.7100	-38.1700
82	"	50	-186.7700	35.0900	-1.7300	0	-2.4900	-46.8400
83	'M-42'	51	251.5000	2.6700	1.7300	0	-2.7100	2.1200
84	"	52	-251.5000	-2.6700	-1.7300	0	-2.4900	5.8900
85	'M-43'	53	251.7400	1.6700	1.7300	0	-2.7100	1.5800
86	"	54	-251.7400	-1.6700	-1.7300	0	-2.4900	3.4200
87	'M-44'	55	188.1900	30.7500	1.7300	0	-2.7100	41.2200
88	"	56	-188.1900	-30.7500	-1.7300	0	-2.4900	51.0100
89	'M-45'	57	107.6700	-21.5900	-8.1900	0	12.7400	-38.1700
90	"	58	-107.6700	35.0900	8.1900	0	11.8300	-46.8400
91	'M-46'	59	172.4000	2.6700	-8.1900	0	12.7400	2.1200
92	"	60	-172.4000	-2.6700	8.1900	0	11.8300	5.8900
93	'M-47'	61	172.6400	1.6700	-8.1900	0	12.7400	1.5800
94	"	62	-172.6400	-1.6700	8.1900	0	11.8300	3.4200
95	'M-48'	63	109.0900	30.7500	-8.1900	0	12.7400	41.2200
96	"	64	-109.0900	-30.7500	8.1900	0	11.8300	51.0100
97	'M-49'	2	-8.0200	82.7400	0	0	0	62.7800
98	"	4	8.0200	97.2600	0	0	0	-106.3500
99	'M-50'	4	-10.4700	86.0600	0	0	0	78.6100
100	"	6	10.4700	93.9400	0	0	0	-102.2700
101	'M-51'	6	-13.8800	88.2500	0	0	0	80.5700
102	"	8	13.8800	91.7500	0	0	0	-91.0500
103	'M-52'	10	-8.0200	82.7400	0	0	0	62.7800
104	"	12	8.0200	97.2600	0	0	0	-106.3500
105	'M-53'	12	-10.4700	86.0600	0	0	0	78.6100

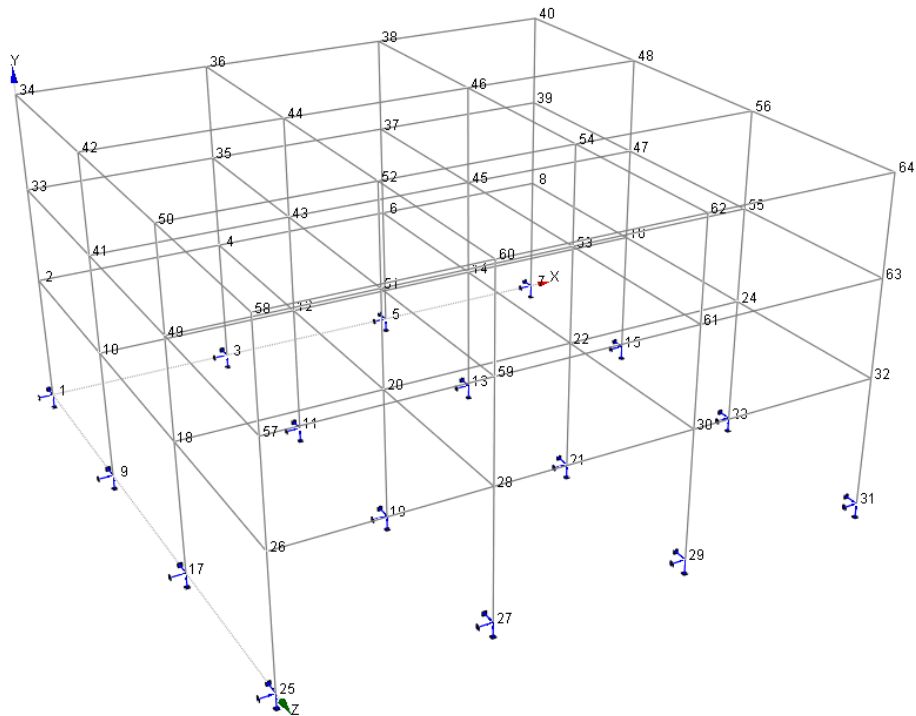
	1	2	3	4	5	6	7	8
	Members	Joint	X-Axial_kN	Y-Shear_kN	X-Shear_kN	Tors_kNm	Myy_kNm	Mxx_kNm
106	"	14	10.4700	93.9400	0	0	0	-102.2700
107	'M-54'	14	-13.8800	88.2500	0	0	0	80.5700
108	"	16	13.8800	91.7500	0	0	0	-91.0500
109	'M-55'	18	-8.0200	82.7400	0	0	0	62.7800
110	"	20	8.0200	97.2600	0	0	0	-106.3500
111	'M-56'	20	-10.4700	86.0600	0	0	0	78.6100
112	"	22	10.4700	93.9400	0	0	0	-102.2700
113	'M-57'	22	-13.8800	88.2500	0	0	0	80.5700
114	"	24	13.8800	91.7500	0	0	0	-91.0500
115	'M-58'	26	-8.0200	82.7400	0	0	0	62.7800
116	"	28	8.0200	97.2600	0	0	0	-106.3500
117	'M-59'	28	-10.4700	86.0600	0	0	0	78.6100
118	"	30	10.4700	93.9400	0	0	0	-102.2700
119	'M-60'	30	-13.8800	88.2500	0	0	0	80.5700
120	"	32	13.8800	91.7500	0	0	0	-91.0500
121	'M-61'	2	-6.6100	75.9600	0	0	0	21.3800
122	"	10	6.6100	104.0400	0	0	0	-105.6000
123	'M-62'	10	-5.1100	90	0	0	0	100.4700
124	"	18	5.1100	90	0	0	0	-100.4700
125	'M-63'	18	-6.6100	104.0400	0	0	0	105.6000
126	"	26	6.6100	75.9600	0	0	0	-21.3800
127	'M-64'	4	-6.6100	75.9600	0	0	0	21.3800
128	"	12	6.6100	104.0400	0	0	0	-105.6000
129	'M-65'	12	-5.1100	90	0	0	0	100.4700
130	"	20	5.1100	90	0	0	0	-100.4700
131	'M-66'	20	-6.6100	104.0400	0	0	0	105.6000
132	"	28	6.6100	75.9600	0	0	0	-21.3800
133	'M-67'	6	-6.6100	75.9600	0	0	0	21.3800
134	"	14	6.6100	104.0400	0	0	0	-105.6000
135	'M-68'	14	-5.1100	90	0	0	0	100.4700
136	"	22	5.1100	90	0	0	0	-100.4700
137	'M-69'	22	-6.6100	104.0400	0	0	0	105.6000
138	"	30	6.6100	75.9600	0	0	0	-21.3800
139	'M-70'	8	-6.6100	75.9600	0	0	0	21.3800
140	"	16	6.6100	104.0400	0	0	0	-105.6000

	1	2	3	4	5	6	7	8
	Members	Joint	X-Axial_kN	Y-Shear_kN	X-Shear_kN	Tors_kNm	Myy_kNm	Mxx_kNm
141	'M-71'	16	-5.1100	90	0	0	0	100.4700
142	"	24	5.1100	90	0	0	0	-100.4700
143	'M-72'	24	-6.6100	104.0400	0	0	0	105.6000
144	"	32	6.6100	75.9600	0	0	0	-21.3800
145	'M-73'	33	11.4300	86.4500	0	0	0	75.9500
146	"	35	-11.4300	93.5500	0	0	0	-97.2600
147	'M-74'	35	6.5700	87.7700	0	0	0	82.9400
148	"	37	-6.5700	92.2300	0	0	0	-96.3300
149	'M-75'	37	2.3300	88.7600	0	0	0	83.3700
150	"	39	-2.3300	91.2400	0	0	0	-90.8000
151	'M-76'	41	11.4300	86.4500	0	0	0	75.9500
152	"	43	-11.4300	93.5500	0	0	0	-97.2600
153	'M-77'	43	6.5700	87.7700	0	0	0	82.9400
154	"	45	-6.5700	92.2300	0	0	0	-96.3300
155	'M-78'	45	2.3300	88.7600	0	0	0	83.3700
156	"	47	-2.3300	91.2400	0	0	0	-90.8000
157	'M-79'	49	11.4300	86.4500	0	0	0	75.9500
158	"	51	-11.4300	93.5500	0	0	0	-97.2600
159	'M-80'	51	6.5700	87.7700	0	0	0	82.9400
160	"	53	-6.5700	92.2300	0	0	0	-96.3300
161	'M-81'	53	2.3300	88.7600	0	0	0	83.3700
162	"	55	-2.3300	91.2400	0	0	0	-90.8000
163	'M-82'	57	11.4300	86.4500	0	0	0	75.9500
164	"	59	-11.4300	93.5500	0	0	0	-97.2600
165	'M-83'	59	6.5700	87.7700	0	0	0	82.9400
166	"	61	-6.5700	92.2300	0	0	0	-96.3300
167	'M-84'	61	2.3300	88.7600	0	0	0	83.3700
168	"	63	-2.3300	91.2400	0	0	0	-90.8000
169	'M-85'	33	1.1500	77.0300	0	0	0	26.6200
170	"	41	-1.1500	102.9700	0	0	0	-104.4300
171	'M-86'	41	0.7100	90	0	0	0	98.5700
172	"	49	-0.7100	90	0	0	0	-98.5700
173	'M-87'	49	1.1500	102.9700	0	0	0	104.4300
174	"	57	-1.1500	77.0300	0	0	0	-26.6200
175	'M-88'	35	1.1500	77.0300	0	0	0	26.6200

	1	2	3	4	5	6	7	8
	Members	Joint	X-Axial_kN	Y-Shear_kN	X-Shear_kN	Tors_kNm	Myy_kNm	Mxx_kNm
176	"	43	-1.1500	102.9700	0	0	0	-104.4300
177	'M-89'	43	0.7100	90	0	0	0	98.5700
178	"	51	-0.7100	90	0	0	0	-98.5700
179	'M-90'	51	1.1500	102.9700	0	0	0	104.4300
180	"	59	-1.1500	77.0300	0	0	0	-26.6200
181	'M-91'	37	1.1500	77.0300	0	0	0	26.6200
182	"	45	-1.1500	102.9700	0	0	0	-104.4300
183	'M-92'	45	0.7100	90	0	0	0	98.5700
184	"	53	-0.7100	90	0	0	0	-98.5700
185	'M-93'	53	1.1500	102.9700	0	0	0	104.4300
186	"	61	-1.1500	77.0300	0	0	0	-26.6200
187	'M-94'	39	1.1500	77.0300	0	0	0	26.6200
188	"	47	-1.1500	102.9700	0	0	0	-104.4300
189	'M-95'	47	0.7100	90	0	0	0	98.5700
190	"	55	-0.7100	90	0	0	0	-98.5700
191	'M-96'	55	1.1500	102.9700	0	0	0	104.4300
192	"	63	-1.1500	77.0300	0	0	0	-26.6200
193	'M-97'	34	35.0900	57.2200	0	0	0	46.8400
194	"	36	-35.0900	62.7800	0	0	0	-63.5300
195	'M-98'	36	32.4100	59.1700	0	0	0	57.6300
196	"	38	-32.4100	60.8300	0	0	0	-62.6100
197	'M-99'	38	30.7500	61.3600	0	0	0	59.1900
198	"	40	-30.7500	58.6400	0	0	0	-51.0100
199	'M-100'	42	35.0900	57.2200	0	0	0	46.8400
200	"	44	-35.0900	62.7800	0	0	0	-63.5300
201	'M-101'	44	32.4100	59.1700	0	0	0	57.6300
202	"	46	-32.4100	60.8300	0	0	0	-62.6100
203	'M-102'	46	30.7500	61.3600	0	0	0	59.1900
204	"	48	-30.7500	58.6400	0	0	0	-51.0100
205	'M-103'	50	35.0900	57.2200	0	0	0	46.8400
206	"	52	-35.0900	62.7800	0	0	0	-63.5300
207	'M-104'	52	32.4100	59.1700	0	0	0	57.6300
208	"	54	-32.4100	60.8300	0	0	0	-62.6100
209	'M-105'	54	30.7500	61.3600	0	0	0	59.1900
210	"	56	-30.7500	58.6400	0	0	0	-51.0100

	1	2	3	4	5	6	7	8
	Members	Joint	X-Axial_kN	Y-Shear_kN	X-Shear_kN	Tors_kNm	Myy_kNm	Mxx_kNm
211	'M-106'	58	35.0900	57.2200	0	0	0	46.8400
212	"	60	-35.0900	62.7800	0	0	0	-63.5300
213	'M-107'	60	32.4100	59.1700	0	0	0	57.6300
214	"	62	-32.4100	60.8300	0	0	0	-62.6100
215	'M-108'	62	30.7500	61.3600	0	0	0	59.1900
216	"	64	-30.7500	58.6400	0	0	0	-51.0100
217	'M-109'	34	8.1900	50.4500	0	0	0	11.8300
218	"	42	-8.1900	69.5500	0	0	0	-69.1400
219	'M-110'	42	6.4500	60	0	0	0	66.6500
220	"	50	-6.4500	60	0	0	0	-66.6500
221	'M-111'	50	8.1900	69.5500	0	0	0	69.1400
222	"	58	-8.1900	50.4500	0	0	0	-11.8300
223	'M-112'	36	8.1900	50.4500	0	0	0	11.8300
224	"	44	-8.1900	69.5500	0	0	0	-69.1400
225	'M-113'	44	6.4500	60	0	0	0	66.6500
226	"	52	-6.4500	60	0	0	0	-66.6500
227	'M-114'	52	8.1900	69.5500	0	0	0	69.1400
228	"	60	-8.1900	50.4500	0	0	0	-11.8300
229	'M-115'	38	8.1900	50.4500	0	0	0	11.8300
230	"	46	-8.1900	69.5500	0	0	0	-69.1400
231	'M-116'	46	6.4500	60	0	0	0	66.6500
232	"	54	-6.4500	60	0	0	0	-66.6500
233	'M-117'	54	8.1900	69.5500	0	0	0	69.1400
234	"	62	-8.1900	50.4500	0	0	0	-11.8300
235	'M-118'	40	8.1900	50.4500	0	0	0	11.8300
236	"	48	-8.1900	69.5500	0	0	0	-69.1400
237	'M-119'	48	6.4500	60	0	0	0	66.6500
238	"	56	-6.4500	60	0	0	0	-66.6500
239	'M-120'	56	8.1900	69.5500	0	0	0	69.1400
240	"	64	-8.1900	50.4500	0	0	0	-11.8300

A-1.2 Below are the structural analysis results using PROKON software.



===== OUTPUT: LINEAR ANALYSIS =====

===== NODAL POINT DISPLACEMENTS at SLS =====

Node	Lcase	X-disp. mm	Y-disp. mm	Z-disp. mm	X-rot. rad.	Y-rot. rad.	Z-rot. rad.
1	1	0.00	0.00	0.00	0.0000	0.0000	0.0000
2	1	1.33	-0.90	-0.03	0.0023	0.0000	-0.0008
3	1	0.00	0.00	0.00	0.0000	0.0000	0.0000
4	1	1.35	-1.45	-0.03	0.0023	0.0000	-0.0002
5	1	0.00	0.00	0.00	0.0000	0.0000	0.0000
6	1	1.39	-1.45	-0.03	0.0023	0.0000	-0.0002
7	1	0.00	0.00	0.00	0.0000	0.0000	0.0000
8	1	1.43	-0.93	-0.03	0.0023	0.0000	0.0002
9	1	0.00	0.00	0.00	0.0000	0.0000	0.0000
10	1	1.33	-1.56	-0.01	-0.0006	0.0000	-0.0008
11	1	0.00	0.00	0.00	0.0000	0.0000	0.0000
12	1	1.35	-2.11	-0.01	-0.0006	0.0000	-0.0002
13	1	0.00	0.00	0.00	0.0000	0.0000	0.0000
14	1	1.39	-2.10	-0.01	-0.0006	0.0000	-0.0002
15	1	0.00	0.00	0.00	0.0000	0.0000	0.0000
16	1	1.43	-1.59	-0.01	-0.0006	0.0000	0.0002
17	1	0.00	0.00	0.00	0.0000	0.0000	0.0000
18	1	1.33	-1.56	0.01	0.0006	0.0000	-0.0008
19	1	0.00	0.00	0.00	0.0000	0.0000	0.0000
20	1	1.35	-2.11	0.01	0.0006	0.0000	-0.0002
21	1	0.00	0.00	0.00	0.0000	0.0000	0.0000
22	1	1.39	-2.10	0.01	0.0006	0.0000	-0.0002
23	1	0.00	0.00	0.00	0.0000	0.0000	0.0000
24	1	1.43	-1.59	0.01	0.0006	0.0000	0.0002
25	1	0.00	0.00	0.00	0.0000	0.0000	0.0000
26	1	1.33	-0.90	0.03	-0.0023	0.0000	-0.0008
27	1	0.00	0.00	0.00	0.0000	0.0000	0.0000
28	1	1.35	-1.45	0.03	-0.0023	0.0000	-0.0002
29	1	0.00	0.00	0.00	0.0000	0.0000	0.0000
30	1	1.39	-1.45	0.03	-0.0023	0.0000	-0.0002
31	1	0.00	0.00	0.00	0.0000	0.0000	0.0000
32	1	1.43	-0.93	0.03	-0.0023	0.0000	0.0002
33	1	2.20	-1.33	0.00	0.0022	0.0000	-0.0005
34	1	2.65	-1.50	0.04	0.0018	0.0000	-0.0006
35	1	2.16	-2.13	0.00	0.0022	0.0000	-0.0001
36	1	2.54	-2.40	0.04	0.0018	0.0000	0.0000
37	1	2.14	-2.12	0.00	0.0022	0.0000	-0.0001
38	1	2.44	-2.40	0.04	0.0018	0.0000	-0.0001
39	1	2.13	-1.37	0.00	0.0022	0.0000	0.0002
40	1	2.34	-1.54	0.04	0.0018	0.0000	0.0005
41	1	2.20	-2.29	0.00	-0.0005	0.0000	-0.0005
42	1	2.65	-2.59	0.01	-0.0004	0.0000	-0.0006
43	1	2.16	-3.09	0.00	-0.0005	0.0000	-0.0001
44	1	2.54	-3.49	0.01	-0.0004	0.0000	0.0000
45	1	2.14	-3.09	0.00	-0.0005	0.0000	-0.0001
46	1	2.44	-3.48	0.01	-0.0004	0.0000	-0.0001
47	1	2.13	-2.33	0.00	-0.0005	0.0000	0.0002
48	1	2.34	-2.63	0.01	-0.0004	0.0000	0.0005
49	1	2.20	-2.29	0.00	0.0005	0.0000	-0.0005
50	1	2.65	-2.59	-0.01	0.0004	0.0000	-0.0006
51	1	2.16	-3.09	0.00	0.0005	0.0000	-0.0001
52	1	2.54	-3.49	-0.01	0.0004	0.0000	0.0000
53	1	2.14	-3.09	0.00	0.0005	0.0000	-0.0001
54	1	2.44	-3.48	-0.01	0.0004	0.0000	-0.0001
55	1	2.13	-2.33	0.00	0.0005	0.0000	0.0002
56	1	2.34	-2.63	-0.01	0.0004	0.0000	0.0005
57	1	2.20	-1.33	0.00	-0.0022	0.0000	-0.0005
58	1	2.65	-1.50	-0.04	-0.0018	0.0000	-0.0006
59	1	2.16	-2.13	0.00	-0.0022	0.0000	-0.0001
60	1	2.54	-2.40	-0.04	-0.0018	0.0000	0.0000
61	1	2.14	-2.12	0.00	-0.0022	0.0000	-0.0001
62	1	2.44	-2.40	-0.04	-0.0018	0.0000	-0.0001
63	1	2.13	-1.37	0.00	-0.0022	0.0000	0.0002
64	1	2.34	-1.54	-0.04	-0.0018	0.0000	0.0005

===== BEAM ELEMENT END FORCES IN LOCAL ELEMENT AXES at ULS ==

Elem	Lcase	Axial kN	Y-Shear kN	X-Shear kN	Torsion kNm	M-yy kNm	M-xx kNm
1-	1	429.85	6.50	2.72	0.00	-3.64	11.77
2		-429.85	11.50	-2.72	0.00	-7.25	-21.77
3-	1	690.03	9.98	2.72	0.00	-3.64	22.58
4		-690.03	-9.98	-2.72	0.00	-7.25	17.34
5-	1	688.82	9.32	2.72	0.00	-3.64	21.93
6		-688.82	-9.32	-2.72	0.00	-7.25	15.36
7-	1	445.07	19.20	2.72	0.00	-3.64	35.40
8		-445.07	-19.20	-2.72	0.00	-7.25	41.40
9-	1	742.96	6.50	-0.66	0.00	0.88	11.77
10		-742.96	11.50	0.66	0.00	1.77	-21.77
11-	1	1003.15	9.98	-0.66	0.00	0.88	22.58
12		-1003.15	-9.98	0.66	0.00	1.77	17.34
13-	1	1001.94	9.32	-0.66	0.00	0.88	21.93
14		-1001.94	-9.32	0.66	0.00	1.77	15.36
15-	1	758.18	19.20	-0.66	0.00	0.88	35.40
16		-758.18	-19.20	0.66	0.00	1.77	41.40
17-	1	742.96	6.50	0.66	0.00	-0.88	11.77
18		-742.96	11.50	-0.66	0.00	-1.77	-21.77
19-	1	1003.15	9.98	0.66	0.00	-0.88	22.58
20		-1003.15	-9.98	-0.66	0.00	-1.77	17.34
21-	1	1001.94	9.32	0.66	0.00	-0.88	21.93
22		-1001.94	-9.32	-0.66	0.00	-1.77	15.36
23-	1	758.18	19.20	0.66	0.00	-0.88	35.40
24		-758.18	-19.20	-0.66	0.00	-1.77	41.40
25-	1	429.85	6.50	-2.72	0.00	3.64	11.77
26		-429.85	11.50	2.72	0.00	7.25	-21.77
27-	1	690.03	9.98	-2.72	0.00	3.64	22.58
28		-690.03	-9.98	2.72	0.00	7.25	17.34
29-	1	688.82	9.32	-2.72	0.00	3.64	21.93
30		-688.82	-9.32	2.72	0.00	7.25	15.36
31-	1	445.07	19.20	-2.72	0.00	3.64	35.40
32		-445.07	-19.20	2.72	0.00	7.25	41.40
2-	1	271.15	-19.52	9.34	0.00	-14.13	-41.01
33		-271.15	33.02	-9.34	0.00	-13.88	-37.79
4-	1	430.75	7.53	9.34	0.00	-14.13	10.40
35		-430.75	-7.53	-9.34	0.00	-13.88	12.20
6-	1	430.66	5.91	9.34	0.00	-14.13	6.34
37		-430.66	-5.91	-9.34	0.00	-13.88	11.38
8-	1	277.36	33.08	9.34	0.00	-14.13	49.65
39		-277.36	-33.08	-9.34	0.00	-13.88	49.58
10-	1	466.19	-19.52	-2.17	0.00	3.36	-41.01
41		-466.19	33.02	2.17	0.00	3.14	-37.79
12-	1	625.79	7.53	-2.17	0.00	3.36	10.40
43		-625.79	-7.53	2.17	0.00	3.14	12.20
14-	1	625.70	5.91	-2.17	0.00	3.36	6.34
45		-625.70	-5.91	2.17	0.00	3.14	11.38
16-	1	472.40	33.08	-2.17	0.00	3.36	49.65
47		-472.40	-33.08	2.17	0.00	3.14	49.58
18-	1	466.19	-19.52	2.17	0.00	-3.36	-41.01
49		-466.19	33.02	-2.17	0.00	-3.14	-37.79
20-	1	625.79	7.53	2.17	0.00	-3.36	10.40
51		-625.79	-7.53	-2.17	0.00	-3.14	12.20
22-	1	625.70	5.91	2.17	0.00	-3.36	6.34
53		-625.70	-5.91	-2.17	0.00	-3.14	11.38
24-	1	472.40	33.08	2.17	0.00	-3.36	49.65
55		-472.40	-33.08	-2.17	0.00	-3.14	49.58
26-	1	271.15	-19.52	-9.34	0.00	14.13	-41.01
57		-271.15	33.02	9.34	0.00	13.88	-37.79
28-	1	430.75	7.53	-9.34	0.00	14.13	10.40
59		-430.75	-7.53	9.34	0.00	13.88	12.20
30-	1	430.66	5.91	-9.34	0.00	14.13	6.34
61		-430.66	-5.91	9.34	0.00	13.88	11.38
32-	1	277.36	33.08	-9.34	0.00	14.13	49.65
63		-277.36	-33.08	9.34	0.00	13.88	49.58
33-	1	107.67	-21.59	8.19	0.00	-12.74	-38.17
34		-107.67	35.09	-8.19	0.00	-11.83	-46.84
35-	1	172.40	2.67	8.19	0.00	-12.74	2.12
36		-172.40	-2.67	-8.19	0.00	-11.83	5.89
37-	1	172.64	1.67	8.19	0.00	-12.74	1.58
38		-172.64	-1.67	-8.19	0.00	-11.83	3.42
39-	1	109.09	30.75	8.19	0.00	-12.74	41.22
40		-109.09	-30.75	-8.19	0.00	-11.83	51.01
41-	1	186.77	-21.59	-1.73	0.00	2.71	-38.17
42		-186.77	35.09	1.73	0.00	2.49	-46.84
43-	1	251.50	2.67	-1.73	0.00	2.71	2.12
44		-251.50	-2.67	1.73	0.00	2.49	5.89
45-	1	251.74	1.67	-1.73	0.00	2.71	1.58
46		-251.74	-1.67	1.73	0.00	2.49	3.42
47-	1	188.19	30.75	-1.73	0.00	2.71	41.22
48		-188.19	-30.75	1.73	0.00	2.49	51.01
49-	1	186.77	-21.59	1.73	0.00	-2.71	-38.17
50		-186.77	35.09	-1.73	0.00	-2.49	-46.84
51-	1	251.50	2.67	1.73	0.00	-2.71	2.12
52		-251.50	-2.67	-1.73	0.00	-2.49	5.89
53-	1	251.74	1.67	1.73	0.00	-2.71	1.58
54		-251.74	-1.67	-1.73	0.00	-2.49	3.42
55-	1	188.19	30.75	1.73	0.00	-2.71	41.22
56		-188.19	-30.75	-1.73	0.00	-2.49	51.01
57-	1	107.67	-21.59	-8.19	0.00	12.74	-38.17
58		-107.67	35.09	8.19	0.00	11.83	-46.84
59-	1	172.40	2.67	-8.19	0.00	12.74	2.12
60		-172.40	-2.67	8.19	0.00	11.83	5.89
61-	1	172.64	1.67	-8.19	0.00	12.74	1.58
62		-172.64	-1.67	8.19	0.00	11.83	3.42
63-	1	109.09	30.75	-8.19	0.00	12.74	41.22
64		-109.09	-30.75	8.19	0.00	11.83	51.01

2-	1	-8.02	82.74	0.00	0.00	0.00	62.78
4		8.02	97.26	0.00	0.00	0.00	-106.35
4-	1	-10.47	86.06	0.00	0.00	0.00	78.61
6		10.47	93.94	0.00	0.00	0.00	-102.27
6-	1	-13.88	88.25	0.00	0.00	0.00	80.57
8		13.88	91.75	0.00	0.00	0.00	-91.05
10-	1	-8.02	82.74	0.00	0.00	0.00	62.78
12		8.02	97.26	0.00	0.00	0.00	-106.35
12-	1	-10.47	86.06	0.00	0.00	0.00	78.61
14		10.47	93.94	0.00	0.00	0.00	-102.27
14-	1	-13.88	88.25	0.00	0.00	0.00	80.57
16		13.88	91.75	0.00	0.00	0.00	-91.05
18-	1	-8.02	82.74	0.00	0.00	0.00	62.78
20		8.02	97.26	0.00	0.00	0.00	-106.35
20-	1	-10.47	86.06	0.00	0.00	0.00	78.61
22		10.47	93.94	0.00	0.00	0.00	-102.27
22-	1	-13.88	88.25	0.00	0.00	0.00	80.57
24		13.88	91.75	0.00	0.00	0.00	-91.05
26-	1	-8.02	82.74	0.00	0.00	0.00	62.78
28		8.02	97.26	0.00	0.00	0.00	-106.35
28-	1	-10.47	86.06	0.00	0.00	0.00	78.61
30		10.47	93.94	0.00	0.00	0.00	-102.27
30-	1	-13.88	88.25	0.00	0.00	0.00	80.57
32		13.88	91.75	0.00	0.00	0.00	-91.05
2-	1	-6.61	75.96	0.00	0.00	0.00	21.38
10		6.61	104.04	0.00	0.00	0.00	-105.60
10-	1	-5.11	90.00	0.00	0.00	0.00	100.47
18		5.11	90.00	0.00	0.00	0.00	-100.47
18-	1	-6.61	104.04	0.00	0.00	0.00	105.60
26		6.61	75.96	0.00	0.00	0.00	-21.38
4-	1	-6.61	75.96	0.00	0.00	0.00	21.38
12		6.61	104.04	0.00	0.00	0.00	-105.60
12-	1	-5.11	90.00	0.00	0.00	0.00	100.47
20		5.11	90.00	0.00	0.00	0.00	-100.47
20-	1	-6.61	104.04	0.00	0.00	0.00	105.60
28		6.61	75.96	0.00	0.00	0.00	-21.38
6-	1	-6.61	75.96	0.00	0.00	0.00	21.38
14		6.61	104.04	0.00	0.00	0.00	-105.60
14-	1	-5.11	90.00	0.00	0.00	0.00	100.47
22		5.11	90.00	0.00	0.00	0.00	-100.47
22-	1	-6.61	104.04	0.00	0.00	0.00	105.60
30		6.61	75.96	0.00	0.00	0.00	-21.38
8-	1	-6.61	75.96	0.00	0.00	0.00	21.38
16		6.61	104.04	0.00	0.00	0.00	-105.60
16-	1	-5.11	90.00	0.00	0.00	0.00	100.47
24		5.11	90.00	0.00	0.00	0.00	-100.47
24-	1	-6.61	104.04	0.00	0.00	0.00	105.60
32		6.61	75.96	0.00	0.00	0.00	-21.38
33-	1	11.43	86.45	0.00	0.00	0.00	75.95
35		-11.43	93.55	0.00	0.00	0.00	-97.26
35-	1	6.57	87.77	0.00	0.00	0.00	82.94
37		-6.57	92.23	0.00	0.00	0.00	-96.33
37-	1	2.33	88.76	0.00	0.00	0.00	83.37
39		-2.33	91.24	0.00	0.00	0.00	-90.80
41-	1	11.43	86.45	0.00	0.00	0.00	75.95
43		-11.43	93.55	0.00	0.00	0.00	-97.26
43-	1	6.57	87.77	0.00	0.00	0.00	82.94
45		-6.57	92.23	0.00	0.00	0.00	-96.33
45-	1	2.33	88.76	0.00	0.00	0.00	83.37
47		-2.33	91.24	0.00	0.00	0.00	-90.80
49-	1	11.43	86.45	0.00	0.00	0.00	75.95
51		-11.43	93.55	0.00	0.00	0.00	-97.26
51-	1	6.57	87.77	0.00	0.00	0.00	82.94
53		-6.57	92.23	0.00	0.00	0.00	-96.33
53-	1	2.33	88.76	0.00	0.00	0.00	83.37
57-	1	11.43	86.45	0.00	0.00	0.00	75.95
59		-11.43	93.55	0.00	0.00	0.00	-97.26
59-	1	6.57	87.77	0.00	0.00	0.00	82.94
61		-6.57	92.23	0.00	0.00	0.00	-96.33
61-	1	2.33	88.76	0.00	0.00	0.00	83.37
63		-2.33	91.24	0.00	0.00	0.00	-90.80
33-	1	1.15	77.03	0.00	0.00	0.00	26.62
41		-1.15	102.97	0.00	0.00	0.00	-104.43
41-	1	0.71	90.00	0.00	0.00	0.00	98.57
49		-0.71	90.00	0.00	0.00	0.00	-98.57
49-	1	1.15	102.97	0.00	0.00	0.00	104.43
57		-1.15	77.03	0.00	0.00	0.00	-26.62
35-	1	1.15	77.03	0.00	0.00	0.00	26.62
43		-1.15	102.97	0.00	0.00	0.00	-104.43
43-	1	0.71	90.00	0.00	0.00	0.00	98.57
51		-0.71	90.00	0.00	0.00	0.00	-98.57
51-	1	1.15	102.97	0.00	0.00	0.00	104.43
59		-1.15	77.03	0.00	0.00	0.00	-26.62
37-	1	1.15	77.03	0.00	0.00	0.00	26.62
45		-1.15	102.97	0.00	0.00	0.00	-104.43
45-	1	0.71	90.00	0.00	0.00	0.00	98.57
53		-0.71	90.00	0.00	0.00	0.00	-98.57
53-	1	1.15	102.97	0.00	0.00	0.00	104.43
61		-1.15	77.03	0.00	0.00	0.00	-26.62
39-	1	1.15	77.03	0.00	0.00	0.00	26.62
47		-1.15	102.97	0.00	0.00	0.00	-104.43
47-	1	0.71	90.00	0.00	0.00	0.00	98.57
55		-0.71	90.00	0.00	0.00	0.00	-98.57
55-	1	1.15	102.97	0.00	0.00	0.00	104.43

63		-1.15	77.03	0.00	0.00	0.00	-26.62
34-	1	35.09	57.22	0.00	0.00	0.00	46.84
36		-35.09	62.78	0.00	0.00	0.00	-63.53
36-	1	32.41	59.17	0.00	0.00	0.00	57.63
38		-32.41	60.83	0.00	0.00	0.00	-62.61
38-	1	30.75	61.36	0.00	0.00	0.00	59.19
40		-30.75	58.64	0.00	0.00	0.00	-51.01
42-	1	35.09	57.22	0.00	0.00	0.00	46.84
44		-35.09	62.78	0.00	0.00	0.00	-63.53
44-	1	32.41	59.17	0.00	0.00	0.00	57.63
46		-32.41	60.83	0.00	0.00	0.00	-62.61
46-	1	30.75	61.36	0.00	0.00	0.00	59.19
48		-30.75	58.64	0.00	0.00	0.00	-51.01
50-	1	35.09	57.22	0.00	0.00	0.00	46.84
52		-35.09	62.78	0.00	0.00	0.00	-63.53
52-	1	32.41	59.17	0.00	0.00	0.00	57.63
54		-32.41	60.83	0.00	0.00	0.00	-62.61
54-	1	30.75	61.36	0.00	0.00	0.00	59.19
56		-30.75	58.64	0.00	0.00	0.00	-51.01
58-	1	35.09	57.22	0.00	0.00	0.00	46.84
60		-35.09	62.78	0.00	0.00	0.00	-63.53
60-	1	32.41	59.17	0.00	0.00	0.00	57.63
62		-32.41	60.83	0.00	0.00	0.00	-62.61
62-	1	30.75	61.36	0.00	0.00	0.00	59.19
64		-30.75	58.64	0.00	0.00	0.00	-51.01
34-	1	8.19	50.45	0.00	0.00	0.00	11.83
42		-8.19	69.55	0.00	0.00	0.00	-69.14
42-	1	6.45	60.00	0.00	0.00	0.00	66.65
50		-6.45	60.00	0.00	0.00	0.00	-66.65
50-	1	8.19	69.55	0.00	0.00	0.00	69.14
58		-8.19	50.45	0.00	0.00	0.00	-11.83
36-	1	8.19	50.45	0.00	0.00	0.00	11.83
44		-8.19	69.55	0.00	0.00	0.00	-69.14
44-	1	6.45	60.00	0.00	0.00	0.00	66.65
52		-6.45	60.00	0.00	0.00	0.00	-66.65
52-	1	8.19	69.55	0.00	0.00	0.00	69.14
60		-8.19	50.45	0.00	0.00	0.00	-11.83
38-	1	8.19	50.45	0.00	0.00	0.00	11.83
46		-8.19	69.55	0.00	0.00	0.00	-69.14
46-	1	6.45	60.00	0.00	0.00	0.00	66.65
54		-6.45	60.00	0.00	0.00	0.00	-66.65
54-	1	8.19	69.55	0.00	0.00	0.00	69.14
62		-8.19	50.45	0.00	0.00	0.00	-11.83
40-	1	8.19	50.45	0.00	0.00	0.00	11.83
48		-8.19	69.55	0.00	0.00	0.00	-69.14
48-	1	6.45	60.00	0.00	0.00	0.00	66.65
56		-6.45	60.00	0.00	0.00	0.00	-66.65
56-	1	8.19	69.55	0.00	0.00	0.00	69.14
64		-8.19	50.45	0.00	0.00	0.00	-11.83

APPENDIX-B

Two-Dimensional Steel Framed Structures

This appendix represents the results and analysis of six-storey and nine-storey structures with three bays that were not mentioned in chapter 5.

1- Six-storey three-bay frame

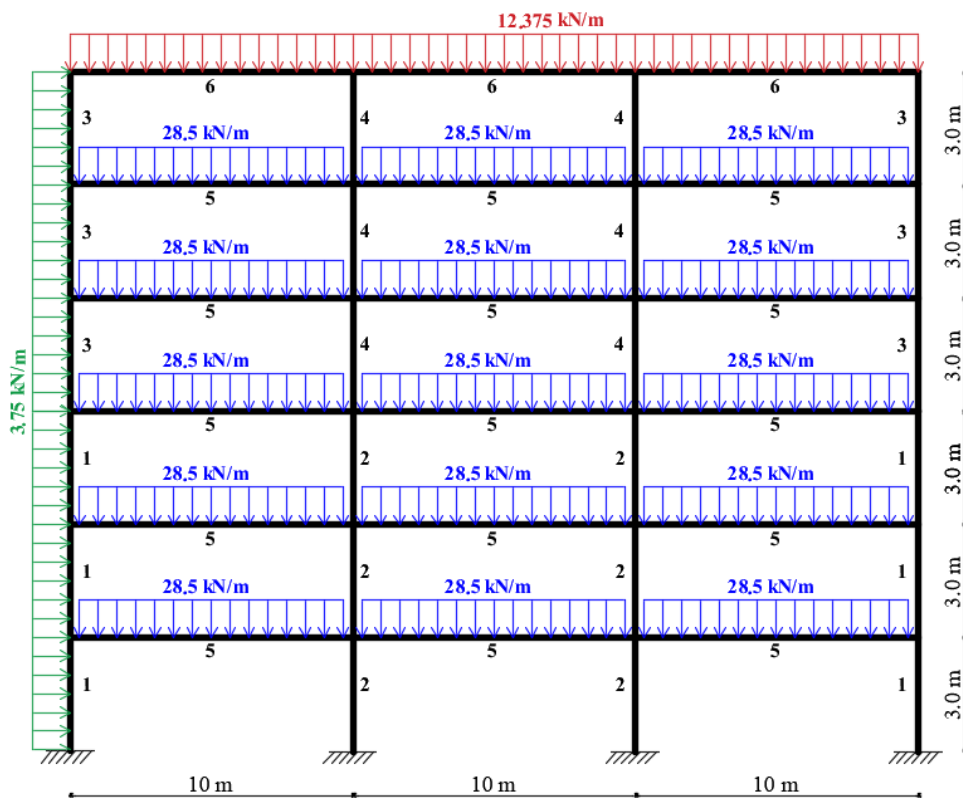


Figure B.1: representation of the layout and loading of the six-storey frame

1.1. Weight objective function:

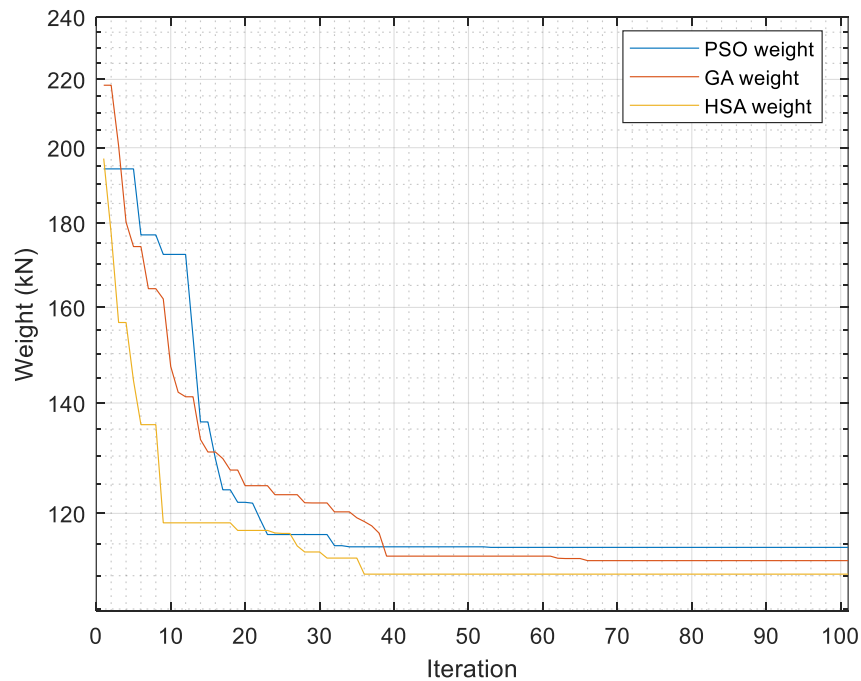


Figure B.2: Comparison of Search history between GA, HSA and PSO for the six-storey frame (W-objective function)

Table B.1: Optimisation output for the six-storey frame (W-objective function)

Element	Section		
	GA	PSO	HSA
Column (1)	203 x 203 x 46	203 x 203 x 46	203 x 203 x 46
Column (2)	203 x 203 x 52	254 x 254 x 73	203 x 203 x 52
Column (3)	152 x 152 x 37	152 x 152 x 37	152 x 152 x 37
Column (4)	152 x 152 x 37	152 x 152 x 37	152 x 152 x 37
Beam (5)	406 x 140 x 46	406 x 140 x 46	406 x 140 x 46
Beam (6)	406 x 140 x 39	356 x 127 x 33	254 x 146 x 31
Total optimum weight of the structure (kN)	111.68	113.71	109.31
Total optimum Embodied Energy of the structure (MJ)	243.719 X 10 ³	248.033 X 10 ³	238.518X10 ³
Total optimum Embodied Carbon of the structure (kgCO ₂ e)	19.405 X 10 ³	19.753 X 10 ³	18.992X10 ³
Total cost of the structure (£)	21210.30	23542.30	20725.90
Number of iteration for best result	64.00	32.00	33.00
Time of finding the best (sec)	13.90	14.13	10.41
Time of running (sec)	22.00	22.00	32.00

1.2. Embodied Energy objective function:

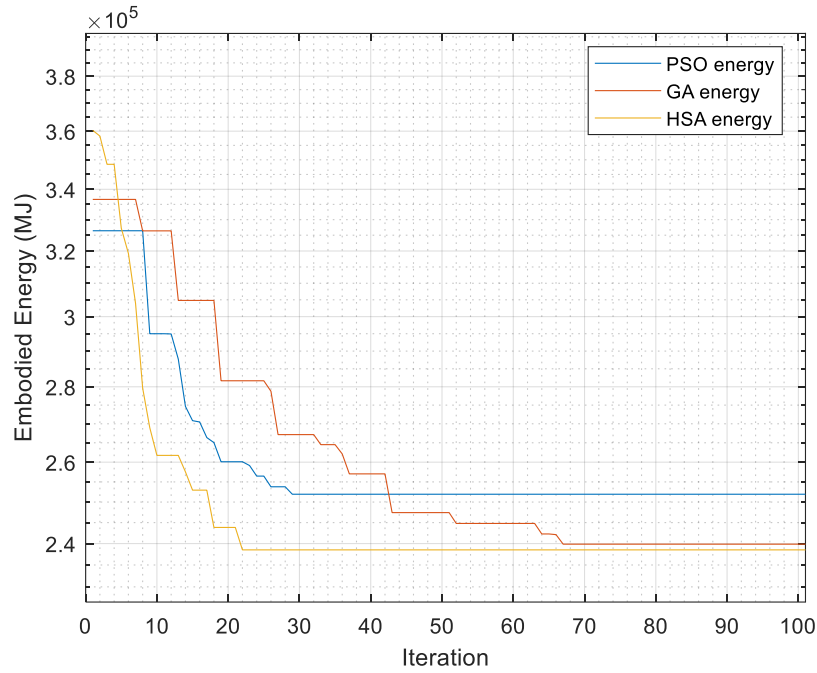


Figure B-3: Comparison of Search history between GA, HSA and PSO for the six-storey frame (EE-objective function)

Table B.2: Optimisation output for the six-storey frame (EE-objective function)

Element	Section		
	GA	PSO	HSA
Column (1)	203 x 203 x 46	254 x 254 x 73	203 x 203 x 46
Column (2)	203 x 203 x 52	203 x 203 x 52	203 x 203 x 52
Column (3)	152 x 152 x 37	203 x 203 x 46	152 x 152 x 37
Column (4)	152 x 152 x 37	152 x 152 x 30	152 x 152 x 37
Beam (5)	406 x 140 x 46	356 x 171 x 45	406 x 140 x 46
Beam (6)	356 x 127 x 33	406 x 140 x 39	254 x 146 x 31
Total optimum weight of the structure (kN)	109.91	115.42	109.31
Total optimum Embodied Energy of the structure (MJ)	239.861 X 10 ³	251.931 X 10 ³	238.518 X 10 ³
Total optimum Embodied Carbon of the structure (kgCO ₂ e)	19.098 X 10 ³	20.057 X 10 ³	18.992 X 10 ³
Total cost of the structure (£)	20879.60	21985.10	20725.90
Number of iteration for best result	67.00	29.00	22.00
Time of finding the best (sec)	15.43	16.28	6.99
Time of running (sec)	23.00	26.00	32.00

1.3. Multi objective function:

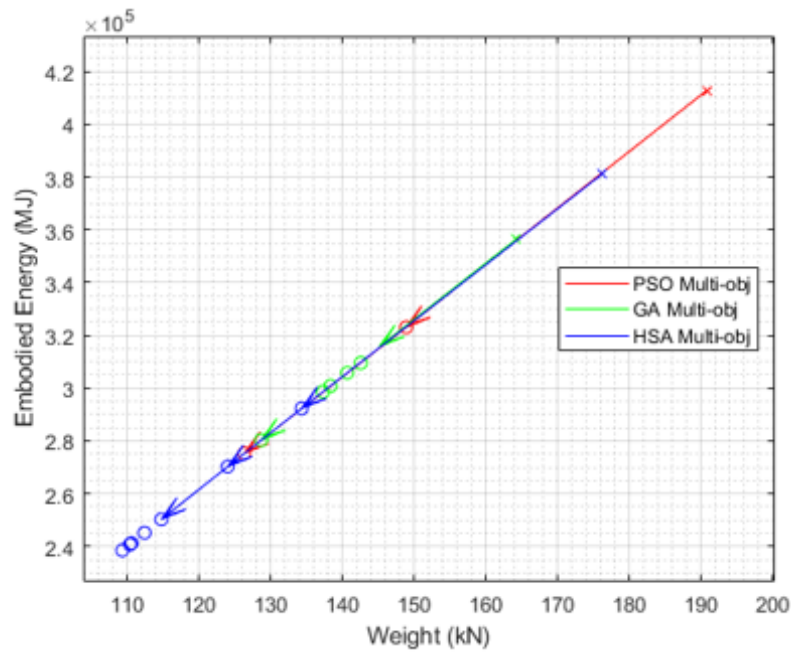


Figure B-4: Comparison of Search history between GA, HSA and PSO for the six-storey frame (Multi-objective function)

Table B-3: Optimisation output for the six-storey frame (Multi-objective function)

Element	Section		
	GA	PSO	HSA
Column (1)	254 x 254 x 73	203 x 203 x 46	203 x 203 x 46
Column (2)	254 x 254 x 73	203 x 203 x 52	203 x 203 x 52
Column (3)	203 x 203 x 52	203 x 203 x 46	152 x 152 x 37
Column (4)	203 x 203 x 46	203 x 203 x 60	152 x 152 x 37
Beam (5)	406 x 140 x 46	457 x 152 x 52	406 x 140 x 46
Beam (6)	457 x 152 x 52	254 x 146 x 37	254 x 146 x 31
Total optimum weight of the structure (kN)	128.66	126.31	109.30
Total optimum Embodied Energy of the structure (MJ)	280.308X10 ³	275.266X10 ³	238.518X10 ³
Total optimum Embodied Carbon of the structure (kgCO ₂ e)	22.336X10 ³	21.931X10 ³	18.992X10 ³
Total cost of the structure (£)	23866.90	23542.30	20725.90
Number of iteration for best result	7.00	10.00	34.00
Time of finding the best (sec)	10.07	48.00	32.82
Time of running (sec)	78.00	334.00	156.00

1.4. Comparasion

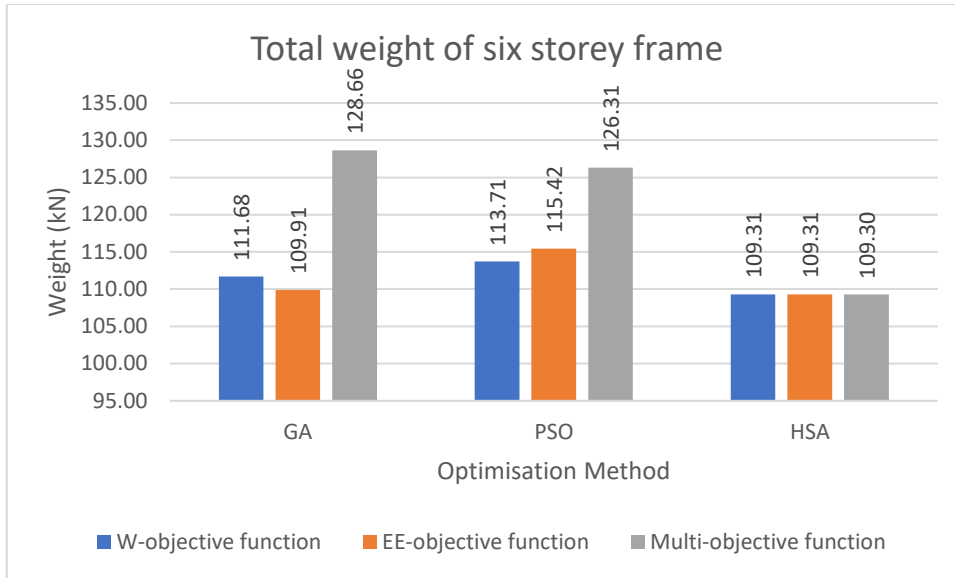


Figure B-5: Schematic representation of the difference in weight among the three objective functions

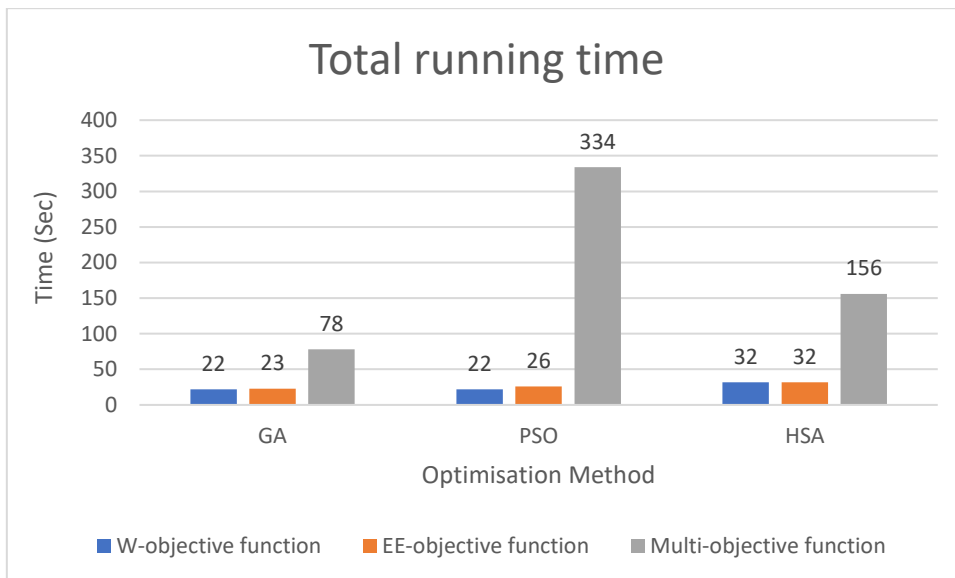


Figure B-6: Running time average for the three algorithms for the three objective functions.

2- Nine-storey three-bay frame

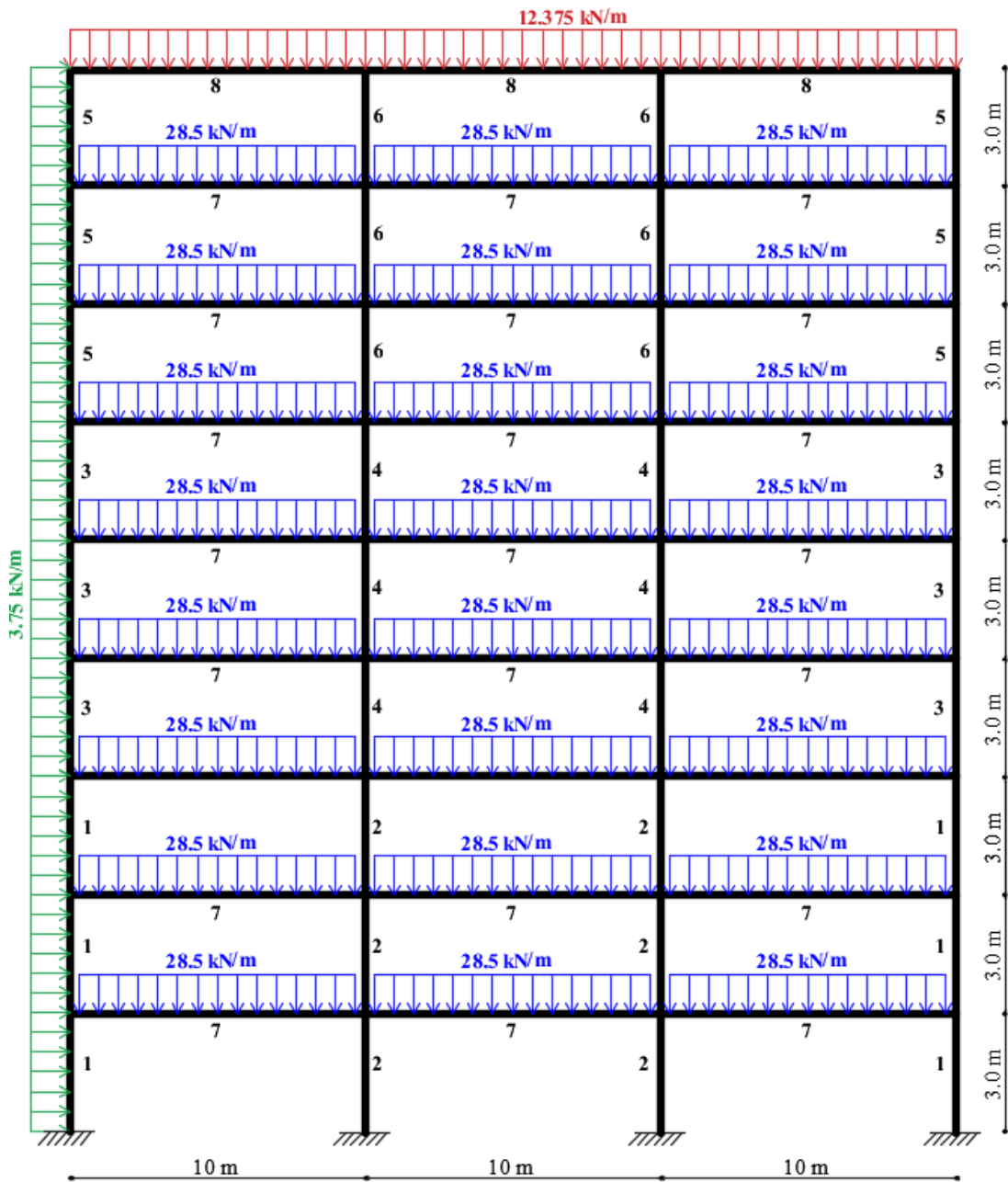


Figure B-7: Layout and loading of the Nine-storey frame.

2.1 Weight objective function:

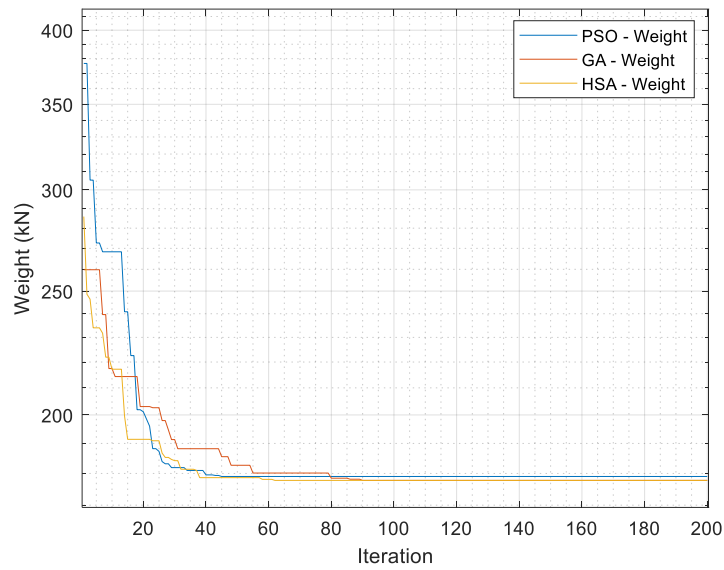


Figure B-8: Comparison of Search history between GA, HSA and PSO for the nine-storey frame (W-objective function)

Table B-4: Optimisation output for the nine-storey frame (W-objective function)

Element	Section		
	GA	PSO	HSA
Column (1)	254 x 254 x 73	254 x 254 x 73	203 x 203 x 71
Column (2)	254 x 254 x 73	254 x 254 x 73	254 x 254 x 73
Column (3)	203 x 203 x 52	203 x 203 x 52	203 x 203 x 52
Column (4)	203 x 203 x 52	203 x 203 x 52	203 x 203 x 52
Column (5)	152 x 152 x 44	152 x 152 x 37	203 x 203 x 46
Column (6)	152 x 152 x 30	152 x 152 x 44	152 x 152 x 30
Beam (7)	406 x 140 x 46	406 x 140 x 46	406 x 140 x 46
Beam (8)	203 x 133 x 30	406 x 140 x 39	203 x 133 x 30
Total optimum weight of the structure (kN)	177.76	181.72	177.76
Total optimum Embodied Energy of the structure (MJ)	387.394 X 10 ³	396.061 X 10 ³	387.393 X 10 ³
Total optimum Embodied Carbon of the structure (kgCO ₂ e)	30.865 X 10 ³	31.554 X 10 ³	30.865 X 10 ³
Number of iteration for best result	78.00	40.00	67.37
Total cost of the structure (£)	33132.90	33914.30	33131.10
Time of finding the best (sec)	36.18	25.79	67.37
Time of running (sec)	59	34	127

2.2. Embodied Energy objective function:

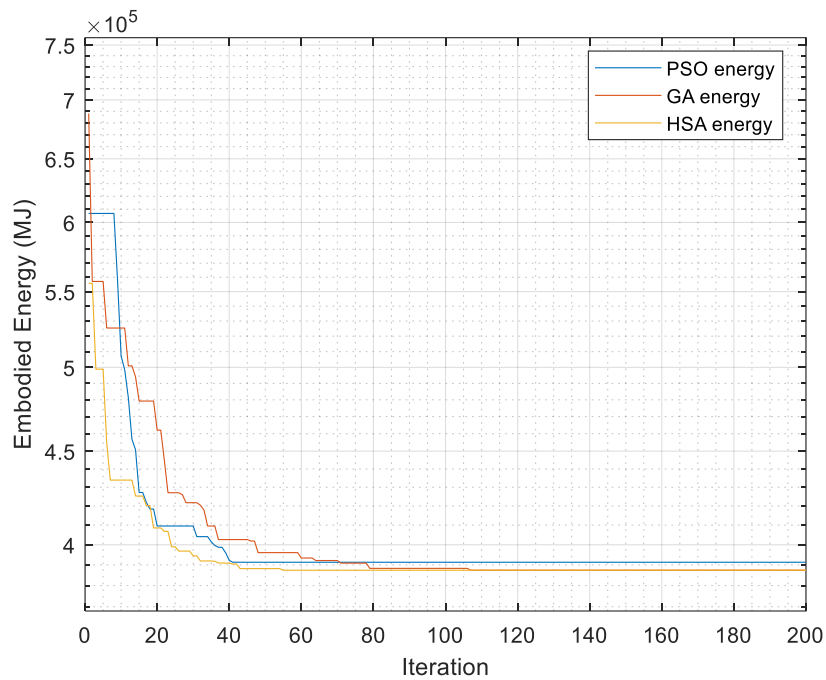


Figure B-9: Comparison of Search history between GA, HSA and PSO for the nine-storey frame (EE-objective function)

Table B-5: Optimisation output for the nine-storey frame (EE-objective function)

Element	Section		
	GA	PSO	HSA
Column (1)	254 x 254 x 73	203 x 203 x 71	203 x 203 x 71
Column (2)	254 x 254 x 73	254 x 254 x 73	254 x 254 x 73
Column (3)	203 x 203 x 52	203 x 203 x 52	203 x 203 x 52
Column (4)	203 x 203 x 52	203 x 203 x 52	203 x 203 x 52
Column (5)	152 x 152 x 44	152 x 152 x 37	203 x 203 x 46
Column (6)	152 x 152 x 30	152 x 152 x 44	152 x 152 x 30
Beam (7)	406 x 140 x 46	406 x 140 x 46	406 x 140 x 46
Beam (8)	203 x 133 x 30	356 x 127 x 33	203 x 133 x 30
Total optimum weight of the structure (kN)	177.76	179.57	177.76
Total optimum Embodied Energy of the structure (MJ)	387.394×10^3	391.299×10^3	387.393×10^3
Total optimum Embodied Carbon of the structure (kgCO _{2e})	30.865×10^3	31.177×10^3	30.865×10^3
Number of iteration for best result	107.00	41.00	55.00
Total cost of the structure (£)	33132.90	33417.50	33131.1
Time of finding the best (sec)	51.31	32.26	35.23
Time of running (sec)	75	46	128

2.3. Multi objective function:

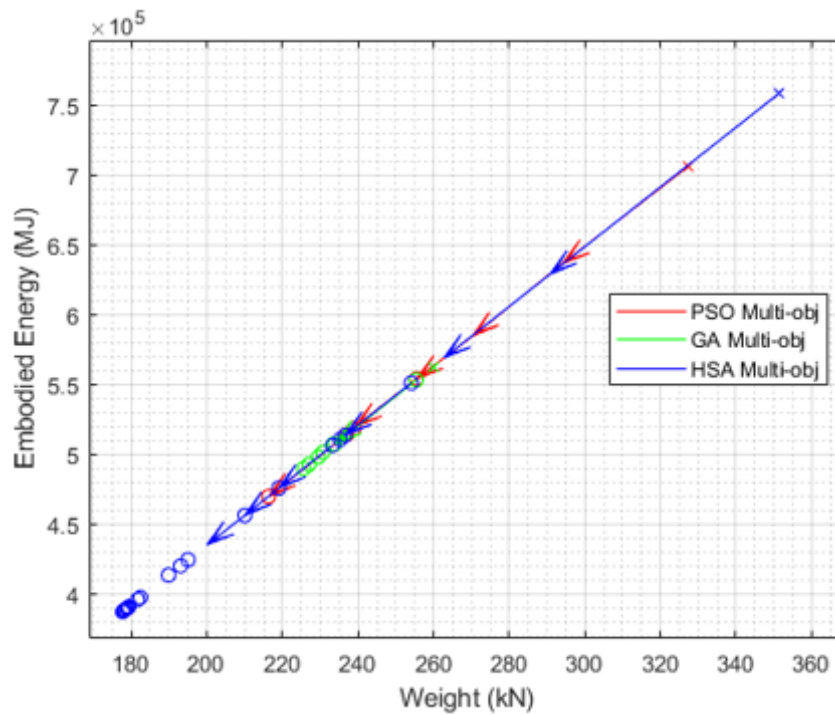


Figure B-10: Comparison of Search history between GA, HSA and PSO for the nine-storey frame (Multi-objective function)

Table B-6: Optimisation output for the nine-storey frame (Multi-objective function)

Element	Section		
	GA	PSO	HSA.
Column (1)	203 x 203 x 71	254 x 254 x 73	203 x 203 x 71
Column (2)	203 x 203 x 86	254 x 254 x 132	254 x 254 x 73
Column (3)	305 x 305 x 137	203 x 203 x 46	203 x 203 x 52
Column (4)	356 x 368 x 153	203 x 203 x 100	203 x 203 x 52
Column (5)	152 x 152 x 51	203 x 203 x 71	203 x 203 x 46
Column (6)	203 x 203 x 52	203 x 203 x 100	152 x 152 x 30
Beam (7)	406 x 140 x 46	406 x 140 x 46	406 x 140 x 46
Beam (8)	406 x 178 x 54	305 x 165 x 40	203 x 133 x 30
Total optimum weight of the structure (kN)	225.65	216.34	177.76
Total optimum Embodied Energy of the structure (MJ)	489.927X10 ³	469.785X10 ³	387.393X10 ³
Total optimum Embodied Carbon of the structure (kgCO ₂ e)	39.104X10 ³	37.494X10 ³	30.865X10 ³
Total cost of the structure (£)	39876.5	38306.9	33131.1
Time of finding the best (sec)	10	16	127
Time of running (sec)	154	587	291

2.4. Comparasion

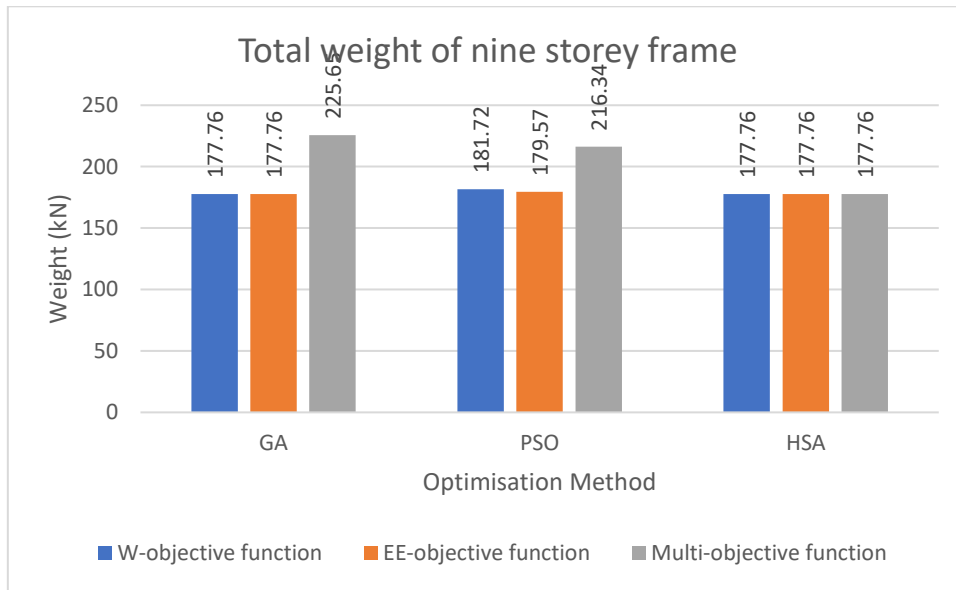


Figure B-11: Schematic representation of the difference in weight among the three objective functions

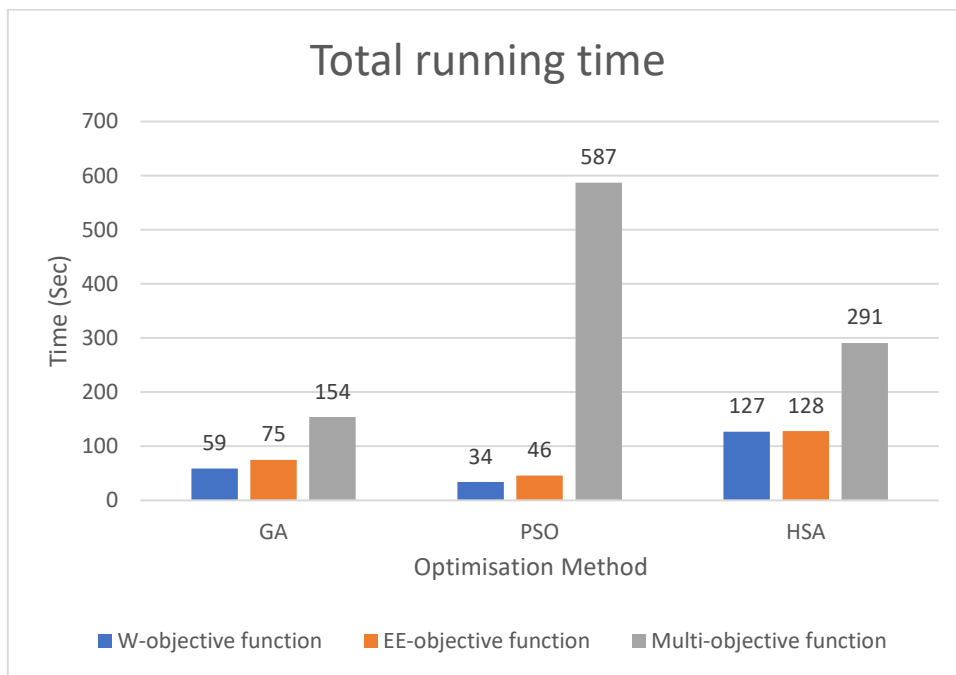


Figure B-12: Running time average for the three algorithms for the three objective functions.

APPENDIX-C

25/04/22 16:19 C:\Users\CVE3ALKHAA\OneDrive - Notti... 1 of 23

% Reading data from Excel and calculate parameters

```
clear all

nodeCoordinates= xlsread('Input3D.xlsm', 'A21:D900');
elementNodes=xlsread('Input3D.xlsm', 'F21:H900');
supportNode=xlsread('Input3D.xlsm', 'R21:R900'); % reading the supports number
loadjoints=xlsread('Input3D.xlsm', 'W21:Z900'); % Nodal load
loadmembers=xlsread('Input3D.xlsm', 'AE21:AF900'); % Distributed load
loadpoints=xlsread('Input3D.xlsm', 'AK21:AM900'); % point load
numberElements=xlsread('Input3D.xlsm', 'E6:E6');
DisloadDirection=xlsread('Input3D.xlsm', 'AD21:AD900');
PointloadDirection=xlsread('Input3D.xlsm', 'AJ21:AJ900');
steelGrade= xlsread('Input3D.xlsm', 'J5:J5');
gammaM0=xlsread('Input3D.xlsm', 'J7:J7'); % Partial factor (resistance of cross-
section whatever the class is)
gammaM1=xlsread('Input3D.xlsm', 'J8:J8'); % Partial factor (resistance of members to
instability)
V= xlsread('Input3D.xlsm', 'J9:J9'); % READING THE Passion's ratio in elastic stage
(V)
E= xlsread('Input3D.xlsm', 'N21:N21'); % READING THE MODULUS OF ELASTICITY (E)
beta= xlsread('Input3D.xlsm', 'J10:J10');
ita= xlsread('Input3D.xlsm', 'J6:J6');
lambdaDashLT0=xlsread('Input3D.xlsm', 'J11:J11');
C1=xlsread('Input3D.xlsm', 'J12:J12'); %factor depends on shape of Bending moment
pointloadADirection=xlsread('Input3D.xlsm', 'AE21:AE900');
LiveLoad=xlsread('Input3D.xlsm', 'AG21:AG900'); % NEEDED ONLY FOR VERTICAL
DEFLECTION*****++++
elementDof=[];
elementDof(:,1)=elementNodes(:,1)*6-5;
elementDof(:,2)=elementNodes(:,1)*6-4;
elementDof(:,3)=elementNodes(:,1)*6-3;
elementDof(:,4)=elementNodes(:,1)*6-2;
elementDof(:,5)=elementNodes(:,1)*6-1;
elementDof(:,6)=elementNodes(:,1)*6;
elementDof(:,7)=elementNodes(:,2)*6-5;
elementDof(:,8)=elementNodes(:,2)*6-4;
elementDof(:,9)=elementNodes(:,2)*6-3;
elementDof(:,10)=elementNodes(:,2)*6-2;
elementDof(:,11)=elementNodes(:,2)*6-1;
elementDof(:,12)=elementNodes(:,2)*6;
%%% To read PRESCRIBED DOF (the supports degree of freedom to find the displacement
and reactions later) %%%%%%%%%
prescribedDof0=[];
prescribedDof0(:,1)=supportNode(:,1)*6-5;
prescribedDof0(:,2)=supportNode(:,1)*6-4;
prescribedDof0(:,3)=supportNode(:,1)*6-3;
prescribedDof0(:,4)=supportNode(:,1)*6-2;
prescribedDof0(:,5)=supportNode(:,1)*6-1;
prescribedDof0(:,6)=supportNode(:,1)*6;
prescribedDof00=prescribedDof0';
SN=size(supportNode); % SN is the number of notes that have supports
SD=SN(1)*6; % Total degree of freedom at the nodes which have supports / 6 is the Dof
each node
prescribedDof=reshape(prescribedDof00,[SD 1]);
```

```

numberNodes=size(nodeCoordinates,1);
GDof=6*numberNodes;
xx=nodeCoordinates(:,2);
yy=nodeCoordinates(:,3);
zz=nodeCoordinates(:,4);
%Length and angle of each member
SectionType=elementNodes(:,3);
memberlength=zeros(numberElements,3);
for i=1:numberElements
    indice=elementNodes(i,:) ; % This is only to find the length of each member below
    % *****✓
    *****
    nn=length(indice);
    xa=xx(indice(2))-xx(indice(1));
    ya=yy(indice(2))-yy(indice(1));
    za=zz(indice(2))-zz(indice(1));
    length_elemen=sqrt(xa*xa+ya*ya+za*za);
    ll=length_elemen;
    memberlength(i,1)=ll;
end
distributedload=zeros(numberElements,12);
ss=size(loadmembers);
for i=1:ss(1)
    rr=loadmembers(i,1); % to find the element number
    L=memberlength(rr,1);
    P=loadmembers(loadmembers(:,1)==rr,2);
    % Load Direction
    % Xy ... member X and load in y direction
    if DisloadDirection(i,1)==1 %condition for the member direction and load✓
direction,
    distributedload(rr,2)=-P*L/2;
    distributedload(rr,8)=-P*L/2;
    distributedload(rr,6)=-P*L^2/12; % I changed sign
    distributedload(rr,12)=P*L^2/12; % I changed sign
    end
    % Xz ... member X and load in z direction
    if DisloadDirection(i,1)==2 %condition for the member direction and load✓
direction,
    distributedload(rr,3)=-P*L/2;
    distributedload(rr,9)=-P*L/2;
    distributedload(rr,5)=P*L^2/12;
    distributedload(rr,11)=-P*L^2/12;
    end
    % Yx ... member Y and load in x direction
    if DisloadDirection(i,1)==3 %condition for the member direction and load✓
direction,
    distributedload(rr,1)=-P*L/2;
    distributedload(rr,7)=-P*L/2;
    distributedload(rr,6)=P*L^2/12; % Changed this sign temporary *****-***-***-✓
    ***
    distributedload(rr,12)=-P*L^2/12; % Changed this sign temporary ***-***-***-✓
    *****
    end
    % Yz ... member Y and load in z direction
    if DisloadDirection(i,1)==4 %condition for the member direction and load✓

```



```

        pointload(rr,2)=-(P*b^2/L^3)*(3*a+b);
        pointload(rr,8)=-(P*a^2/L^3)*(a+3*b);
        pointload(rr,6)=-(P*a*b^2/L^2); % I HAVE CHANGED THIS SIGN
        pointload(rr,12)=(P*a^2*b/L^2); % I HAVE CHANGED THIS SIGN
    end
    if a==b;
        pointload(rr,2)=-P/2;
        pointload(rr,8)=-P/2;
        pointload(rr,6)=-P*L/8; % I HAVE CHANGED THIS SIGN
        pointload(rr,12)=P*L/8; % I HAVE CHANGED THIS SIGN
    end
end
% Xz ... member X and load in z direction
if PointloadDirection(i,1)==2; %condition for the member direction and load✓
direction,
    if a>b || a<b
        pointload(rr,3)=-(P*b^2/L^3)*(3*a+b);
        pointload(rr,9)=-(P*a^2/L^3)*(a+3*b);
        pointload(rr,5)=(P*a*b^2/L^2);
        pointload(rr,11)=-(P*a^2*b/L^2);
    end
    if a==b;
        pointload(rr,3)=-P/2;
        pointload(rr,9)=-P/2;
        pointload(rr,5)=P*L/8;
        pointload(rr,11)=-P*L/8;
    end
end
% Yx ... member Y and load in x direction
if PointloadDirection(i,1)==3; %condition for the member direction and load✓
direction,
    if a>b || a<b
        pointload(rr,1)=-(P*b^2/L^3)*(3*a+b);
        pointload(rr,7)=-(P*a^2/L^3)*(a+3*b);
        pointload(rr,6)=(P*a*b^2/L^2); % I have changed this sign
        pointload(rr,12)=- (P*a^2*b/L^2); % I have changed this sign
    end
    if a==b;
        pointload(rr,1)=-P/2;
        pointload(rr,7)=-P/2;
        pointload(rr,6)=P*L/8; % I have changed this sign
        pointload(rr,12)=-P*L/8; % I have changed this sign
    end
end
% Yz ... member Y and load in z direction
if PointloadDirection(i,1)==4; %condition for the member direction and load✓
direction,
    if a>b || a<b
        pointload(rr,3)=-(P*b^2/L^3)*(3*a+b);
        pointload(rr,9)=-(P*a^2/L^3)*(a+3*b);
        pointload(rr,4)=-(P*a*b^2/L^2);
        pointload(rr,10)=(P*a^2*b/L^2);
    end
    if a==b;
        pointload(rr,3)=-P/2;

```

```

        pointload(rr,9)=-P/2;
        pointload(rr,4)=-P*L/8;
        pointload(rr,10)=P*L/8;
    end
end
% Zx ... member Z and load in x direction
if PointloadDirection(i,1)==5; %condition for the member direction and load
direction,
    if a>b || a<b
        pointload(rr,1)--(P*b^2/L^3)*(3*a+b);
        pointload(rr,7)--(P*a^2/L^3)*(a+3*b);
        pointload(rr,5)--(P*a*b^2/L^2);
        pointload(rr,11)=(P*a^2*b/L^2);
    end
    if a==b;
        pointload(rr,1)=-P/2;
        pointload(rr,7)=-P/2;
        pointload(rr,5)=-P*L/8;
        pointload(rr,11)=P*L/8;
    end
end
% Zy ... member Z and load in y direction
if PointloadDirection(i,1)==6 ; %condition for the member direction and load
direction,
    if a>b || a<b
        pointload(rr,2)--(P*b^2/L^3)*(3*a+b);
        pointload(rr,8)--(P*a^2/L^3)*(a+3*b);
        pointload(rr,4)=(P*a*b^2/L^2);
        pointload(rr,10)--(P*a^2*b/L^2);
    end
    if a==b;
        pointload(rr,2)=-P/2;
        pointload(rr,8)=-P/2;

        pointload(rr,4)=P*L/8;
        pointload(rr,10)=-P*L/8;
    end
end
end
% pointloadReaction=pointload;
pointloadAction=(pointload); %%% change directions in order to get the actual load on
structure
ll=memberlength(:,1);
forceVector=zeros(GDof,1);
ww=size(elementDof); % to find the ww1 which is the total number of Dof
ww1=ww(1)*ww(2); %ww is the size of all Dof of all elements, (all elemens=6*number of
elements)
indx=reshape(elementDof,[ww1 1]);
aa=reshape(distributedloadAction,[ww1 1]);
bb=reshape(pointloadAction,[ww1 1]);
indx(:,2)=aa;
indx(:,3)=bb;
for i=1:GDof
    cc(i,1)=sum(indx(indx(:,1)==i,2));
    cc(i,2)=sum(indx(indx(:,1)==i,3));

```

```

end
cc(:,3)=reshape(Nodalload',[GDof 1]);
cc(:,4)=cc(:,1)+cc(:,2)+cc(:,3);
forceVector=cc(:,4);
Load1=[1:numberElements]'; % sorting out the loads of all members in one matrix✓
""named Load1""
Load=loadmembers; % reading the actual applied loads from excel that already been reed✓
in FrameAnalysis2D
for ii=1:length(Load) % load to match and fill the matrix
    c=Load(ii,1);
    Load1(c,2)=Load(ii,2);
end
end
G = E/(2*(1+V));% shear modulus
L0=memberlength(:,1);
epsilon=sqrt(235/steelGrade);
save('Excel_variables.mat')

```

% Compute forces

```

function NewParams = computeForce(EA,EIy,EIz,GJ,ForceParams)
elementDof = ForceParams.elementDof;
l1 = ForceParams.l1;
loadpoints = ForceParams.loadpoints;
numberElements = ForceParams.numberElements;
GDof = ForceParams.GDof;
% elementNodes = ForceParams.elementNodes;
loadmembers = ForceParams.loadmembers;
activeDof = ForceParams.activeDof;
forceVector = ForceParams.forceVector;
memberlength = ForceParams.memberlength;
elementNodes = ForceParams.elementNodes;
xx = ForceParams.xx;
yy = ForceParams.yy;
zz = ForceParams.zz;
nodeCoordinates = ForceParams.nodeCoordinates;
DisloadDirection = ForceParams.DisloadDirection;
% steelGrade = ForceParams.steelGrade;
indexes=ForceParams.indexes;
KgElement = zeros(12,12,numberElements);
ki = zeros(12,12,numberElements);
T = zeros(12,12,numberElements);
Tt = zeros(12,12,numberElements);
L=zeros(1,numberElements);
% Tt = T;
stiffness = zeros(GDof);
% took constants out of loop
for i=1:numberElements
    indice=elementNodes(i,:) ; % This is only to find the length of each member below
    nn=length(indice);
    xa=xx(indice(2))-xx(indice(1));
    ya=yy(indice(2))-yy(indice(1));
    za=zz(indice(2))-zz(indice(1));
    length_eleme=sqrt(xa*xa+ya*ya+za*za);

```

```

L(i)=length_elemen;
LL(:, :, i)=L;
x1=nodeCoordinates(indice(1),2);
y1=nodeCoordinates(indice(1),3);
z1=nodeCoordinates(indice(1),4);
x2=nodeCoordinates(indice(2),2);
y2=nodeCoordinates(indice(2),3);
z2=nodeCoordinates(indice(2),4);
k1 = EA(i)/L(i); % 12EA/L
k2 = 12*EIz(i)/L(i)^3; %12*EIz/L^3
k3 = 6*EIz(i)/L(i)^2; %6*EIz/L^2
k4 = 4*EIz(i)/L(i); %4*EIz/L
k5 = 2*EIz(i)/L(i); %2*EIz/L
k6 = 12*EIy(i)/L(i)^3; % 12*EIy/L^3
k7 = 6*EIy(i)/L(i)^2; %6*EIy/L^2
k8 = 4*EIy(i)/L(i); %4*EIy/L
k9 = 2*EIy(i)/L(i); %2*EIy/L
k10 = GJ(i)/L(i); %G*J/L
a=[k1 0 0; 0 k2 0; 0 0 k6];
b=[ 0 0 0;0 0 k3; 0 -k7 0];
c=[k10 0 0;0 k8 0; 0 0 k4];
d=[-k10 0 0;0 k9 0;0 0 k5];
KI = [a b -a b;b' c (-b)' d; (-a)' -b a -b;b' d' (-b)' c];
if x1==x2 && y1==y2;
    if z2 > z1
        Lambda = [0 0 1 ; 0 1 0 ; -1 0 0];
    else
        Lambda = [0 0 -1 ; 0 1 0 ; 1 0 0];
    end
else
    CXx = (x2-x1)/L(i);
    CYx = (y2-y1)/L(i);
    CZx = (z2-z1)/L(i);
    D = sqrt(CXx*CXx + CYx*CYx);
    CXy = -CYx/D;
    CYy = CXx/D;
    CZy = 0;
    CXz = -CXx*CZx/D;
    CYz = -CYx*CZx/D;
    CZz = D;
    Lambda = [CXx CYx CZx ;CXy CYy CZy ;CXz CYz CZz];
end
TI(1:3,1:3) = Lambda;
TI(4:6,4:6) = Lambda;
TI(7:9,7:9) = Lambda;
TI(10:12,10:12) = Lambda;
T(:, :, i)=TI;
Tt(:, :, i)=TI';
KgElement(:, :, i)=TI'*KI*TI;
ki(:, :, i) = KI;
stiffness(elementDof(i, :),elementDof(i, :))= stiffness(elementDof(i, :),elementDof(i, :)) ✓
+KgElement(:, :, i);
end
% Displacement Calculations
U=stiffness(activeDof,activeDof)\forceVector(activeDof);

```

```

displacements=zeros(GDof,1);
displacements(activeDof)=U;
ggg=displacements(1:6:end);
kkk=displacements(indexes);
% Internal forces of each element
distributedloadReactions=zeros(numberElements,12);
ss=size(loadmembers);
for i=1:ss(1)
    rr=loadmembers(i,1); % to find the element number
    L=memberlength(rr,1);
    P=loadmembers(loadmembers(:,1)==rr,2);
    % Load Direction
    % Xy ... member X and load in y direction
    if DisloadDirection(i,1)==1 %condition for the member direction and load✓
direction,
        distributedloadReactions(rr,2)=-P*L/2; % IT WAS -
        distributedloadReactions(rr,8)=-P*L/2; % IT WAS -

        distributedloadReactions(rr,6)=-P*L^2/12; % CHANGED TRIAL -----*****✓
% IT WAS -
        distributedloadReactions(rr,12)=P*L^2/12; % CHANGED TRIAL -----*****✓
% IT WAS +
        end
    % Xz ... member X and load in z direction
    if DisloadDirection(i,1)==2 %condition for the member direction and load✓
direction,
        distributedloadReactions(rr,3)=-P*L/2;
        distributedloadReactions(rr,9)=-P*L/2;
        distributedloadReactions(rr,5)=-P*L^2/12; % IT WAS +
        distributedloadReactions(rr,11)=P*L^2/12; % IT WAS -
        end
    % Yx ... member Y and load in x direction
    if DisloadDirection(i,1)==3 %condition for the member direction and load✓
direction,
        distributedloadReactions(rr,1)=P*L/2; % changed , which was 1
        distributedloadReactions(rr,7)=P*L/2; % changed , which was 7
        distributedloadReactions(rr,6)=P*L^2/12; % CHANGED TRIAL -----***** %✓
IT WAS +
        distributedloadReactions(rr,12)=-P*L^2/12; % CHANGED TRIAL -----*****✓
% IT WAS -
        end
    % Yz ... member Y and load in z direction
    if DisloadDirection(i,1)==4 %condition for the member direction and load✓
direction,
        distributedloadReactions(rr,3)=-P*L/2; % IT WAS -
        distributedloadReactions(rr,9)=-P*L/2; % IT WAS -
        distributedloadReactions(rr,4)=-P*L^2/12;
        distributedloadReactions(rr,10)=P*L^2/12;
        end
    % Zx ... member Z and load in x direction
    if DisloadDirection(i,1)==5 %condition for the member direction and load✓
direction,
        distributedloadReactions(rr,2)=-P*L/2; % changed , which was 1
        distributedloadReactions(rr,8)=-P*L/2; % changed , which was 7
        distributedloadReactions(rr,5)=-P*L^2/12;

```



```

distributedloadReactions(rr,11)=P*L^2/12;
end
% Zy ... member Z and load in y direction
if DisloadDirection(i,1)==6 %condition for the member direction and load
direction,
distributedloadReactions(rr,2)=-P*L/2; % IT WAS - >>>2 -
distributedloadReactions(rr,8)=-P*L/2; % IT WAS - >>>8 -
distributedloadReactions(rr,6)=-P*L^2/12; %<<<<<<<<<<<< IT WAS 4 >>>6 -
distributedloadReactions(rr,12)=P*L^2/12; %<<<<<<<<<<<< IT WAS 10 >>>>12 +
end
end
%Distributed load on each member
%Point load on each member
pointloadReactions=zeros(numberElements,12);
jj=size(loadpoints);
for i=1:jj(1)
L=memberlength(i,1);
rr=loadpoints(i,1); % to find the elemnet number from matrix
L=memberlength(rr,1);
P=loadpoints(loadpoints(:,1)==rr,2);
a=loadpoints(loadpoints(:,1)==rr,3);
b=L-a;
% *** Load direction conditions *****
% Xy ... member X and load in y direction
if PointloadDirection(i,1)==1; % %condition for the member direction and
load direction,
if a>b || a<b
pointloadReactions(rr,2)=-(P*b^2/L^3)*(3*a+b);
pointloadReactions(rr,8)=-(P*a^2/L^3)*(a+3*b);
pointloadReactions(rr,6)=-(P*a*b^2/L^2); % CHANGED TRIAL -----
****-----
pointloadReactions(rr,12)=(P*a^2*b/L^2); % CHANGED TRIAL -----
****-----
end
if a==b;
pointloadReactions(rr,2)=-P/2;
pointloadReactions(rr,8)=-P/2;
pointloadReactions(rr,6)=-P*L/8; % CHANGED TRIAL -----
pointloadReactions(rr,12)=P*L/8; % CHANGED TRIAL -----
end
end
% Xz ... member X and load in z direction
if PointloadDirection(i,1)==2; %condition for the member direction and load
direction,
if a>b || a<b
pointloadReactions(rr,3)=-(P*b^2/L^3)*(3*a+b);
pointloadReactions(rr,9)=-(P*a^2/L^3)*(a+3*b);
pointloadReactions(rr,5)=(P*a*b^2/L^2);
pointloadReactions(rr,11)=-(P*a^2*b/L^2);
end
if a==b;
pointloadReactions(rr,3)=-P/2;
pointloadReactions(rr,9)=-P/2;
pointloadReactions(rr,5)=P*L/8;
pointloadReactions(rr,11)=-P*L/8;

```

```

    end
end
% Yx ... member Y and load in x direction
if PointloadDirection(i,1)==3; %condition for the member direction and load
direction,
    if a>b || a<b
        pointloadReactions(rr,2)=- (P*b^2/L^3)*(3*a+b); % changed , which was 1
        pointloadReactions(rr,8)=- (P*a^2/L^3)*(a+3*b); % changed , which was 7
        pointloadReactions(rr,6)=(P*a*b^2/L^2); % CHANGED TRIAL -----
****-----
        pointloadReactions(rr,12)=- (P*a^2*b/L^2); % CHANGED TRIAL -----
****-----
    end
    if a==b;
        pointloadReactions(rr,2)=-P/2;
        pointloadReactions(rr,8)=-P/2;
        pointloadReactions(rr,6)=P*L/8; % CHANGED TRIAL -----
        pointloadReactions(rr,12)=-P*L/8; % CHANGED TRIAL -----
    end
end
end
% Yz ... member Y and load in z direction
if PointloadDirection(i,1)==4; %condition for the member direction and load
direction,
    if a>b || a<b
        pointloadReactions(rr,3)=- (P*b^2/L^3)*(3*a+b);
        pointloadReactions(rr,9)=- (P*a^2/L^3)*(a+3*b);
        pointloadReactions(rr,4)=- (P*a*b^2/L^2);
        pointloadReactions(rr,10)=(P*a^2*b/L^2);
    end
    if a==b;
        pointloadReactions(rr,3)=-P/2;
        pointloadReactions(rr,9)=-P/2;
        pointloadReactions(rr,4)=-P*L/8;
        pointloadReactions(rr,10)=P*L/8;
    end
end
end
% Zx ... member Z and load in x direction
if PointloadDirection(i,1)==5; %condition for the member direction and load
direction,
    if a>b || a<b
        pointloadReactions(rr,2)=- (P*b^2/L^3)*(3*a+b); % changed , which was 2
        pointloadReactions(rr,8)=- (P*a^2/L^3)*(a+3*b); % changed , which was 8
        pointloadReactions(rr,5)=- (P*a*b^2/L^2);
        pointloadReactions(rr,11)=(P*a^2*b/L^2);
    end
    if a==b;
        pointloadReactions(rr,2)=-P/2;
        pointloadReactions(rr,8)=-P/2;
        pointloadReactions(rr,5)=-P*L/8;
        pointloadReactions(rr,11)=P*L/8;
    end
end
end
% Zy ... member Z and load in y direction
if PointloadDirection(i,1)==6; %condition for the member direction and load
direction,

```



```

    if a>b || a<b
        pointloadReactions(rr,2)=-(P*b^2/L^3)*(3*a+b);
        pointloadReactions(rr,8)=-(P*a^2/L^3)*(a+3*b);
        pointloadReactions(rr,4)=(P*a*b^2/L^2);
        pointloadReactions(rr,10)=-(P*a^2*b/L^2);
    end
    if a==b;
        pointloadReactions(rr,2)=-P/2;
        pointloadReactions(rr,8)=-P/2;
        pointloadReactions(rr,4)=P*L/8; % 4
        pointloadReactions(rr,10)=-P*L/8; % 10
    end
end
end
ww=size(elementDof); % to find the ww1 which is the total number of Dof
ww1=ww(1)*ww(2); %ww is the size of all Dof of all elements,(all elemens=6*number of
elements)
indxx=reshape(elementDof,[ww1 1]);
mm=reshape(distributedloadReactions,[ww1 1]);
nn=reshape(pointloadReactions,[ww1 1]);
indxx(:,2)=mm;
indxx(:,3)=nn;
for i=12:-1:1 % -1:1 is implicit preallocation from reverse indexing, right, it was
not there in Frame3D File :)
    Reactions(i,1)=sum(indxx(indxx(:,1)==i,2));
    Reactions(i,2)=sum(indxx(indxx(:,1)==i,3));
end
Reactions11=pointloadReactions+distributedloadReactions;
Reaction22=Reactions11';
DofEachElement = zeros(ww(2),1,numberElements);
Di=DofEachElement;
internalForces=Di;
for i=1:numberElements
    ElementReactions=Reaction22(:,i);
    DofEachElement(:,i)=elementDof(i,:);
    Di(:,i)=displacements(DofEachElement(:,i),1);
    internalForces(:,i)=ki(:,i)*T(:,i)*Di(:,i)+ElementReactions;
end
ww=size(elementDof); % to find the ww1 which is the total number of Dof
ww1=ww(1)*ww(2);
memberforces=reshape(internalForces,[ww1 1]);
AxialNed=memberforces(1:6:end);
Ned1=abs(AxialNed(1:2:end));
Ned1(:,2)=abs(AxialNed(2:2:end));
Ned0=(max(Ned1'))';
ShearVed=memberforces(2:6:end);
Ved1=abs(ShearVed(1:2:end));
Ved1(:,2)=abs(ShearVed(2:2:end));
Ved0=(max(Ved1'))';
FZ=memberforces(3:6:end);
FZed1=abs(FZ(1:2:end));
FZed1(:,2)=abs(FZ(2:2:end));
FZed0=(max(FZed1'))';
MX=memberforces(4:6:end);
MXed1=abs(MX(1:2:end));

```

```

MXed1(:,2)=abs(MX(2:2:end));
MXedx0=(max(MXed1'))';
MY=memberforces(5:6:end);
MYd1=abs(MY(1:2:end));
MYed1(:,2)=abs(MY(2:2:end));
MYed0=(max(MYed1'))';
MZ=memberforces(6:6:end);
MZed1=abs(MZ(1:2:end));
MZed1(:,2)=abs(MZ(2:2:end));
MZed0=(max(MZed1'))';
shear=ShearVed(1:2:end);
shear(:,2)=ShearVed(2:2:end);
moment=MZ(1:2:end);
moment(:,2)=MZ(2:2:end);
    NewParams.Forces=memberforces;
    NewParams.Ned0 = Ned0;
    NewParams.Ved0 = Ved0;
    NewParams.FZed0 = FZed0;
    NewParams.MXedx0 = MXedx0;
    NewParams.MYed0 = MYed0;
    NewParams.MZed0 = MZed0;
    NewParams.shear = shear;
    NewParams.moment = moment;
    NewParams.ggg=ggg;
    NewParams.kkk=kkk;
end

```

% Design procedure

```

function [MRx, FBRPartY, FBRPartZ,LTBRC1,LTBRC2,CRc,SRx,Defl] = parametersDesign(Ux,
i,sc,id,DesignParams)
elementNodes = DesignParams.elementNodes;
ggg = DesignParams.ggg; Ned = DesignParams.Ned;
gammaM0 = DesignParams.gammaM0; gammaM1 = DesignParams.gammaM1;
steelGrade = DesignParams.steelGrade;
epsilon = DesignParams.epsilon;
ita = DesignParams.ita; Ved = DesignParams.Ved;
MYed = DesignParams.MYed; MZed = DesignParams.MYed;
L = DesignParams.L; C1 = DesignParams.C1;
E = DesignParams.E; G = DesignParams.G;
lambdaDashLT0 = DesignParams.lambdaDashLT0;
moment1 = DesignParams.moment1;
moment3 = DesignParams.moment3;
moment5 = DesignParams.moment5;
beta = DesignParams.beta;
LiveL = DesignParams.LiveL;
tw=Ux(i,4); h=Ux(i,2);
tf=Ux(i,5); b=Ux(i,3);
cwtw=Ux(i,8); r=Ux(i,6);
cftf=Ux(i,9); d=Ux(i,7);
Iw=Ux(i,25); It=Ux(i,26);
Iy=Ux(i,15); iy=Ux(i,17);
Iz=Ux(i,16); iz=Ux(i,18);
Wely=Ux(i,19); Welz=Ux(i,20);
Wply=Ux(i,21); Wplz=Ux(i,22);

```

```

A=Ux(i,27);
%-----yeildstrength reduction fy-----
tff=tf*1000; % to get it back to normal unit mm
if steelGrade==275
    fy(tff(:,1)<=16,1)=275;
    fy(tff(:,1)>16,1)=265;
    fy(tff(:,1)>40,1)=255;
    fy(tff(:,1)>63,1)=245;
else % SteelGrade==355
    fy(tff(:,1)<=16,2)=355;
    fy(tff(:,1)>16,2)=345;
    fy(tff(:,1)>40,2)=335;
    fy(tff(:,1)>63,2)=325;
fy=fy(:,2);
end
fys=fy*1000; % to convert n/mm2 to kN/m2 , because all the units are meter and kN.
%-----section classification-----
alfa=0.5*(1+(Ned/(fys*d*tw))); % alfa equations
if alfa >1 || alfa <=-1
    alfa=1;
end
psi=((2*Ned)/(A*fys))-1; % psi equations
%-----
sectionclassification=3;
if cftf<=(9*epsilon)
    if alfa>0.5 && cwtw<=((396*epsilon)/(13*alfa-1))
        sectionclassification = 1;
    elseif alfa<=0.5 && cwtw<=((36*epsilon)/(alfa))
        sectionclassification = 1;
    end
elseif cftf<=(10*epsilon) && cftf>(9*epsilon)
    if alfa>0.5 && cwtw<=((456*epsilon)/(13*alfa-1))
        sectionclassification = 2;
    elseif alfa<=0.5 && cfw<=((41.5*epsilon)/(alfa))
        sectionclassification = 2;
    end
elseif cftf<=(14*epsilon) && cftf>(10*epsilon)
    if psi>-1 && cwtw<=((42*epsilon)/(0.67+0.33*psi))
        sectionclassification = 3;
    elseif psi<=-1 && cwtw<=((62*epsilon)*(1-psi)*(sqrt(-psi)))
        sectionclassification = 3;
    end
end
%-----cross-sectional resistance-----
% $$$ compression resistance
NoRd=(A*fys)/(gammaM0);
CRc=Ned/NoRd; %<=1 CRc is a breviation for compression resistance for both beam and
column

Av=A-2*b*tf+(tw+2*r)*tf;
hw=h-2*tf;
if Av<ita*hw*tw
    Av=ita*hw*tw;
end
VoRd=(Av*(fys/sqrt(3)))/gammaM0;

```

```

SRx=Ved/VcRd; %<=1 SR is a breviation for Shear Ressistance for beam and column
%-----
lambda1=93.9*epsilon;
Mcr=C1*(pi^2*E*Iz/L^2)*sqrt((Iw/Iz)+((L^2*G*It)/(pi^2*E*Iz)));
% flexural buckling about y-y
lambdaDashPartYY=(L/iy)*(1/lambda1); %Lcr needs to be found in case of there is a
support in mid of Lcrz or Lcry-----+++++
if h/b>1.2
    if tf<=0.04
        bucklingcurvePartYY='a';
        bucklingcurvePartZZ='b';
    end
    if tf>0.04
        bucklingcurvePartYY='b';
        bucklingcurvePartZZ='c';
    end
elseif h/b<=1.2
    if tf<=0.1
        bucklingcurvePartYY='b';
        bucklingcurvePartZZ='c';
    end
    if tf>0.1
        bucklingcurvePartYY='d'; % There is no c, in 2D structures, it is for
buckling about z-z
        bucklingcurvePartZZ='d'; % There is no c, in 2D structures, it is for
buckling about z-z
    end
end
% alpha -- imperfection factor for beam about Y-Y
if bucklingcurvePartYY=='a'
    alphaPartYY=0.21;
elseif bucklingcurvePartYY=='b'
    alphaPartYY=0.34;
else
    alphaPartYY=0.76;
end
if bucklingcurvePartZZ=='b'
    alphaPartZZ=0.34;
end
if bucklingcurvePartZZ=='c'
    alphaPartZZ=0.49;
end
if bucklingcurvePartZZ=='d'
    alphaPartZZ=0.76;
end
phiPartYY=0.5*(1+alphaPartYY*(lambdaDashPartYY-0.2)+lambdaDashPartYY^2);
chiPartYY=1/(phiPartYY+sqrt(phiPartYY^2-lambdaDashPartYY^2));
NbyRd=chiPartYY*A*fys/gammaM1;
FBRPartY=(Ned/NbyRd); % Flexural buckling Resistance about y-y axis
lambdaDashPartZZ=(L/iz)*(1/lambda1); %Lcr needs to be found-----
% alpha -- imperfection factor for beam about Z-Z
phiPartZZ=0.5*(1+alphaPartZZ*(lambdaDashPartZZ-0.2)+lambdaDashPartZZ^2);
chiPartZZ=1/(phiPartZZ+sqrt(phiPartZZ^2-lambdaDashPartZZ^2));
NbzRd=chiPartZZ*A*fys/gammaM1;
FBRPartZ=(Ned/NbzRd); % Flexural buckling Resistance about y-y axis

```



```

end
if Cmy<0.4
    Cmy=0.4;
end
sigmaVMax=L/360; %360
sigmaV=((LiveL*L^4)/(384*E*Iy));
VD=sigmaV/sigmaVMax;
end
Cmz=Cmy; % Cmz = Cmy =CmLT. for columns and beams According to EC3 page 80 ✓
.....
%-----
% $$$ Bending Moment
RO=((2*Ved)/(VcRd))-1)^2;
fyr=(1-RO)*fys;
if sectionslassification==1 || sectionslassification==2
    Kyy = Cmy*(1 + ((lambdaDashPartYY-0.2)*(Ned/NbyRd)));
    Kzz=Cmz*(1+((2*lambdaDashPartZZ)-0.6)*(Ned/NbzRd));
    if Kyy > Cmy*(1+(0.8*(Ned/NbyRd)))
        Kyy=Cmy*(1+(0.8*(Ned/NbyRd)));
    end
    if Kzz > Cmz*(1+(1.4*(Ned/NbzRd)))
        Kzz=Cmz*(1+(1.4*(Ned/NbzRd)));
    end
    Kyz=0.6*Kzz;
    Kzy=0.6*Kyy;
    Wy=Wply;
    McRd=(Wy*fys/gammaM0);
    if Ved>=VcRd/2
        McRd=(Wy*fyr/gammaM0);
    end
    McbzRd=(Wplz*fys/gammaM1);

else % when sectionslassification==3
    Kyy = Cmy*(1 + (0.6*lambdaDashPartYY*(Ned/NbyRd)));
    Kzz=Cmz*(1+(0.6*lambdaDashPartZZ*(Ned/NbzRd)));

    if Kyy > (1+(0.6*(Ned/NbyRd)))
        Kyy = (1+(0.6*(Ned/NbyRd)));
    end
    if Kzz > (1+(0.6*(Ned/NbzRd)))
        Kzz = (1+(0.6*(Ned/NbzRd)));
    end
    Kyz=Kzz;
    Kzy=0.8*Kyy;
    Wy=Wely;
    McRd=(Wy*fys/gammaM0);
    if Ved>=VcRd/2
        McRd=(Wy*fyr/gammaM0);
    end
    McbzRd=(Welz*fys/gammaM1);
end
lambdaDashLT=sqrt(Wy*fys/Mcr);
phiLT=0.5*(1+alphaLT*(lambdaDashLT-lambdaDashLT0)+beta*lambdaDashLT^2);
chiLT=1/(phiLT+sqrt(phiLT^2-beta*lambdaDashLT^2));
if chiLT>1
    chiLT=1;

```

```

end
% MbRd - Lateral torsional buckling
MbRd=chiLT*Wy*fys/gammaM1;
MRx=MZed/McRd;
LTBRC1=(Ned/NbyRd)+(Kyy*(MYed/MbRd))+(Kyz*(MZed/McbzRd));
LTBRC2=(Ned/NbzRd)+(Kzy*(MYed/MbRd))+(Kzz*(MZed/McbzRd));

if sc == 2 %column
    Defl = HD;
else
    Defl = VD;
end
end
end

```

% Design constraints

```

function [c,ceq] = constraints(x,params)
Load1 = params.Load1;
L0 = params.L0;
E = params.E;    G = params.G;
elementNodes=params.elementNodes;
UCs = params.UCs;    UBs = params.UBs;
numberElements = params.numberElements;
for id=numberElements:-1:1
    SectionType=elementNodes(:,3);
    sc = SectionType(id,1);
    row=x(id);
    if sc==2 % if it's column
        EIy(id) = E* UCs(row,15); % save values to rememeber chosen row
        EIz(id) = E* UCs(row,16); % save values to rememeber chosen row
        EA(id) = E* UCs(row,27);
        GJ(id) = G* UCs(row,26); % "J is "It" in the UCs" means J >> Torsional ✓
    constant for each element
    else
        EIy(id) = E* UBs(row,15);
        EIz(id) = E* UBs(row,16); % save values to rememeber chosen row
        EA(id) = E* UBs(row,27);
        GJ(id) = G* UBs(row,26); % "J is "It" in the UBs" means J >> Torsional ✓
    constant for each element
    end
end
forces = computeForce(EA,EIy,EIz,GJ,params);
Ned0 = forces.Ned0; Ved0 = forces.Ved0;
MYed0 = forces.MYed0;    MZed0 = forces.MZed0;
shear = forces.shear;    moment = forces.moment;
params.ggg = forces.ggg;
c=zeros(numberElements*7,1);
for i=1:numberElements
    choice = x(i);
    SectionType=elementNodes(:,3);
    sc = SectionType (i,1);
    Ned=Ned0(i,1);    Ved=Ved0(i,1);

```



```

MYed=MYed0(i,1);    MZed=MZed0(i,1);
L=L0(i,1);
shear1=shear(i,1);
moment1=moment(i,1); % moment1 is M1 at the left end
moment3=shear1*0.5*L+moment1+Load1(i,2)*0.5*L*0.25*L; % Moment at the mid of the member
moment5=moment(i,2); % moment2 is M2 at the end right
%send values to design
params.MYed=MYed;    params.Ned=Ned;
params.MZed=MZed;    params.Ved=Ved;
params.L=L;
params.moment1=moment1;
params.moment3=moment3;
params.moment5=moment5;
if sc==2
    UXs=UCs;
else
    UXs=UBs;
end
[MR, FBRelY, FBRelZ,LTBRE1,LTBRE2,CR,SR,ED] = parametersDesign(UXs,choice,sc,i,
params);
c(1+(i-1)*8) = MR -1; % constraints are in form C < 0, if C < 1, then we make C
- 1 < 1 - 1
c(2+(i-1)*8) = FBRelY -1;
c(3+(i-1)*8) = FBRelZ -1;
c(4+(i-1)*8) = LTBRE1 -1;
c(5+(i-1)*8) = LTBRE2 -1;
c(6+(i-1)*8) = CR -1;
c(7+(i-1)*8) = SR -1;
c(8+(i-1)*8) = ED -1;
ceq = [];
end
end

% Objective function+++++
function OptimumDesign = myCost(UCs,UBs,x,params,Type)
w=0;
e_st=0; e_pa=0;
e_tt=0; e_ee=0;
e_cs=0; e_cp=0;
e_ct=0; e_ce=0;
Cm=0; Ct=0;
Cf=0; Cc=0;
Ce=0;
EE=0; EC=0;
Cost=0;
numberElements = params.numberElements;
elementNodes = params.elementNodes;
SectionType=elementNodes(:,3);
l1 = params.l1;
p = 7.850; % [tonne/m3]
persistent pen_m
if isempty(pen_m)
pen_m1=max(UCs(:,1))-min(UCs(:,1));

```



```

pen_m2=max(UBs(:,1))-min(UBs(:,1));
%create max constraint for every element
pen_m=pen_m1*(SectionType==2)+pen_m2*(SectionType==1);
%expand constraint factor for 8 parameters
pen_m= repmat(pen_m,[1 8])';
pen_m=pen_m(:);
end
for i=1:numberElements
    sc = SectionType (i,1);
    choice = x(i);
    if sc==2
        A=UCs(choice,27); % read
        a=UCs(choice,13); % read
        mass=UCs(choice,1); % read    % just added.....
    else
        A=UBs(choice,27); % read
        a=UBs(choice,13); % read
        mass=UBs(choice,1);    % just added.....
    end
    L = ll(i);
    % weight value
    w = w + (10*mass*L); % total mass accumulation
    % energy value
    e_st = e_st + (20.1*1000*mass*L); % material
    e_pa = e_pa + 21*a*L; % painting (fire & Corrosion)
    e_tt = e_tt + (1.66*101.85*mass*L); % 0.97*100km*1.05factor  Transportation
    e_ee = e_ee + (958.672*mass*L); % (0.17*163.6*0.54/0.84)*38.3*0.0014*1000
    EE= e_st+e_pa+e_tt+e_ee;
    % carbon
    e_cs = e_cs + 1.6275*1000*mass*L; %1.55*1.05*w;
    e_cp = e_cp + 0.87*a*L; %Carbon for painting
    e_ct = e_ct + 1.66*7.5075*mass*L; % 0.0715*100*1.05 Transportation carbon
    e_ce = e_ce + 72.9397*mass*L; % (0.17*163.6*0.54/0.84)*2.914*0.0014*1000 Erection
    EC=e_cs+e_cp+e_ct+e_ce; %UNUSED? for future?
    % cost
    Cm = Cm+900*mass*L; % S275>1550 AND S355>1638
    Ct = Ct+50*mass*L; % in Economic page ...
    Ce = Ce+243*mass*L; % in spon's page 326
    Cf = Cf + 16*a*L; % in spon's page 172-38
    Cc = Cc + 8.9*a*L; % in spon's page 171
    Cost=Cm+Ct+Ce+Cf+Cc; %UNUSED? for future?
end% for i
penalty=0;
if Type>99
    [c,~] = constraints(x,params);
    flag = any(c >= 1e-4 ); % if there is violation of constraints, with tolerance
    penalty=flag*sum(pen_m.*(c >= 1e-4).*((c+1).^2));
end
switch rem(Type,10) % switch last digit
    case 1
        OptimumDesign = w*(1+penalty);
    case 2
        OptimumDesign = EE*(1+penalty);
    case 3
        OptimumDesign = EC*(1+penalty);

```

```

    case 4
        OptimumDesign = [w EE].*(1+penalty*[1 1]);
    otherwise
        OptimumDesign = 1;
end
if rem(Type,100)>9
    sep=['\n' repmat('--',[1 35]) '\n'];
    fprintf(['\n' sep]);
    disp(['Candidate: ' num2str(x)])
    fprintf('\nWeight: %2.2f\nEmbodied energy: %2.3fx10^3\nEmbodied carbon: %2.3x10^3\nCost: %4.1f',w,EE/1e3,EC/1e3,Cost);
    fprintf(sep);
end
end
end

```

% Calling optimisation methods

```

%% Initialize variables
clear functions %#ok
close all
rng('default')
% load constants from excel:
% LOADSAVE
load('Excel_variables.mat')
load('UCs.mat')
load('UBs.mat')
% load('NumericalValues.mat');
tff=0; L=0; ggg=0;
Ned=0; MYed=0; MZed=0; Ved=0;
moment1=0; moment3=0; moment5=0;
%displacement values
indexes=1:GDof;
%can be generalized for N dimensions but whatever... we don't build tesseracts
r=[ [4:6:GDof] [5:6:GDof] [6:6:GDof]];
%removing rotations
indexes(r)=[];
%precompute DoF we are interested in
activeDof=(1:GDof)';
logUA = ~(ismember(activeDof,prescribedDof));
activeDof = activeDof(logUA);
%precompute yieldstrength reduction fy - around 22% profiling time reduction
tfC=UCs(:,5)*1e3;
tfB=UBs(:,5)*1e3;
fyC_275=zeros(46,1);
fyC_355=zeros(46,1);
fyB_275=zeros(107,1);
fyB_355=zeros(107,1);
if steelGrade==275
    fyC_275(tfC(:,1)<=16,1)=275;
    fyC_275(tfC(:,1)>16,1)=265;
    fyC_275(tfC(:,1)>40,1)=255;
    fyC_275(tfC(:,1)>63,1)=245;
    fyB_275(tfB(:,1)<=16,1)=275;

```

```

fyB_275(tfB(:,1)>16,1)=265;
fyB_275(tfB(:,1)>40,1)=255;
fyB_275(tfB(:,1)>63,1)=245;
else % SteelGrade==355
fyC_355(tfC(:,1)<=16,2)=355;
fyC_355(tfC(:,1)>16,2)=345;
fyC_355(tfC(:,1)>40,2)=335;
fyC_355(tfC(:,1)>63,2)=325;
fyC_355=fyC_355(:,2);
fyB_355(tfB(:,1)<=16,2)=355;
fyB_355(tfB(:,1)>16,2)=345;
fyB_355(tfB(:,1)>40,2)=335;
fyB_355(tfB(:,1)>63,2)=325;
fyB_355=fyB_355(:,2);
end
% constants for visualization
Ncomb=9; %combinations of all objectives and algorithms called - not perfect as of
today 27.11.2021
Nobjectives=2;
candidate=cell(Ncomb,1); Plots.figs=cell(Ncomb,1); Plots.names=cell(Ncomb,1);
times=zeros(Ncomb,1); bestTimes=zeros(Ncomb,2,Nobjectives); costHistory=cell(Ncomb,
2); figNum=0;
% ^ 1st-time, 2nd-iter ^
% data to send into functions
params = struct('activeDof',activeDof,'beta',beta , 'C1',C1 , 'DisloadDirection',
DisloadDirection, 'E',E,...
'elementDof',elementDof,'epsilon',epsilon,'forceVector',forceVector,...
'fyC_275',fyC_275*1e3,'fyB_275',fyB_275*1e3,... %1e3 to convert n/mm2 to kN/m2
'fyC_355',fyC_355*1e3,'fyB_355',fyB_355*1e3,'G',G,...
'elementNodes',elementNodes,'ggg',ggg,'gammaM0',gammaM0,'gammaM1',gammaM1,...
'indexes',indexes, 'ita',ita, 'KgElement',0,'Ved',Ved, 'MYed',MYed,...
'MZed',MZed, 'll',ll,'LiveL',0, 'Loadl',Loadl, 'loadpoints',loadpoints,'L',L,...
'lambdaDashLT0',lambdaDashLT0, 'moment1',moment1, 'moment3',moment3,...
'moment5',moment5,'Ned',Ned,'phiColumnYY',0, 'phiColumnZZ',0, 'phiLT',0,...
'psai',0, 'psi',0, 'ki',0,'memberlength',memberlength,'nodeCoordinates',
nodeCoordinates,...
'numberElements',numberElements, 'stiffness',0, 'T',0, 'Tt',0,'loadmembers',
loadmembers,...
'loadjoints',0,'GDof',GDof,'steelGrade',steelGrade,...
'V',V,'xx',xx,'yy',yy,'zz',zz,'L0',L0,'UCs',UCs,'UBs',UBs);
afterParams = struct('continuity',0, 'grouping',1, 'is2D',0, 'multiObj',0,
'Nobjectives',Nobjectives, 'type',[]);
%% Optimisation part
A=[1 2 3 4 9 10 11 12];
B=[5 6 7 8];
C=[13 14 15 16 17 18];
D=[19 20 21 22 23 24 25 26];
Order={A,B,C,D};
BuildGroupFunction(Order)
nvars=length(Order); % number of groups
intcon = 1:nvars;
ub=zeros(nvars,1);
for id=1:nvars %read bounds from corresponding groups
ub(id) = 46*(SectionType(Order{id}(1)) == 2) + 107 * (SectionType(Order{id}(1)) ==
1);

```

```

end
lb = ones(nvars,1);
% Single Objective definitions
singleoptions=gaoptimset('Generations',150,'EliteCount',100,'SelectionFcn',↵
@selectiontournament, ...
    'PopulationSize',150,'CrossoverFraction',0.7,'FitnessScalingFcn', @fitscalingrank,↵
...
    'PlotFcns', {@gplotbestf, @gplotbestindiv, @gplotexpectation});
psoptions = optimoptions('particleswarm','SwarmSize',100,'HybridFcn',↵
@fmincon,'Display','final','PlotFcn',{pswplotbestf});
hsoptions = struct('population', 150, 'bw', 5, 'HMCR', 0.6,'PAR', 0.5, 'MaxItr', 150);
% Multi Objective definitions
% {@gplotbestf, @gplotbestindiv, @gplotexpectation,@gplotdistance,↵
@gplotgenealogy, @gplotselection, @gplotscorediversity, @gplotrange});
% !!! renamed gplotpareto function so that it works as we want !!!
multioptions = optimoptions(@gamultiobj,'PlotFcn',@gplotpareto2,'Generations',↵
100,'TolCon',1e-7,'SelectionFcn',@selectiontournament, ...
    'CrossoverFcn',@crossoverheuristic,'PopulationSize',100,'HybridFcn',↵
@fgoalattain,'Display','final');
mopsoptions = struct('c', [0.1,0.2], 'iw', [0.5 0.001], 'max_iter', 100, 'swarm_size',↵
100, 'grid_size', 70,...
    'alpha', 0.1, 'beta', 2, 'gamma', 2, 'mu', 0.1);
mohsoptions = struct('population', 100, 'bw', 0.3, 'HMCR', 0.3,'PAR', 0.3, 'MaxItr',↵
100);
% User interaction
clc
Choice = input(['Which parameters optimize to?\n 1 - weight\n 2 - energy\n 3 -↵
carbon\n 4 - multi (weight+energy)\n'...
    'H - HSA, E - mod HSA, G - Ga, P - PSO\nExample: 1G - GA weight, 4P - PSO mixed,↵
2EGHP-runs every model on energy cost...\n>> '], 's');
algoT = {'PSO','GA','HSA','EGHS'};
goalT = {'weight','energy','carbon','Multi-obj (weight+energy)'};
goalO = 'single';
algoID=find( ismember('PGHE',Choice) );
%goalID=find( ismember('123',Choice) );
goalID=Choice(1)-48;
nAlgs=length(algoID);
nGoal=length(goalID);
for iGoal=goalID % for future
    for iAlg=algoID
        if iGoal==4, goalO = 'multi'; end
        fprintf('\nStarting %s objective %s - %s...\n',goalO,algoT{iAlg},goalT{iGoal})
        tic
        ConF=@(x) constraints(group(round(x)),params);
        if iGoal==4
            CostMOPSO=@(row)myCost(UCs,UBs,group(round(row)),params,103);
            CostGAMulti=@(row)myCost(UCs,UBs,group(round(row)),params,103);
            CostMOHSA=CostMOPSO;
            afterParams.multiObj=1;
            switch iAlg
                case 4
                    error('MOHSA integer not implemented yet')
                case 3
                    [x_best,history] = mohsa(CostMOHSA,lb,ub,mohsoptions);
                case 2

```

```

        [x_best,history] = gamultiobj2(CostGAMulti,nvars,[],[],[],[],lb,ub,
ub,ConF,multioptions);

        % Find unique from population
        x_gaMul = unique(round(x_best),'rows');

        x_best=x_gaMul(1,:); %group done in afterOptim
    case 1
        [best,history,swarm] = mopso(CostMOPSO,lb',ub',ConF,mopsotions);
        x_best = round(best.x);
    end
else
    CostGA=@(row)myCost(UCs,UBs,group(row),params,iGoal); %last param either 1
or 2
    CostPSO=@(row)myCost(UCs,UBs,group(round(row)),params,100+iGoal);
    CostHSA=CostPSO;
    afterParams.multiObj=0;
    switch iAlg %call algorithm
        case 4
            x_best = hsa_integer(CostHSA,lb,ub,hsotions);
        case 3
            x_best = hsa1(CostHSA,lb,ub,hsotions);
        case 2
            x_best = ga(CostGA,nvars,[],[],[],[],lb,ub,ConF,intcon,
singleoptions);
            le=length(singleoptions.PlotFcns); % check which subplot we show
            subplot(le,1,1)
            %cleanup strange graph-cost phenomena, probably wrong
            %plotfcn
            L=han.Children(2);
            [val,pos]=max(L.YData);
            L.YData(1:pos)=val;
            L=han.Children(1);
            L.YData(1:pos)=L.YData(1:pos)+val;
        case 1
            x_best = round(particleswarm(CostPSO,nvars,lb,ub,psotions));
    end%switch iAlg
end%if iGoal==3

    afterParams.type=sprintf( '%s %s', algoT{iAlg}, goalT{iGoal} );
    afterOptim(afterParams)

end%for iAlg
end%for iGoal
%% Visualization part
Range = 1:figNum;
visualize(figNum,Plots,cellfun(group,candidate(Range),'UniformOutput',0),params,
afterParams.continuity)

%% Comparison
comparison(goalID,costHistory,bestTimes,Plots,Range)
%% Helpful functions
%moved to separate files

```