JETIR.ORG ISSN: 2349-5162 | ESTD Year : 2014 | Monthly Issue
JOURNAL OF EMERGING TECHNOLOGIES AND INNOVATIVE RESEARCH (JETIR)
An International Scholarly Open Access, Peer-reviewed, Refereed Journal

Paper Optimization in the Design of Steel-Concrete Bridges-COMPOPT

[1]Chikumbuso Lungu, [2]Dr. Michael N. Mulenga

[1]Research Scholar, [2]Research Supervisor
[1]Department of Civil and Environmental Engineering,
[1]University of Zambia, Lusaka, Zambia

Abstract: A study to demonstrate the positive outcome of incorporating optimization techniques in the design procedure for designing steel-concrete composite bridges was conducted. Optimization algorithms have the capability of finding optimal or near optimal solutions in complex problems. In order to accomplish this task, a software known as COMPOPT, was developed in C# programming language to optimize composite bridge designs based on a classical method of optimization known as 'direct search'. The software output was then compared to structural designs that would be obtained through the typical 'traditional' design procedure. Traditional design procedure is a term in this study that alludes to normal procedure of design that practicing structural engineers follow in the design process, including use of design aids such as structural analysis and design software like Prokon and Robot. In summary, the output structural designs from the developed C# application were compared to typical structural designs from selected practicing structural engineers in Zambia. The results showed that the design output from COMPOPT performed better than the typical designs on several key parameters.

Key words: **Optimization Techniques; Direct search method; Steel-Concrete Composite Bridges; Bridge Optimization;**

1. Introduction

Optimization is the process of finding the best or most effective use of a situation 'optimal solution' with a given a set of limitation. The mathematical tools and concepts used to arrive at this 'optimal solution' are termed as optimization techniques. This has been a growing area of interest to civil engineers because of its applicability. (Himani & Dr. Monisha, 2015)

Optimization can be applied to various types of structures in civil engineering design. Pedro, et al (2017) noted that structural optimization is a very relevant field and has been a growing focus on research. Initial emphasis had been given to truss structures and some important advances were carried out. However, it is essential to note that the main focus of these studies was the implementation and development of different optimization procedures applied to academic examples. (Pedro, et al., 2017)

There is a general sense that this field hasn't received adequate attention. The slow progress in this field is largely attributed to tedious procedures involved in the calculation. But the emergence of computers has brought some significant interest in application of these techniques to structural engineering problems. (Numan, 2012)

However, this research focused on the application of optimization techniques to Structural Design of Concrete-Steel Composite Bridges. This type of bridge is mainly composed of a reinforced concrete decking supported on steel girders. Standard rolled section steel beams are rarely used as main girders, instead plate girders are usually used and have proven to be more economical since the plate size and thickness chosen for efficiency.

In the design of multi-girder composite bridges, a number of similarly sized longitudinal plate girders are arranged at uniform spacing across the width of the bridge, as shown in the typical cross section in Figure 1.1. The deck slab spans transversely between the longitudinal girders and cantilevers transversely outside the outer girders. The girders are braced together at supports and at some intermediate positions. Composite action between the reinforced concrete deck slab and the longitudinal girders is achieved by means of shear connectors welded on the top flanges of the steel girders. Figures 1.1 and 1,2 show a cross section and longitudinal profile of a composite bridge.

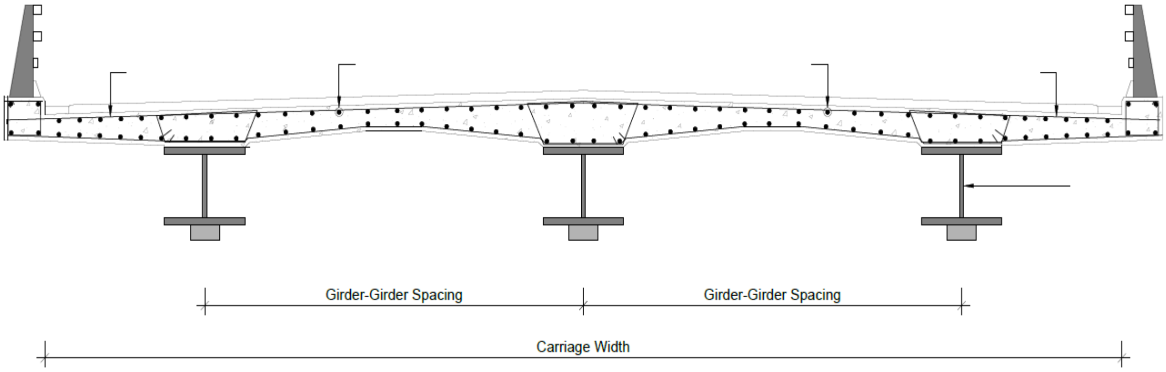

Figure 1.1: Cross Section View

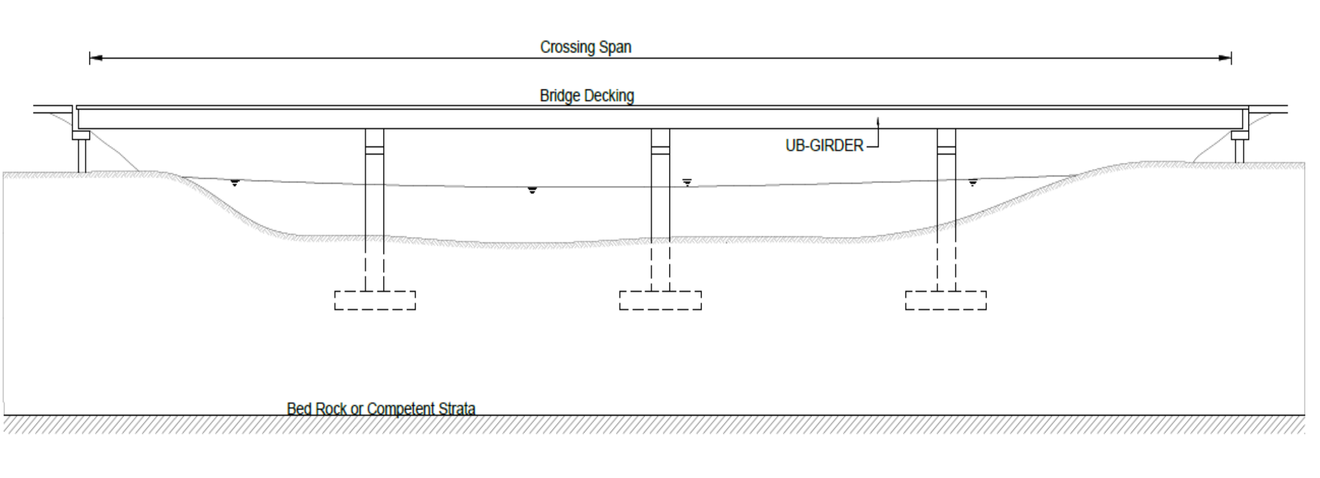

Figure 1.2.: Longitudinal profile of a bridge

Therefore, this study developed an optimization software tool that: (1) optimizes the selection of girders and reinforced concrete decking, (2) Optimize the topology of the bridge spatial parameters such as the spacing in between piers, and (3) compared output design from the software to typical design that would be produced by a practicing structural engineer.

2. Existing Approaches to Structural Optimization of Composite Bridges

There have been significant strides in researching optimization techniques and their application. Most research has been attempting to demonstrate how advanced optimization techniques could be used on various engineering structures. It is important to note that recent technological revolution has enabled the use of optimization techniques. Before the advent of computers, the tedious iterative nature of calculations involved in optimization hindered engineers to adopt optimization techniques. But with the increase in computing power brought in by the electronic era, using these techniques has made possible and easy. (Himani & Dr. Monisha, 2015); (Numan, 2012)

Among the most recent published papers, Fabeane et al (2017), probably defined the optimization problem very well in terms of composite bridges. They presented the problem as follows: Finding $X = \{X_1, X_2, \ldots, X_n\}^T$ that minimizes (or maximizes) the function $f(X)$ under the following constraints:

$$g_j(X) \leq 0, \quad where\ j = 1,2, \ldots, m \quad Equation\ 2.1$$

$$l_j(X) \leq 0, \quad where\ j = 1,2, \ldots, p \quad Equation\ 2.2$$

$$X_i^l \leq X_i \leq X_i^u \ \ldots\ldots\ldots\ldots\ldots\ldots \text{ Equation 2.3}$$

$f(X)$ represents the objective function of the problem which could be, for instance 'least weight of the bridge' or the 'most economical bridge design'. Equations 2.1 and 2.2 represents the constraints on the problem that should be respected so that the obtained solution falls in between these specified bounds. An example of a constraint in bridge design would be deflection. The optimum design needs to have a deflection that falls within the bounds of the permissible deflection allowed by standard codes. Equation 2.3 is called lateral restriction. (Fabeane, et al., 2017)

Now, the current tradition approach to optimization in the structural design industry is limited to the experience and knowledge of the designer. In their paper, Kazakis, et al., noted that what practicing structural engineers deem as an "optimal design" is their choice among rather limited set of design alternatives, dictated by their experience and intuition. (Kazakis, et al., 2017)

Optimization techniques nested in numerical mathematics however a plethora of alternative designs than what the traditional approach presents. Traditional optimization is limited to the designer's knowledge and experience but advanced optimization techniques can analyse over a thousand alternative design in a second (this is made possible with advanced computing power).

2.1. Metaheuristics approach vs Classical approach

There are two (2) main categories of optimization techniques in numerical computation namely: Classical optimization techniques and advanced optimization techniques. Classical optimization techniques use techniques of differential calculus in locating optimum solutions. These methods are relatively old, and could be dated as far back as the Newtonian era. Metaheuristics optimization techniques on the other hand, were developed quite recently. They're based on certain characteristics and behaviour of biological, molecular, swarm and neurobiological systems. (Singiresu, 2019)

Examples of metaheuristic algorithms include: genetic algorithms, simulated annealing, particle swarm optimization, ant colony optimization and neural network methods of optimization. Metaheuristics optimization methods mimic the way nature approaches optimization in these systems. This has allowed researchers in this field to benefit from the billions of years' nature has had experimenting on this problem and finding a workable solution to it. For instance, the Ant colony optimization is based on the behaviour of real ant colonies, which are able to find the optimal path from their nest to a food source.

Natures problems are riddled with a multitude of variables and constraints which make the optimization problem very complex. But nature has found a way around this and this makes the algorithms based on nature very powerful and useful in engineering problems. Metaheuristic algorithms have been popular due to their ability to provide solution to complex engineering problem. A lot research recently conducted in the study of optimization has been mostly exploring the use of application of metaheuristics optimization techniques. However, these algorithms have limitations on applications. Despite been able to handle a lot of variables, convergence to an optimum solution isn't always assured and there is no guarantee that the solution will be found globally. (Chopard & Tomassini, 2018)

In comparison to metaheuristic algorithms, classical optimization techniques have an advantage in that convergence is always assured, given that the problem isn't too complex and objective function is 'continuous and differentiable'. The methodology of this study led to use of this type of algorithm given the nature of the problem. A classical method known as direct search method was adopted.

2.2. Direct methods

Singiresu (2019), posits that a function of one variable $f(x)$ is said to have a local minimum at $x = x^*$ if $f(x^*) \leq f(x^* + h)$ for all positive and negative values of h. Similarly, $f(x)$ is said to have a local maximum at $x = x^*$ if $f(x^*) \geq f(x^* + h)$ for all values of h significantly close to zero. By Singireu's definitions, it follows that, a function $f(x)$ has an absolute global optimum at $x = x^*$ if $f(x^*) \geq f(x)$ or $f(x^*) \leq f(x)$ in the domain over which $f(x)$ is defined. This is the mathematical basis upon which all optimization methods are based on including the direct search methods (Singiresu, 2019)

Direct search methods are applicable to multi-dimensional unconstrained optimization problems. Given that these methods also don't require gradient function, direct search methods could be used in problems where the data set is scattered and there's no easily identifiable function. Example of direct methods include: Random search method and 'univariate and pattern searches' (Chapra & Canale, 2015)

This study endeavoured to use a derivation of this method through a C# program. C# programming language is an object-oriented language. Object oriented programming is a paradigm based on the concept of 'objects' which can contain data and code. With this capability, the direct search method was implemented on the C# platform. (Hanson, 2004)

2.3. Composite Bridge optimization

A number of research has been done on application of optimization techniques applied to composite bridges. Most of it actually attempts to use metaheuristic optimization techniques, comparing the rates of convergence or simply testing them on a design parameter. Another criterion by which the research on this topic can be categorized is by observing which variables are of interest. There are studies investigating the optimization of topological parameters and those optimizing cross section parameters.

Kazakis, et al. (2017), defined structural topology optimization as a procedure of rearranging of structural elements and material into a design domain, thereby eliminating unnecessary material volume. In a composite bridge, this means the spatial arrangement of piers and and positioning of girders. (Kazakis, et al., 2017)

A study conducted by Lythell & Sternberg (2020) on cost optimization of composite bridges noted that most design offices adopt a trial-and-error based approach when designing composite bridges. They hypothesised that implementing this iteration in a computer software with an optimization algorithm would produce more cost-effective preliminary designs. When tested, their results showed that the software was a viable tool for preliminary designs. (Lythell & Stenberg, 2020)

In summary, a recurring theme was observed in the literature reviewed which is: Optimization techniques ultimately improve design results. Additionally, all the papers reviewed posited that introduction of an optimization improved the design results. The only disadvantage is that redundancy on the resulting solution is reduced.

3. Methodology

The study took a two-fold approach to attaining the set objectives of this research. The first step involved developing an optimization tool based on the direct search method, using C# programming language. The second step in the methodology was designed to test the output from the developed software tool, that has incorporated an optimization technique in its design procedure. The output from the software was compared to typical designs that would be obtained using the conventional method, which is, the usual way such a structure would be design by a practicing structural engineer.

3.1. Formulation of the Optimization Algorithm

An algorithm, as defined by Skiena (2020) is a procedural sequence of steps laid out to perform a specific function. An optimization algorithm therefore is a sequence of steps that outlines a procedure which when followed, seeks to find the optimal solution according to the objective function. (Skiena, 2020)

The general optimization algorithm generally takes the form:

Objective function: $C = f(x_1, x_2, \dots\dots, x_n)$ …. Equation 3.1

Constraint function: $g_j(x_1, x_2, \dots\dots, x_n) \le 0, j =$ …. Equation 3.2

$1, \dots, m$ *where* x_i *are the variables*

In this research, the authors goal was to make a computer program that uses an optimization technique in the design process for structural design of bridges. The logic steps that the program followed are outlined in figure 3.1.

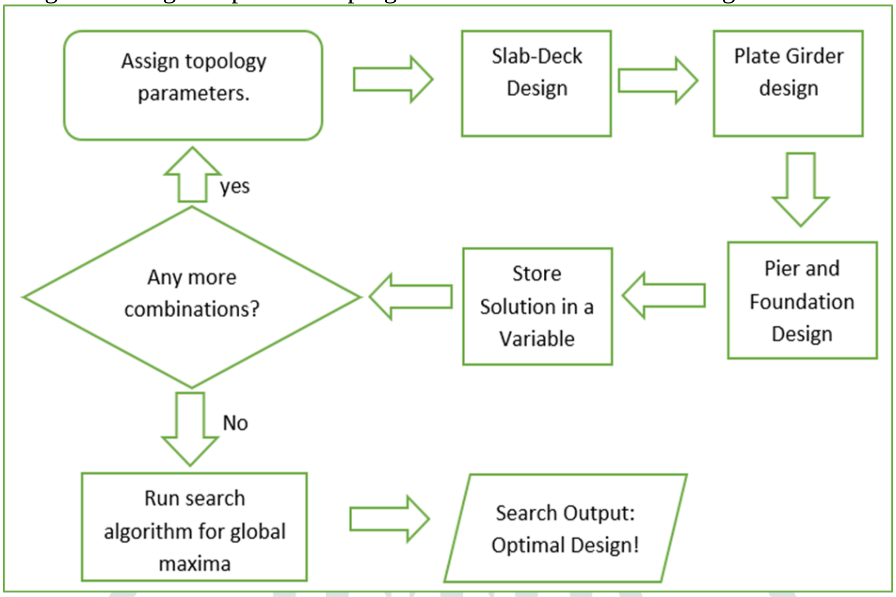

Figure 3.1. Logic Steps

The logic steps in figure 3.1 were implemented in COMPOPT. The algorithm is designed in a loop. With each loop, a solution is stored and after all possible designs (combinations) have been exhausted, the program runs an optimization function through the data set of all feasible designs and finds the most economical design. The objective function in this program, seeks the design that has the least weight.

Least weight criteria was used because of it direct correlation to economy of the design. The relationship is presented as follows:

Weight of bridge design (accumulative weight of individual elements) α Cost of Design………Equation 3.3

COMPOPT has 7 forms in total. 6 of these forms have been devoted to prompt the user to enter certain bridge parameters required for the program to execute the functions. The description of each step is as follows:

Step 1: Start

The 'Static Void Main ()' functions initiates the program, as shown in figure 3.2.

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Threading.Tasks;
using System.Windows.Forms;

namespace bridge_optimizer
{
    0 references
    static class Program
    {
        /// <summary>
        ///  The main entry point for the application.
        /// </summary>
        [STAThread]
        0 references
        static void Main()
        {
            Application.SetHighDpiMode(HighDpiMode.SystemAware);
            Application.EnableVisualStyles();
            Application.SetCompatibleTextRenderingDefault(false);
            Application.Run(new Form1());
        }
    }
}
```

Figure 3.2: Program.cs_Program Initializer

Step 2: Assignment of Parameters

Six (6) forms have been dedicated to prompt the 'User' to enter the topology, material and cross-sectional parameters required for structural design calculations. Topology variables are variables that define the spatial layout of solid structure. Material parameters define material properties for use in the program (Kazakis, et al., 2017).

The topology parameters of a composite bridge coded in the software include:

- Distance between Piers,
- Carriage way width,
- Positioning and orientation of girders,
- Thickness of the decking and
- Head clearance.

Examples of material properties include

- Concrete strength class (i.e.C15, C20, C25, C30 and so on),
- Steel grade
- Reinforcement grade

Implementing the instruction in C# required both Back-End and Front-End programming. Front end constituted 'user interface' graphics design and arrangement of items on the forms. Back end programming constituted coding the 'engine' running calculations logics steps, as shown in figures 3.3 and 3.4.

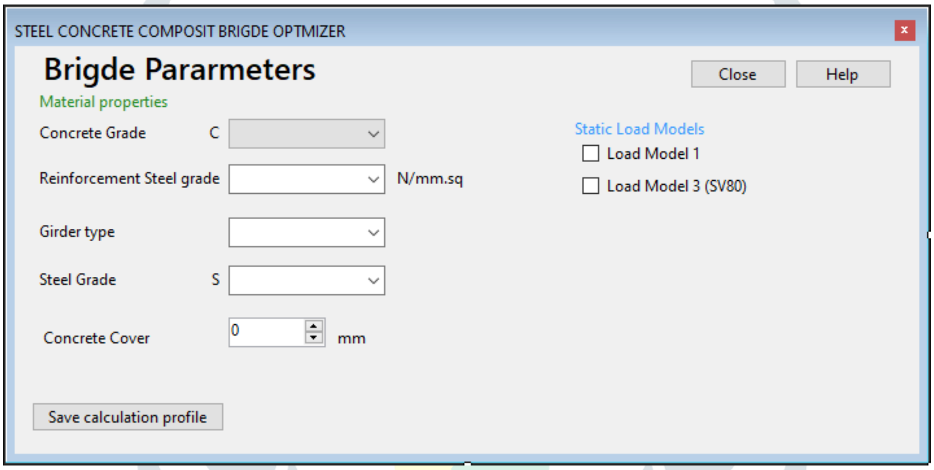

Figure 3.3: Form 1 showing Front End programming (User Interface Design)

```
1 reference
public void button1_Click(object sender, EventArgs e)
{
    if (comboBox2.Text=="" ||comboBox1.Text=="" || comboBox3.Text==""||comboBox4.Text=="")
    {
        MessageBox.Show("Please Enter all the bridge parameters before you can proceed to the next page...", "Error");
    }
    else
    {
        concrete_grade = float.Parse(comboBox1.Text);
        rein_steel_grade= float.Parse(comboBox2.Text);
        girder_type= comboBox3.Text;
        steel_grade= float.Parse(comboBox4.Text);
        concrete_cover= float.Parse(numericUpDown1.Text);

        var Form2 = new Form2();
        Form2.Show();
        this.Hide();
        //save data to database

    }

}
```

Figure 3.4: Illustration of Back end programming (Event handler for the 'Next Button')

Step 3: Slab Decking Design

Concrete slab deck design comes after the topology parameters are assigned by the user. Both live loads and dead loads are considered in this step. The bridge decking is the structural component that receives the live loading first. Therefore, logic entails that it be design first.

Live load models outlined in the Eurocode are considered and the slab deck is designed using unit strip method accounting for the dead loading as well.

Design of the slab decking serves as a constraint function in the optimization general function. The following check had to be satisfied for a design the program could to the next step.

1. Check for the cover

The program checked if the entered value for the cover was greater or equal to the cover derived from provisions given in BS EN 1992-1-1:2004 Equation 3.4 (British Standards Institute, 2004)

$$C_{nom} = C_{min} + \Delta C_{dev} \;\;\ldots. Equation\; 3.4$$

Where values for $C_{min,dur}$ were taken obtained from BS 8500-1. (British Standard Institute, 2006)

2. Actions on the Bridge Decking

Permanent actions considered in the design include:

- Reinforced concrete slab with unit weight of 25kN/m^3
- Surfacing with a unit weight of 24kN/m^3

Variable action considered in the include

- Traffic Load Model 1 following the provisions shown in BS EN 1991-2 2003. (British Standards Institute, 2003)

3. Flexural Design Checks

The program also checked if the slab deck section had adequate moment by use of Equation 3.5.

$$M_c \geq M_{ult} \;\;\ldots. Equation\; 3.5$$

where M_c is the flexural strength of the section and M_{ult} is the ultimate moment imposed in the slab decking.

4. Shear Checks

Similary, the program checked for shear capacity of the section by use of Equation 3.6.

$$V_c \geq v \;\;\ldots. Equation\; 3.6$$

where V_c is the shear capacity of the section and v is the applied shear force.

5. Deflection checks

The program lastly, checked if the maximum deflection was less than the permissible deflection.

Step 4: Steel Girder Design

The structural design of the steel girders came after the 'slab decking design' step. This is because the girders receive the loads from the slab decking and are obviously the next structural element in the load path.

Standard sections were used instead of plate girders to limit the scope and to increase variations in options. The methodology approach assumed that the beams are fully retrained due to the shear studs in composite bridges. The following checks were made in each iteration:

1. Resistance of cross section to bending ULS

The program made the following check:

$$\frac{M_{Ed}}{M_{c,Rd}} \leq 1.0 \;\;\ldots. Equation\; 3.7$$

where M_{Ed} is the design value of the bending moment and M_{cRd} is the design resistance for bending about one principal axis which takes different forms depending on the classification of the section.

2. Resistance to shear forces and buckling ULS

Equation 3.8 below checked for section resistance to shear loading.

$$\frac{V_{Ed}}{V_{c,Rd}} \leq 1.0 \;\;\ldots. Equation\; 3.8$$

where V_{Ed} is the design shear force imposed on the member and $V_{c,Rd}$ is taken as the design shear resistance of the section.

The shear buckling check utilized Equation 3.9. (clause 5.1, EC 3-5)

$$\frac{h_w}{t_w} > 72\frac{\varepsilon}{\eta} \;\;\ldots. Equation\; 3.9$$

3. Resistance to flange induced buckling ULS

For flange induced buckling check, Eurocode 3-5 requires that the ratio h_w/t_s should satisfy the following expression:

$$\frac{h_w}{t_w} > k\frac{E}{f_{yf}}\sqrt{\frac{A_w}{A_{fc}}} \;\ldots\ldots Equation\ 3.10$$

where A_w is the area of the web, A_{fc} is the area of the compression flange, and f_{yf} is the yield strength of the compression flange.

4. Resistance of the web to transverse forces ULS

Design resistance of webs of standards sections was checked according to Clause 6 of Euro Code 3-5. Equation 3.11 is the expression for local buckling.

$$F_{Rd} = \frac{f_{yw}L_{eff}t_w}{\gamma_{M1}} \;\ldots . Equation\ 3.11$$

where F_{Rd} is the design resistance of webs to local buckling, t_w is the thickness of the web, γ_{M1} is the partial safety factor = 1.0 and L_{eff} is the effective length of the web.

5. Defection SLS

Permissible deflection was as per Equation 3.12.

$$Permissible\ deflection \leq \frac{Span}{360} \;\ldots . Equation\ 3.12$$

3.2. Development of COMPOPT

A number of factors had to be considered when undertaking this task. The following steps were taken when developing COMPOPT.

1. Development of an algorithm that the program was to follow.
2. Secondly, choice of programming language the ‘optimization program’ had to written in.
3. Using the appropriate syntax, the code for each block of instructions following the order of the algorithm developed in step one (1) was written.
4. The program was compiled and packaged. The errors noticed were corrected until the program ran smoothly.

3.3. Experimental Design Example

In order to test the Output Solution obtained from the Software, a comparative study was done. COMPOPT solution were compared with a typical design that would be obtained using the conventional method. Design Solutions obtained using the conventional method, five (5) practicing structural engineers were asked to design a Composite bridge, given the same conditions. Their results were then compared to the output from the software, for the design problem described below.

Design Problem: Design a simply supported Concrete-steel composite bridge. The deck carries a 100mm thick surfacing, together with traffic load (LM1) udl with 5.5 kN/m^2 and tandem axle load of 100 kN (300 kN/3m lane width). You may use any number of Piers you deem necessary for the crossing and also any number girders supporting the slab decking, given the following parameters for the crossing:

- Crossing span: 20m
- Carriage way width: 10m
- Use Universal Beams as Girders

For the purposes of comparison, the engineer’s solutions are summarized in Table 4.2. The comparison was based on the following:

- Weight of concrete for the decking,
- Area of tension reinforcement per unit width,
- Total weight of girders required, and
- Number of piers required.

3.4. Comparison of Results and discussion

3.4.1. Design Output from COMPOPT

The results from the Optimization Tool are summarized in Table 4.1.

Table 4.1. Summary of Design Solution from COMPOPT

Parameters	**Results**
Bridge Topology Parameters	
Crossing Span	20m
Carriage way width	10m
Pier - Pier Spacing	5m c/c
Main Girder-Girder Spacing	2.5m
Slab Decking	
Slab thickness	233.57mm
Tension Main Reinforcement	T32@250mm
Tension Reinforcement Area	3220mm^2/m run
Girder System	
Main Girder Section	4No. 356x171x67UB
Main Girder Spacing	2.5m
Piers	
Pier-Pier Spacing	3No. Pier @ 5m

From the above results, the following calculations were determined:

1) Weight of Concrete required for the Decking

$$Weight\ of\ Concrete = 24(Crossing\ Span\ \times\ Carriage\ way\ width\ \times slab\ thickness)$$
$$Weight\ of\ Concrete = 24kN/m^3\ \times\ 20m\ \times 10m\ \times 0.25m$$
$$Weight\ of\ Concrete = \mathbf{1200\ kN}$$

2) Area of Tension Reinforcement per unit width

$$Area\ of\ tension\ rebar\ required = \mathbf{3220mm^2/m\ run}$$

3) Total weight of girders required

-Girder Selected: 4No. 356x171x67UB
-Unit weight: 67.1kg/m

$$Total\ weight\ of of\ girders = Unit\ weight\ \times crossing\ span\ \times No.\times 10m/s^2$$
$$Total\ weight\ of of\ girders = 67.1kg/m\ \times 20m\ \times 4No.\times 10m/s^2$$
$$Total\ weight\ of of\ girders = \mathbf{53.68\ kN}$$

4) Numbers of Piers

$$Number\ of\ Piers = \mathbf{3No.}$$

3.4.2. Comparison of Results from Structural Engineers and COMPOPT

Table 4.2 and Figure 4.2.1 to 4.2.4 summarize the comparisons.

Table 4.2: Comparison of Results

	Optimization Tool (COMPOPT)	Engineer 'A'	Engineer 'B'	Engineer 'C'	Engineer 'D'	Engineer 'E'
Deck slab thickness	250mm	400mm	300mm	300mm	250mm	300mm
Weight of Decking (kN)	1200	1920	1440	1440	1200	1440
Area of Tension Rebar Required (mm^2/m run)	3220	4600	3930	5360	4600	5360
Girder Section Chosen	356x171x67 UB	533x210x109 UB	406x178x74 UB	457x152x67 UB	406x178x67 UB	533x210x82 UB
No. of Girders (No.)	4	4	5	3	4	4
Total Girders Weight (kN)	53.67	87.2	74.2	67.2	62.68	65.76
No. of Piers Required (No.)	3	1	2	3	3	3

3.4.3. Weight of Concrete Deck Comparison

A comparison of the weight of the decking from each solution is shown in Figure 4.2.1. it is observed that the solution from ‘Engineer A’ had the heaviest bridge decking with a weight of 1920kN. This decking had the highest depth of 400mm. The solutions from ‘Engineer C’ and the ‘COMPOPT’ had the least weight of 1200 KN for a deck depth of 400mm. It was also clear that the depth of the bridge deck slab was found to be the highest contributing factor to the overall weight of the deck.

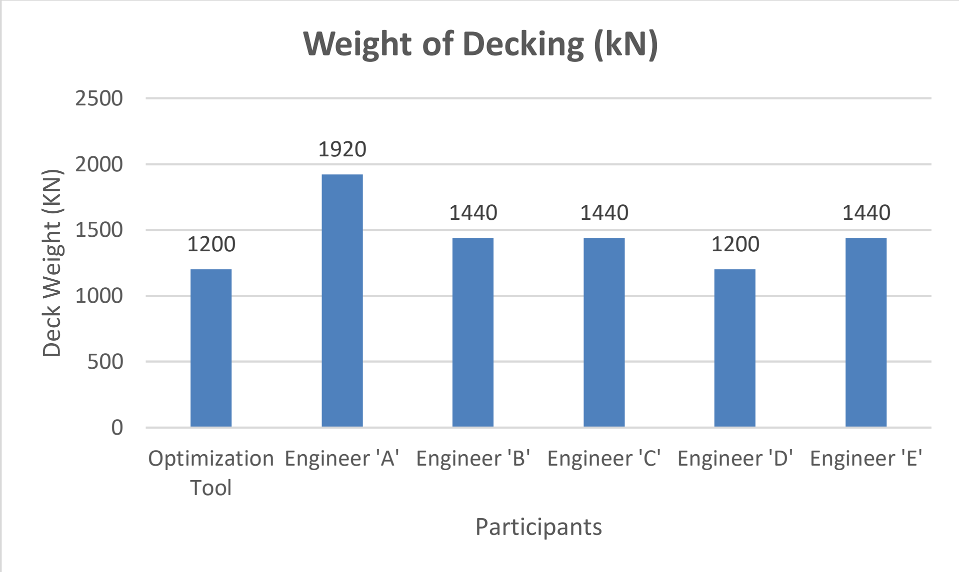

Figure 4.2.1. Weight of Bridge Decking

3.4.4. Comparison of Area of Tension Reinforcement Required

Figure 4.2.2 is a bar chart comparing the area of reinforcement required on each solution.

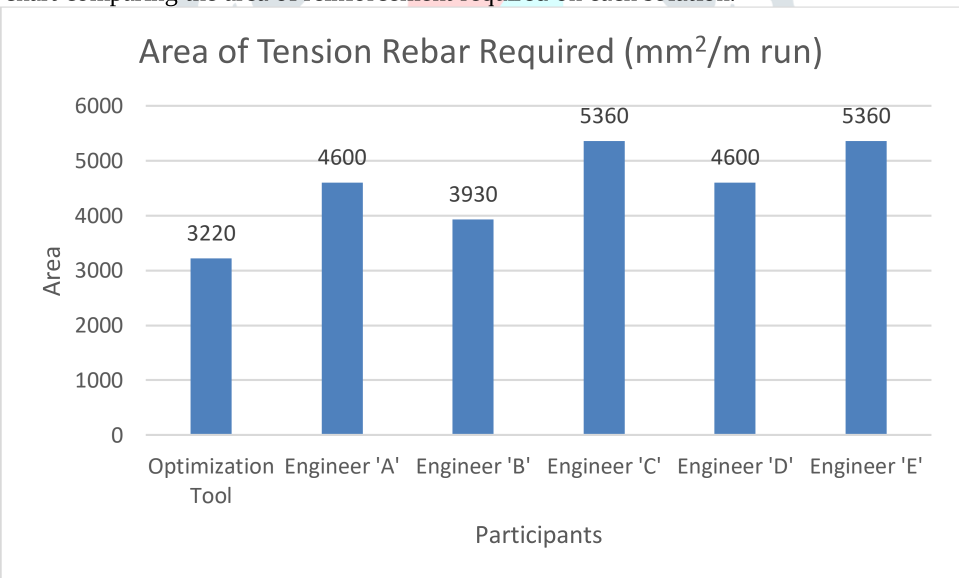

Figure 4.2.2. Area of Tension Rebar Required for each solution

Engineers E and C obtained the highest amount of Tension reinforcement per unit strip of the bridge decking. The solution from COMPTOPT gave the least required area of Tension Rebar.

3.4.5. Comparison of Total Weight of Girders Required

Figure 4.2.3 is a comparison of the girder sizes obtained in each solution. COMPTOPT gave the least weight of girders whilst Engineer A obtained the heaviest girders.

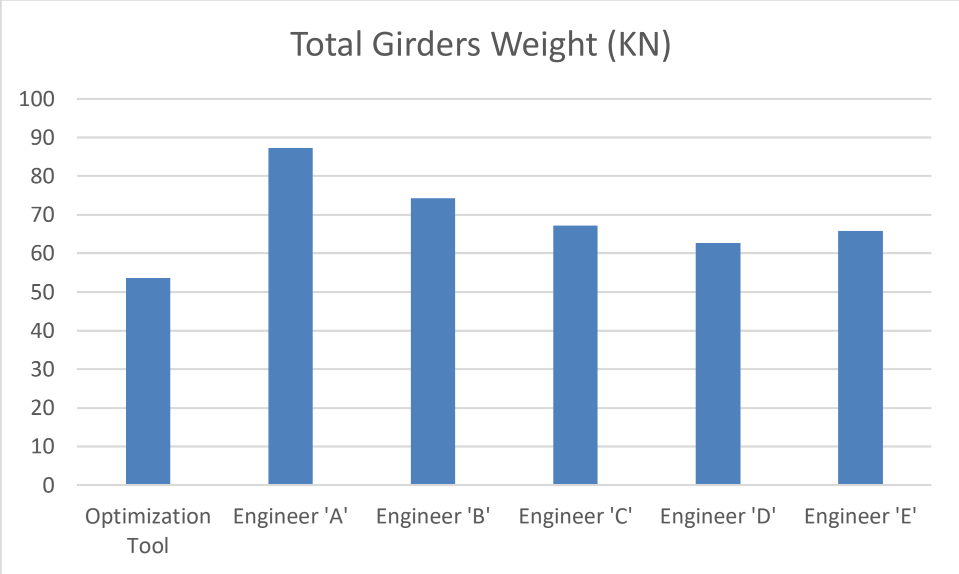

Figure 4.2.3. Comparison of Total Girder Weight in each solution

3.4.6. Comparison of the Total Number Piers Required

Figure 4.2.4 compares the number of piers used per each design output. It is clear that the solution from the optimization tool agrees with most of the solutions from the practicing Engineers.

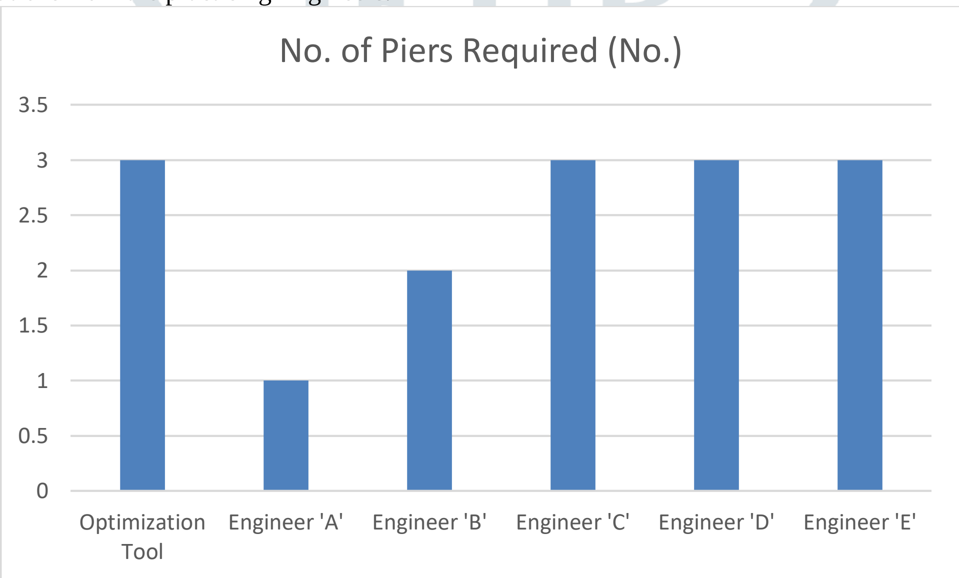

Figure 4.2.4. Comparison of Number of piers Required

4. Conclusions and Recommendations

The aim of the comparative study was to test the output from the optimization tool against conventional designs. As anticipated, the output from the optimization tool performed well on most of the parameters as discussed in the previous chapter. This chapter outlines the conclusions, recommendations and limitations of the study.

4.1. Conclusions

4.1.1. Identifying the design criteria used for concrete-steel composite bridges.

Literature was reviewed on the methods used to design concrete –steel composite bridges. The literature showed that the methods used here in Zambia conform to the Eurocode and the British standard, BS 5400.

For bridge deck design, the unit strip method was used by all 5 Practicing Engineers who participated in the study. Even though two of them used Prokon and Robot as design aids, their main design approach for deck design is the Unit Strip method. The girders were all designed using the checks outlined in the codes.

4.1.2. Identifying optimization techniques and algorithms that are useful in composite bridge design.

From the desk study conducted, the research revealed that there are broadly two categories of Optimization Techniques namely, Advanced Optimization techniques and Classical Optimization techniques. The study focused on the latter due to practicality of the solutions (Design Output) given the design problem presented in section 3.3. The study also showed that most of classical methods are more useful and practical of the two categories. These are able to reach optimum solutions deterministically given that the problem isn't too complex. Advanced methods are more applicable to complex problems where an optimum solution can't be found analytically. Moreover, advanced techniques only guarantee a near optimum solution.

To demonstrate the use and practicality of classical methods, a software tool to optimize topology parameters in the given bridge was developed. The tool is based on the direct search method using nested for-loops in C-Sharp programming language.

4.1.3. Development of a software tool for optimizing the sizing of girders and topology of the composite bridge design.

This specific objective was met by developing the optimizing software tool described in section 3. The program is able to optimize the bridge deck slab, Universal Beam selection and spatial position of piers. The solutions from this tool were compared to various design solutions obtained from practicing structural engineers in the industry.

4.1.4. Comparison of the 'optimal design' with that obtained from the conventional design.

A comparative study was based on four (4) parameters as described in Section 4.7. The parameters included:

- Weight of concrete deck.
- Area of Tension Reinforcement
- Weight of Girders in the solution
- Number of Piers in the solution

4.2. Recommendations

The comparative study showed that COMPOPT scored favorably on all four parameters. The study also showed that the concrete decking contributes most significantly to the overall weight of the bridge and that the software is useful for preliminary design stage. For further studies, more features can be added to COMPOPT to include optimal preliminary design of piers and foundations

References

[1] British Standard Institute, 2006. *BS8500-1: Concrete-Complementary British Standard to BS EN 206-1: Method of Specifying and guidance to the Specifier*, London: BSI.

[2] British Standards Institute, 2003. *Eurocode 1: Actions on structures- Part 1.2: Actions on structures. Traffic Loads on Bridges,* London: BSI. Print.

[3] British Standards Institute, 2004. *Eurocode 2: Design of Concrete Structures- Part 1-1: General rules and rules for buildings,* Londoon: BSI. Print.

[4] Chanakya, A., 2009. *Design of Structural Elements: Concrete, Steelwork, Masonry and Timber Designs to British Standards and Eurocodes.* 3rd ed. New York: Spon Press.

[5] Chapra, S. C. & Canale, R. P., 2015. *Numeral Methods for Engineers.* 7th ed. New York: Mcgraw-Hill Education.

[6] Chopard, B. & Tomassini, M., 2018. *An Introduction to Metaheuristics for Optimization.* 1st ed. Cham: Springer.

[7] Fabeane, R., Kripka, M. & Pravia, Z. M. C., 2017. COMPOSITE BRIDGES: STUDY OF PARAMETERS TO OPTIMIZED DESIGN. *International Journal of Bridge Engineering,* v(2), pp. 1-20.

[8] Hanson, R. D., 2004. A research C# compiler. *Software Practice and Experience,* Issue 34, p. 1211–1224.

[9] Himani, A. & Dr. Monisha, S., 2015. OPTIMIZATION OF NTRU CRYPTOSYSTEM USING ACO ALGORITHM. *International Journal of Engineering Research-Online,* III(2), pp. 357-368.

[10] Kazakis, G., Kanellopoulos, I., Sotiropoulos, S. & Lagaros, N., 2017. Topology optimization aided structural design: Interpretation, computational aspects and 3D printing. *Heliyon,* III(1), pp. 1-33.

[11] Kazakis, G., Kanellopoulos, I., Sotiropoulos, S. & Lagaros, N. D., 2017. Topology optimization aided structural design: Interpretation, computational aspects and 3D printing. *Heliyon,* 3(10), pp. 1-33.

[12] Lythell, M. & Stenberg, J., 2020. *Cost optimization of composite bridges,* Stockholm: KTH Royal Institute of Technology.

[13] Numan, G., 2012. *Design and optimization of steel portal frames according to Eurocode using genetic algorithm,* Lunds: Lunds University press.

[14] Pedro, R. L., Miguel, L. F. F., Demarche, J. D. & Lopez, R. H., 2017. An efficient approach for the optimization of simply supported steel-concrete Composite I-Girder Bridges. *Advances in Engineering Software.*

[15] Singiresu, R. S., 2019. *Engineering Optimization: Theory and Practice.* 5th ed. Hoboken: John Wiley & Sons Inc..

[16] Skiena, S. S., 2020. *The Algorithm Design Manual.* 3rd ed. Cham: Springer.