# Prestressed Concrete I-Girder Optimization via Genetic Algorithm

**Tito Adibaskoro*** **& Made Suarjana**

Faculty of Civil and Environmental Engineering, Institut Teknologi Bandung, Jalan Ganesa No.10, Bandung 40132, Indonesia
*E-mail: titoadibaskoro@gmail.com

**Abstract.** Prestressed concrete has been gaining popularity in the construction industry because of its many advantages, which include reduced dead load due to less material used and overall cost savings. Nonetheless, a single prestressed concrete I-girder as a structural element in highway bridges is still significantly costly and massive, so optimization can yield a significant amount of cost savings as well as reduced material consumption. In this study, prestressed concrete I-girder optimization was carried out by implementing a genetic algorithm (GA), a method inspired by nature's evolution and natural selection. This study evaluates a number of aspects of applying a genetic algorithm for optimization of material cost of a prestressed concrete I-girder design. A new method for calculating the fitness value is proposed, which was proven to be essential for the application developed in this study. The best solution that resulted from the optimization process is presented, defined by being the least costly solution while still maintaining compliance with the AASHTO LRFD 2007 design code, which includes ultimate strength, service stresses and deflection, detailing requirements, geometrical feasibility, etc. Lastly, a sensitivity analysis was carried out, discussing the influence of the starting conditions on the output of the optimization process.

**Keywords**: *genetic algorithm; highway bridges; i-girder; optimization; prestressed concrete.*

## 1 Introduction

The use of prestressed concrete in the construction industry has increased to the point where nowadays it is rare for major construction projects not to consider prestressed concrete as a viable alternative solution [1]. However, the reduction of the initial cost is not substantial because the savings in material consumption are usually balanced out by the need for better material quality and more complex formwork. In the long run, however, lower maintenance cost and the longer lifespan make prestressed concrete more economical [2]. Nevertheless, a single prestressed concrete element can still be significantly costly and consume a large amount of material, so attempts at optimization can easily be worth the effort.

Received March 23rd, 2018, Revised October 5th, 2018, Accepted for publication February 19th, 2019.
Copyright ©2019 Published by ITB Journal Publisher, ISSN: 2337-5779, DOI: 10.5614/j.eng.technol.sci.2019.51.2.2

The genetic algorithm (GA) has become a popular tool for tackling complex optimization problems such as those in engineering design due to its mathematical and computational simplicity, its ability to work through search spaces with many local optima, its capability of handling non-differentiable and non-continuous functions, and its capability to be set up for discrete search spaces (structural design variables tend to be discrete) [3-8]. The idea of the genetic algorithm was first introduced and systematically investigated by Holland [9-11], proving its robustness and flexibility. Today, genetic algorithms are widely applied in engineering for design optimization, manufacturing and other processes [12].

Because the method is stochastic, genetic algorithms produce one or several closely optimized solutions instead of only the theoretically most optimal one. This, in fact, is very advantageous in the process of engineering design, as sometimes unquantifiable engineering judgment can play a major role in the decision making process.

## 2 Objectives and Limitations

The objective of this study was to utilize a genetic algorithm for optimizing the cost of prestressed concrete I-girders for highway bridges. In this paper, deliberate addition of discontinuity to the fitness function is proposed, which will be proven to be necessary in this application of GA. Furthermore, the effects of different parameters and initial conditions on the overall performance of the optimization process were also investigated.

This study assumed the girders to be simply supported. They had a span length of 30 m and center-to-center girder spacing of 2 m, while AASHTO LRFD 2007 [13] was used as the design code. Only material costs were considered, calculated based on the volume of materials used. Concrete and mild steel unit costs were based on the *Journal of Building Construction, Interior & Material Prices 2017* [14], while unit prices for prestressing steel were based on the study presented in [15], adjusted for inflation [16].

## 3 Optimization Method

### 3.1 Encoding

In order to be manipulated by genetic operators, the design parameters were encoded into a string of characters called a chromosome. The design parameters were mapped via binary encoding, with the chromosomes being made up of a series of ones and zeros. To cover the whole search space, each chromosome was set to a length of 36 digits, 13 of which represented design parameters, 8 determined the concrete cross-sectional geometry (see Figure 1), 1 defined the compressive strength of concrete, and the rest described the tendon geometry and jacking parameters. All of the design parameters are presented in Table 1, arranged according to their allocated locations in the chromosome (the K-value designates the uniaxial compressive strength of a 150 mm concrete cube specimen in kgf/cm$^2$).

**Table 1** Girder design parameters.

| Binary (bit(s)) 1 | 2 | 3 | 4 | K- (kgf/cm$^2$) | $h_{f11}$ (mm) | $h_{f12}$ (mm) | $h_{wb}$ (mm) | $h_{f21}$ (mm) | $h_{f22}$ (mm) | $b_{f1}$ (mm) | $b_{f2}$ (mm) | $b_w$ (mm) | Strand diameter (inch) | Strands per tendon | Number of tendons | Jacking force (UTS) |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 | 00 | 000 | 0000 | 300 | 150 | 50 | 800 | 50 | 150 | 300 | 300 | 100 | 0.5 | 7 | 1 | 0.65 |
| 1 | 01 | 001 | 0001 | 325 | 200 | 100 | 900 | 100 | 200 | 400 | 400 | 150 | 0.6 | 12 | 2 | 0.7 |
| - | 10 | 010 | 0010 | 350 | 250 | 150 | 1000 | 150 | 250 | 500 | 500 | 200 | - | 19 | 3 | 0.75 |
| - | 11 | 011 | 0011 | 375 | 300 | 200 | 1100 | 200 | 300 | 600 | 600 | 250 | - | 22 | 4 | 0.8 |
| - | - | 100 | 0100 | 400 | 350 | 250 | 1200 | 250 | 350 | 700 | 700 | 300 | - | 27 | - | - |
| - | - | 101 | 0101 | 450 | 400 | 300 | 1300 | 300 | 400 | 800 | 800 | 350 | - | 31 | - | - |
| - | - | 110 | 0110 | 500 | 450 | 350 | 1400 | 350 | 450 | 900 | 900 | 400 | - | - | - | - |
| - | - | 111 | 0111 | 550 | 500 | 400 | 1500 | 400 | 500 | 1000 | 1000 | 450 | - | - | - | - |
| - | - | - | 1000 | 600 | - | - | - | - | - | - | - | - | - | - | - | - |
| - | - | - | 1001 | 650 | - | - | - | - | - | - | - | - | - | - | - | - |
| - | - | - | 1010 | 700 | - | - | - | - | - | - | - | - | - | - | - | - |
| - | - | - | 1011 | 750 | - | - | - | - | - | - | - | - | - | - | - | - |
| - | - | - | 1100 | 800 | - | - | - | - | - | - | - | - | - | - | - | - |
| - | - | - | 1101 | - | - | - | - | - | - | - | - | - | - | - | - | - |
| - | - | - | 1110 | - | - | - | - | - | - | - | - | - | - | - | - | - |
| - | - | - | 1111 | - | - | - | - | - | - | - | - | - | - | - | - | - |

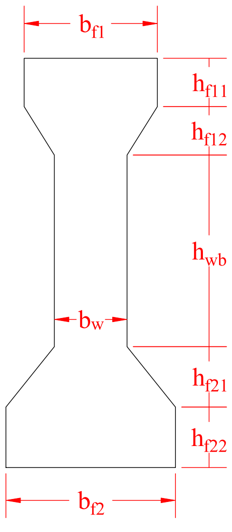


**Figure 1** Section geometry variables.

### 3.2 Genetic Operators

The genetic operators utilized in this study were crossover and mutation. Both were set to have a certain probability of occurrence. Mutation occurs in the form of bit flipping, a natural choice for the case of binary encoding. Crossover occurs at multiple points instead of just one, as multi-point crossover has been proved to be more robust in preventing premature convergence [17]. The mutation probability was set to 0.008, while the crossover probability was set to 1.

### 3.3 Evaluation

Evaluation is the process of quantifying the success of individuals by assigning them fitness values. In this process, every individual is decoded into design parameters and then evaluated against the optimization objective and predefined constraints.

#### 3.3.1 Unconstrained Fitness Function

As the goal of the optimization process in this study was to minimize cost, an individual that offered lower cost scored higher after having been evaluated through the unconstrained fitness function. The unconstrained fitness function used in this study is expressed in Eq. (1).

$$
\begin{aligned}
&f_{\text{unpenalized}}\left(x_{i,j}\right) = P_j - \text{cost}\left(x_{i,j}\right)\\
&P_j = 1.5 \times \max\Big(\text{cost}\left(x_{1,j}\right);\ \text{cost}\left(x_{2,j}\right);\\
&\qquad\qquad \text{cost}\left(x_{3,j}\right);\ \cdots\ ;\ \text{cost}\left(x_{m,j}\right)\Big)
\end{aligned}
\tag{1}
$$

#### 3.3.2 Penalty Function

The penalty function is meant to reduce the unconstrained fitness value if one or more constraints are violated. The constraints considered in this study were derived from the assessments of stresses for service condition (compression and tension at transfer, permanent service load, and service limit state), ultimate flexural resistance and ductility requirements, ultimate shear resistance and transverse reinforcement limits, web slenderness ratio, deflection (immediate and long-term), and geometrical feasibility of the tendon’s (without prestressing ducts or anchorages colliding against each other).

A penalty is applied in the following manner in Eqs. (2) and (3):

$$
f_{\text{penalized}}\left(x_{i,j}\right) = f_{\text{unpenalized}}\left(x_{i,j}\right) - f_{\text{penalty}}\left(x_{i,j}\right) \geq 0 \tag{2}
$$

$$f_{\text{penalty}}\left(x_{i,j}\right)=r_j\left(\sum_{k=1}^{n}\left(d_k\left(x_{i,j}\right)\right)^2\right) \tag{3}$$

In order to assure effective scaling, $r_j$ is set to be as large as possible, yet no $f_{\text{unpenalized}}(x_{i,j})$ is reduced by more than one half.

### 3.3.3 Additional Discontinuity

A deliberate discontinuity is introduced between feasible and infeasible regions by further reducing the fitness values of infeasible solutions for the sole reason of being infeasible. The application of such discontinuity is described in Eqs. (4) and (5) and visualized in Figure 2. Similar to the previous description, $r_j$ is set to be as large as possible, yet no $f_{\text{unpenalized}}(x_{i,j})/A$ is reduced by more than one half.

$$f_{\text{penalized}}\left(x_{i,j}\right)=\frac{f_{\text{unpenalized}}\left(x_{i,j}\right)}{A}-r_j\left(\sum_{k=1}^{n}\left(d_k\left(x_{i,j}\right)\right)^2\right)\geq 0 \tag{4}$$

$$A=\begin{cases} 1 & \text{for } \sum_{k=1}^{n}\left(d_k\left(x_{i,j}\right)\right)^2=0 \\ 2 & \text{for } \sum_{k=1}^{n}\left(d_k\left(x_{i,j}\right)\right)^2>0 \end{cases} \tag{5}$$

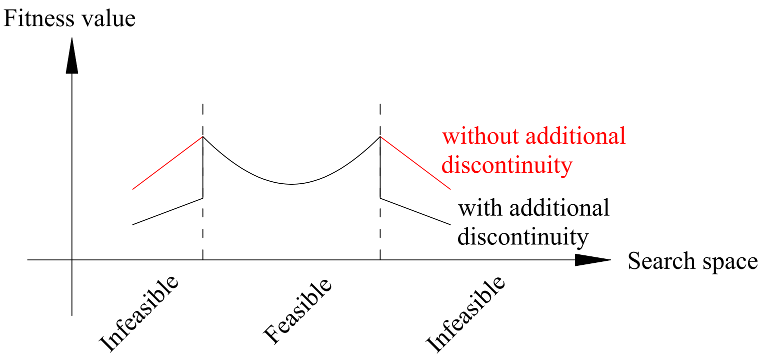


**Figure 2** Visualization of the additional discontinuity.

## 3.4 Selection

The goal of selection is to allow individuals with high fitness to dominate the population and the ones with low fitness to be reduced in number or eliminated. The selection process is carried out by employing roulette wheel with elitism.

### 3.4.1 Roulette Wheel

Roulette wheel is applied in order to pick the individuals to be passed on to the next generation. The probability of each individual being selected is proportional to the associated fitness value, as described in Eq. (6).

$$p_{i,j} = \frac{f_{\text{penalized}}\left(x_{i,j}\right)}{\sum_{i=1}^{m} f_{\text{penalized}}\left(x_{i,j}\right)} \tag{6}$$

Consequently, successful individuals are more likely to be picked more than once while unsuccessful ones are less likely to be selected.

### 3.4.2 Elitism

First introduced by De Jong [19], elitism allows the algorithm to keep some of the fittest individuals ('elite members') from the roulette wheel and genetic operators. Applying elitism improves performance since losing and then rediscovering the best individuals wastes significant computation time [4,18]. The proportion of the population that is considered to be elite members is called the elite proportion, which was set to 0.55.

### 3.4.3 Convergence

The whole optimization algorithm is terminated after convergence is achieved. Convergence represents the process having reached an optimum solution, indicated by the standard deviation of the fitness values of all individuals, where that of the most recent generation is smaller than a predetermined value, which was set to 0.01 in this study.

## 4 Prestressed Concrete I-Girder Optimization

### 4.1 Optimization Result

The most optimized design is displayed as console output in Table 2 (dimensions in mm if not mentioned), while the anchoring device and reinforcement layouts are shown in Figure 3 (first from the left) and Figure 4. In order to make sure that the prestressing ducts do not collide with each other, a check was carried out every 0.3 meter. Figure 3 (second, third, and fourth from the left) shows examples of some sections.

## 4.2 Difference from Conventional Design

The most obvious difference between the optimized design and a conventional design shown was the choice of concrete strength. In a conventional design, picking higher concrete strength (K-450 or higher) is more sensible, allowing a wider range of stress between compression and tension. Instead, the optimization algorithm picked a less costly K-375 concrete and was able to find the small window of code-compliant and cost-effective designs. Do bear in mind that using even lower concrete strength is possible, but the cross-sectional area will have to be larger so that the design actually becomes less cost-effective. In a conventional design process, finding out that a certain lower-strength concrete will lead to the most cost-effective result and discovering a suitable design would be very difficult to realize.

**Table 2** Most Optimum Design

```
Cost : 68.5543 Million Rupiahs
Chromosome : 10110010101011000010100100101001011
Number of Errors : 0

Specifications (from Chromosome) :
K = 375
hf11 = 200
hf12 = 150
hwb = 1300
hf21 = 250
hf22 = 200
bf1 = 500
bf2 = 500
bw = 200
StrandDiameterInch = 0.6
StrandperTendon = 12
NumberofTendons = 2
InitialTendonPrestressperUTS = 0.8

Specifications (from Calculation) :
Number of Top Longitudinal D-22 Rebars: 5
Number of Bottom Longitudinal D-22 Rebars: 7
Transverse D-13 Spacing at 0 mm < x < 3000 mm : 200 mm
Transverse D-13 Spacing at 3000 mm < x < 6000 mm : 200 mm
Transverse D-13 Spacing at 6000 mm < x < 9000 mm : 400 mm
Transverse D-13 Spacing at 9000 mm < x < 12000 mm : 600 mm
Transverse D-13 Spacing at 12000 mm < x < 15000 mm : 600 mm
Transverse D-13 Spacing at 15000 mm < x < 18000 mm : 600 mm
Transverse D-13 Spacing at 18000 mm < x < 21000 mm : 600 mm
Transverse D-13 Spacing at 21000 mm < x < 24000 mm : 400 mm
Transverse D-13 Spacing at 24000 mm < x < 27000 mm : 200 mm
Transverse D-13 Spacing at 27000 mm < x < 30000 mm : 200 mm
Stirrup D-16 Locations for Local Bursting (mm) at Tendon Number 1 : 202.5, 262.5
Stirrup D-16 Locations for Local Bursting (mm) at Tendon Number 2 : 202.5, 262.5
Stirrup D-19 Locations for Bursting at General Zone (mm) : 160, 220, 280, 340,
   882.6675, 942.6675, 1002.6675, 1062.6675, 1122.6675
Number of D-22 Spalling Rebars : 2
```

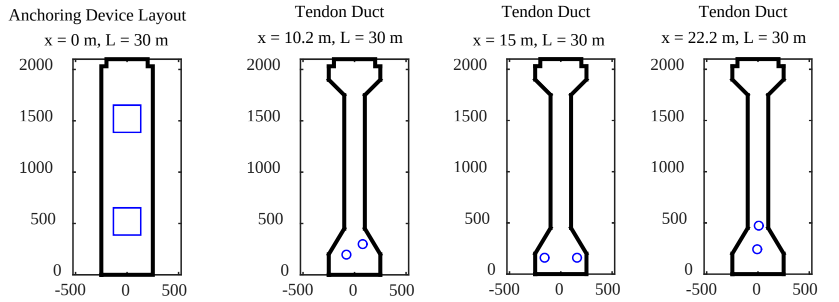


**Figure 3** Anchoring device layout and prestressing duct in different locations.

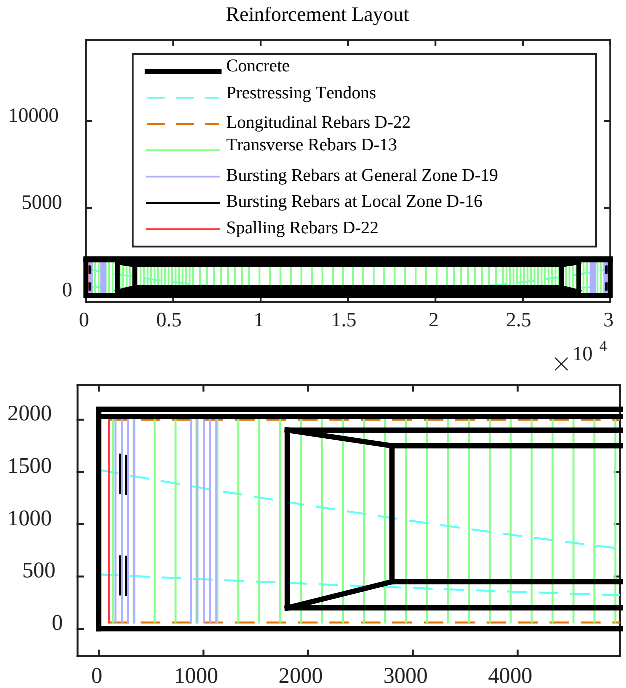


**Figure 4** Reinforcement layout.

## 5 Algorithm Evaluation

The parameters in a GA need to be set by considering the balance between the tendency of exploring solutions within the search space and exploiting discoveries made throughout the optimization process [4]. In this section, these parameters and other aspects are evaluated, including the proposed additional discontinuity and sensitivity analysis. While one parameter was investigated, the others followed the default values and settings described in Section 3.

## 5.1 Elite Proportion

Elite proportion was investigated in the range of 0 (no elite members) to 1 (every individual is elite an member). A smaller elite proportion leans towards exploration while a larger elite proportion towards exploitation.

As shown in Figure 5, the optimization result is barely affected by the value of the elite proportion, but the variation in terms of steps required to achieve convergence is significant.

On the extreme left of Figure 5 (right), convergence is not achieved (note that the maximum number of steps allowed is 72), meaning that elitism is necessary. The number of steps to convergence is at its lowest around elite proportion values of 0.45-0.65. Therefore, the value of elite proportion was set to 0.55.

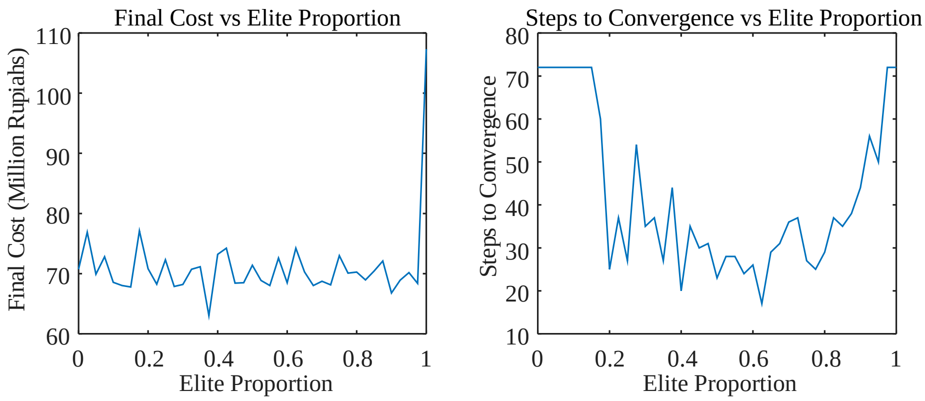


**Figure 5** Average final cost and steps to convergence vs elite proportion.

## 5.2 Mutation Probability

Because the mutation probability needs to be kept small, the evaluation was carried out with mutation probability in the range of 0 and 0.1. As can be seen in Figure 6, a trend between final cost and mutation probability cannot be concluded, but the mutation probability significantly affected the number of steps required to reach convergence. In fact, a valid solution was not reached at a mutation probability of about 0.075-0.1, confirming the necessity of keeping this value small.

In order to get better insight, another evaluation was carried out with mutation probability in the range of 0-0.01, with the result presented in Figure 7. The final cost slightly decreased as the mutation probability increased towards 0.01

while the number of steps to convergence increased slightly as well. Based on the graph displayed in Figure 7, the mutation probability was set to 0.008.

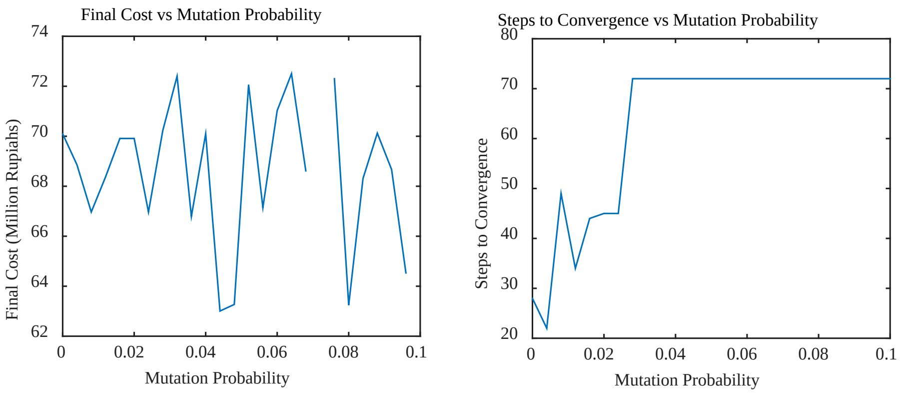


**Figure 6** Average final cost and steps to convergence vs mutation probability (1).

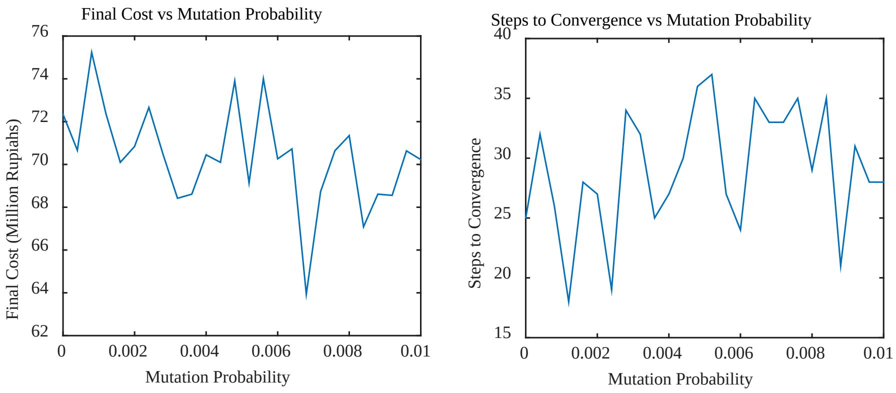


**Figure 7** Average final cost and steps to convergence vs mutation probability (2).

## 5.3 Crossover Probability

In a similar way as mutation probability, the effect of crossover probability was investigated. The relationships between crossover probability, final result and steps required to reach convergence are shown in Figure 8. Increasing crossover probability resulted in a better optimization result but more steps were required to reach convergence. Since the increased number of steps was not significant, the crossover probability was set to 1.

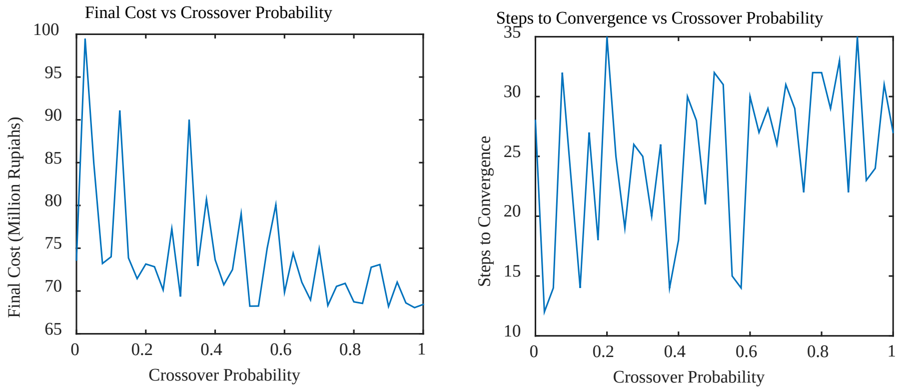


**Figure 8** Average final cost and steps to convergence vs crossover probability.

## 5.4 Additional Discontinuity

In the same way as described above, a comparison was made between cases with additional discontinuity being applied and disregarded. The result is shown in Table 3. Although convergence was reached, the result without additional discontinuity showed infinity cost, meaning that the algorithm did not converge towards a feasible solution.

**Table 3** With/without additional discontinuity comparison.

```
Average Final Cost Before Add. Discontinuity : Inf Million Rupiahs
Average Final Cost After Add. Discontinuity : 70.9038 Million Rupiahs
Average Steps Before Add. Discontinuity : 64.4
Average Steps After Add. Discontinuity : 27.6
Note: Inf cost = one or more constraints are violated
```

## 5.5 Sensitivity Analysis

This analysis aimed to see how much the algorithm is affected by the state of the initial population. Instead of being completely random, as in normal cases, the initial population in this analysis was set to be within a certain cost range (regardless of constraint violations).

As can be seen from Figure 9, with average initial cost higher than around Rp 70 million, a higher cost caused the algorithm to put more effort into the optimization process (more generations to reach convergence), yet the final cost after optimization was barely affected. Meanwhile, a lower average initial cost yielded a significant decline in both the optimization result and efficiency. The reason is that by forcing the cost to be lower than Rp 70 million, the individuals in the initial population suffer from severe constraint violation, which is not a good start for the optimization algorithm. Therefore, the extra effort required to reach convergence is unable to make up for the poor quality of the initial population.

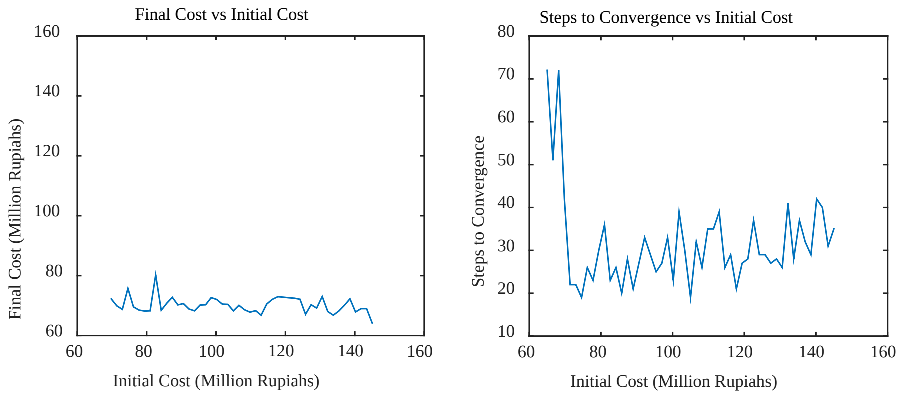


**Figure 9** Average final cost and steps to convergence vs initial cost.

## 6 Conclusion

A genetic algorithm was effectively implemented for prestressed concrete I-girder optimization. The non-standard method of elitism and the proposed additional discontinuity proposed in this paper are essential to reach a satisfactory result. Furthermore, the values of mutation probability, crossover probability and elite proportion that delivered the best optimization performance were presented and further investigated. The sensitivity analysis proved that the algorithm is robust, as the state of the initial population only affects the number of generations required to reach convergence but not the optimization result.

## Nomenclature

| | | |
|---|---|---|
| $d_k$ | = | penalty function if constraint $k$ is violated |
| $f_{\text{penalized}}$ | = | penalized fitness function |
| $f_{\text{unpenalized}}$ | = | unpenalized fitness function |
| $m$ | = | number of the population in a single generation |
| $n$ | = | total number of predefined constraints |
| $p_{i,j}$ | = | probability of individual $i$ of generation $j$ to be picked |
| $r_j$ | = | a constant for scaling the penalty function for generation $j$ |
| $x_{i,j}$ | = | individual number $i$ of generation $j$ |

## References

[1] Naaman, A., *Prestressed Concrete Analysis and Design*, 2nd ed., Michigan, United Staets, Techno Press 3000, 2011.

[2] Nawy, E.G., *Prestressed Concrete: A Fundamental Approach*, 5th ed. Upper Saddle River, Prentice Hall, 2010.

[3] Zalzala, A.M.S. & Fleming, P.J. (eds.), *Genetic Algorithms in Engineering Systems.* Stevenage, Herts, United Kingdom: The Institution of Electrical Engineers, 1997.

[4] Coley, D.A., *An Introduction to Genetic Algorithms for Scientists and Engineers*, Singapore, World Scientific, 1999.

[5] Sivanandam, S.N. & Deepa, S.N., *Introduction to Genetic Algorithms*, Berlin: Springer, 2007.

[6] Man, K.F., Tang, K.S. & Kwong, S., *Genetic Algorithms: Concepts and Designs*, Berlin: Springer, 2012.

[7] Jenkins, W.M., *Technical Note: Towards Structural Optimization via the Genetic Algorithm*, Computers & Structures, **40**, pp. 1321-1327, 1991.

[8] Gan, J. & Warwick, K., *A Genetic Algorithm with Dynamic Niche Clustering for Multimodal Function Optimisation*, in Artificial Neural Nets and Genetic Algorithms, Portoroz, pp. 248-255,1999.

[9] Holland, J.H., *Information Processing in Adaptive Systems*, in Proceedings of the International Union of Physiological Sciences, **3**, Leiden, pp. 330-339, 1962.

[10] Holland, J.H., *Genetic Algorithms and the Optimal Allocation of Trials*, SIAM Journal on Computing, **2**(2), pp. 88-105, 1973.

[11] Holland, J.H., *Adaption in Natural and Artificial Systems*, Ann Harbor, 1975.

[12] Chen, T.Y. & Chen, C.J., *Improvements of Simple Genetic Algorithm in Structural Engineering*, International Journal for Numerical Methods in Engineering, **40**, pp. 1323-1334, 1997.

[13] American Association of State Highway and Transportation Officials, *AASHTO LRFD Bridge Design Specifications: SI Units*. Washington, D.C.: American Association of State Highway and Transportation Officials, 2007.

[14] Yayasan Pandu Bangun Persada Nusantara Batavia, *Journal of Building Construction, Interior & Material Prices, 36th Ed.,* Jakarta, Indonesia, 2017. (Text in Indonesian)

[15] Zebua, F.Z. & Tarigan, J., *Comparison between Post-Tensioned and Reinforced Concrete for Floor Slabs Design*, Jurnal Teknik Sipil USU, **3**(1), 2014. (Text in Indonesian)

[16] *Worldwide Inflation Data.* Accessed from http://inflation.eu/inflation-rates/indonesia/historic-inflation/cpi-inflation-indonesia.aspx. (June 30th, 2017).

[17] De Jong, K.A. & W.M. Spears, *A Formal Analysis of the Role of Multi-point Crossover in Genetic Algorithms*, Annals of mathematics and Artificial intelligence, **5**(1), pp. 1-26, 1992.

[18] Mitchell, M., *An Introduction to Genetic Algorithms. Cambridge*, Massachusetts: Bradford Books, 1998.
[19] De Jong, K.A., *Analysis of the Behavior of a Class of Genetic Adaptive Systems*, The University of Michigan, PhD Thesis 1975.
[20] Adibaskoro, T., *Prestressed I-Girder Optimization Using Genetic Algorithm*, Thesis, Institut Teknologi Bandung, Bandung, Indonesia, 2014.