Harmony search based algorithms for the optimum cost design of reinforced concrete cantilever retaining walls

A. Kaveh¹,*, A. Shakouri Mahmud Abadi²

Received: September 2009, Revised: February 2010, Accepted: April 2010

Abstract

Cost optimization of the reinforced concrete cantilever soil retaining wall of a given height satisfying some structural and geotechnical design constraints is performed utilizing harmony search and improved harmony search algorithms. The objective function considered is the cost of the structure, and design is based on ACI 318-05. This function is minimized subjected to design constraints. A numerical example of the cost optimization of a reinforced concrete cantilever retaining wall is presented to illustrate the performance of the presented algorithms and the necessary sensitivity analysis is performed.

Keywords: Structural optimization, Reinforced concrete, Cantilever retaining walls, Harmony search algorithm, Modified harmony search

1. Introduction

Earth retaining structures constitute an integral part of the infrastructure and reinforced concrete retaining walls as earth structures are frequently constructed for a variety of applications, most commonly for bridge abutments, road, transportation systems, lifelines and other constructed facilities.

In order to economize the cost of the reinforced concrete retaining walls under design constraints, the designer needs to vary the dimensions of the wall several times, making design process rather tedious and monotonous. Since it is extremely difficult to obtain a design satisfying all the safety requirements, it is beneficial to cast the problem as an optimization problem. Some studies have been made in this direction by Dembicki & Chi [1], Keskar & Adidam [2], Sarhbas & Erbatur [3], Rhomberg & Street [4], Basudhar & Lakshman [5], Sivakumar & Munwar [6], and Yepes [7]. Although some mathematical programming based methods have been developed for optimum design problems, however, their applications are limited due to the fact that they require gradient information and usually seek to improve the solution in the neighborhood of a starting point. In recent years, structural optimization has witnessed the emergence of some novel and innovative design techniques. These stochastic search techniques make use of the ideas adopted from the nature, and do not suffer the discrepancies of mathematical programming based optimum design methods. The basic idea behind these techniques is to simulate the natural phenomena such as survival of the fittest, immune system, swarm intelligence and the cooling process of molten metals into a numerical algorithm. One of the recent additions to these techniques is the harmony search algorithm. This approach is based on the musical performance process that takes place when a musician searches for a better state of harmony. Jazz improvisation seeks musically pleasing harmony similar to the optimum design process which seeks optimum solutions. The pitch of each musical instrument determines the aesthetic quality, just as the objective function value is determined by the set of values assigned to each decision variable. In the process of musical production a musician selects and brings together number of different notes from the whole notes and then plays these with a musical instrument to find out whether it gives a pleasing harmony. The musician then tunes some of these notes to achieve a better harmony. Similarly it is then checked whether this candidate solution improves the objective function or not. This candidate solution is then checked to find out whether it satisfies the objective function or not, similar to the process of finding out whether euphonic music is obtained or not. Applications to structural design of structures can be found in the work of Saka [8,9], Saka & Erdal [10], Kaveh & Talatabahi [11], and Kaveh et al. [12].

* Corresponding Author: alikaveh@iust.ac.ir
1 School of Civil Engineering, Iran University of Science and Technology, Narmak, Tehran-16, Iran
2 Building and Housing Research Centre, Tehran-16, Iran
In this study, the harmony search and improved harmony search algorithms are used to determine the optimum design of reinforced concrete retaining walls. The objective function considered is taken as the cost of the structure, and design is based on ACI 318-05. This function is minimized subjected to design constraints. A numerical example is presented in order to illustrate the performance of the present algorithms and the necessary sensitivity analysis is performed.

2. Design variables of the problem

Fig. 1 shows the seven continuous design variables considered for modeling of the walls. These variables consist of the thickness of top stem (T1), the thickness of key and stem (T2), the toe width (T3), the heel width (T4), the height of top stem (T5), the footing thickness (T6), and the key depth (T7).

3. Optimum design process

Design of conventional retaining walls consists of two phases:
1. Check for stability
   • Check for overturning
   • Check for sliding
   • Check for bearing capacity failure
2. Checking of each component for strength and the steel reinforcement

The harmony search algorithm initiates the design process by selecting random values for the thickness of top stem (T1), the thickness of key and bottom stem (T2), the toe width (T3), the heel width (T4), the height of top stem (T5), the footing thickness (T6), and the key depth (T7). Then the algorithm checks the wall for stability and if these dimensions satisfy the stability criteria, the algorithm calculates the required reinforcement and checks the strength. Harmony Search algorithm tries to find the best value for each design variable to minimize the objective function. The design process consists of six steps as follow:

Step 1. Select the value of the harmony memory parameters (HM, HMCR, PARmin and PARmax).
Step 2. The harmony memory matrix is initialized (values for thickness of top stem (T1), thickness of key and bottom stem (T2), toe width (T3), heel width (T4), height of top stem (T5), footing thickness (T6) and key depth (T7) are chosen).
Step 3. Check whether the newly selected design vector should be pitch-adjusted.
Step 4. With the values selected for the T1, T2, T3, T4, T5, T6 and T7, the algorithm checks the wall for stability and if these dimensions satisfy the stability criteria, the algorithm calculates the required reinforcement and checks the strength.
Step 5. Calculate the objective function value for the newly selected design 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 magnitudes of the objective function.
Step 6. Repeat steps 2 and 6 until the pre-selected maximum number of iterations is reached.

The details of controlling the stability and the strength are given in the Appendix.

4. Objective function

By minimizing a suitable and explicit cost function, one can reach to an optimum solution for a concrete cantilever retaining wall. The optimal design of a concrete cantilever retaining wall is proposed to be determined by the minimum of the costs of concrete and steel reinforcement. The objective function can be expressed as follows:

\[ Q = V_{conc} \times (C_1 + C_2) + W_{steel} \times (C_3 + C_4) \]  

By considering \( Q = \frac{Q}{C_1 + C_2} \), we have:

\[ \tilde{Q} = \frac{Q}{(C_1 + C_2)} \]

Minimize \( \tilde{Q} = V_{conc} + W_{steel} \left( \frac{C_3 + C_4}{C_1 + C_2} \right) \)  

Subjected to:

\[ FS_{(overturning)} \geq 1.5 \quad \text{Ref. [13]} \]  
\[ FS_{(sliding)} \geq 1.5 \]  
\[ FS_{(bearing\ capacity)} \geq 2 \]  
\[ M_{ul}/(\phi_b M_n) \leq 1 \]  
\[ V_{ul}/(\phi_v V_n) \leq 1 \]

Where \( V_{conc} \) and \( W_{steel} \) are the volume of concrete and the weight of reinforcement steel in the unit of length (ft³/ft or m³/m), \( C_1 \) and \( C_2 \) are the cost of the concrete and steel ($/ft or $/kg), \( C_3 \) and \( C_4 \) are the cost of concreting and erecting reinforcement ($/ft or $/kg).

Experience show the value of \( \frac{C_3 + C_4}{C_1 + C_2} \) is in the range of 0.035 to 0.045.
5. Harmony search algorithm

The method consists of five basic steps. Detailed explanation of these steps can be found in the work of Geem & Kim [14], Lee & Geem [15,16], and Geem [17] which are summarized in the following:

Step 1. Harmony search parameters are initialized.
Step 2. Harmony memory matrix is initialized.
Step 3. New harmony memory matrix is improvised.
Step 4. Harmony memory matrix is updated.
Step 5. Steps 3 and 4 are repeated until the termination criterion is satisfied.

Step 1. A possible value range for each design variable of the optimum design problem is specified. A pool is constructed by collecting these values together from which the algorithm selects values for the design variables. Furthermore, the number of solution vectors in harmony memory (HMS) that is the size of the harmony memory matrix, the harmony considering rate (HMCR), the pitch adjusting rate (PAR) and the maximum number of searches are also selected in this step.

Step 2. Harmony memory matrix is initialized. Each row of the harmony memory matrix contains the values of design variables which are randomly selected feasible solutions from the design pool for that particular design variable. Hence, this matrix has $n$ columns, where $n$ is the total number of design variables and HMS rows which is selected in the first step. HMS is similar to the total number of individuals in the population matrix of the genetic algorithm. The harmony memory matrix has the following form:

$$[H] = \begin{bmatrix}
X_{1,1} & X_{2,1} & \cdots & X_{k,1} & X_{n,1} \\
X_{1,2} & X_{2,2} & \cdots & X_{k,2} & X_{n,2} \\
\vdots & \vdots & \ddots & \vdots & \vdots \\
X_{1,hms-1} & X_{2,hms-1} & \cdots & X_{k,hms-1} & X_{n,hms-1} \\
X_{1,hms} & X_{2,hms} & \cdots & X_{k,hms} & X_{n,hms}
\end{bmatrix} \quad (8)$$

Here $x_{ij}$ is the value of the ith design variable in the jth randomly selected feasible solution. These candidate designs are sorted such that the objective function value corresponding to the first solution vector is the minimum. In other words, the feasible solutions in the harmony memory matrix are sorted in descending order according to their objective function value. It is worthwhile to mention that not only the feasible designs satisfying the constraints are inserted into the harmony memory matrix.

Step 3. In generating a new harmony matrix, the new value of the ith design variable can be chosen from any discrete value within the range of ith column of the harmony memory matrix with the probability of HMCR which varies between 0 and 1. In other words, the new value of $x_i$ can be one of the discrete values of the vector $\{x_{i,1}, x_{i,2}, \ldots, x_{i,hms}\}$ with the probability of HMCR. The same is applied to all other design variables. In the random selection, the new value of the ith design variable can also be chosen randomly from the entire pool with the probability of $1 - \text{HMCR}$. That is

$$x_i^{\text{new}} = \begin{cases}
\{x_{i,1}, x_{i,2}, \ldots, x_{i,hms}\}^T & \text{with probability } \text{HMCR} \\
\{x_i, x_{i,2}, \ldots, x_{i,n}\}^T & \text{with probability } (1 - \text{HMCR})
\end{cases} \quad (9)$$

Where $n_s$ is the total number of the values for the design variables in the pool. If the new value of the design variable is selected from those of the harmony memory matrix, this value is then checked whether it should be pitch adjusted. This operation uses pitch adjustment parameter PAR that sets the rate of adjustment for the pitch chosen from the harmony memory matrix as follows:

$$\text{Is } x_i^{\text{new}} \text{ to be pitch-adjusted?} \begin{cases}
\text{Yes} & \text{with probability of } \text{PAR} \\
\text{No} & \text{with probability of } (1 - \text{PAR})
\end{cases} \quad (10)$$

Supposing that the new pitch-adjustment decision for $x_i^{\text{new}}$ come out to be yes from the test, and if the value selected for $x_i^{\text{new}}$ from the harmony memory is the kth element in the general discrete set, then the neighboring value $k + 1$ or $k - 1$ is taken for the new $x_i^{\text{new}}$. This operation prevents stagnation and improves the harmony memory for diversity with a greater change in reaching the global optimum.

Step 4. After selecting the new values for each design variable, the objective function value is calculated for the new harmony vector. If this value is better than the worst harmony vector in the harmony memory, 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. Steps 3 and 4 are repeated until the termination criterion which is the pre-selected maximum number of cycles is reached. This number is selected large enough such that within this number of design cycles no further improvement is observed in the objective function.

6. Improved harmony search algorithm

The HMCR and PAR parameters introduced in Step 3 help the algorithm to find globally and locally improved solutions, respectively.

PAR in HS algorithm is a very important parameter in fine-tuning of the optimized solution vectors, and can be potentially useful in adjusting the convergence rate of the algorithm to an optimal solution. Thus, fine adjustment of this parameter is of great importance. The traditional HS algorithm uses fixed value for PAR. In the HS method PAR value adjusted in initialization step (Step 1) and cannot be changed during new generations. The main drawback of this method appears in the number of iterations the algorithm needs to find an optimal solution. Small PAR values can result in poor performance of the algorithm and considerable increase in iterations needed to find optimum solution, Mahdavi et al. [18].

The key difference between the improved harmony search
(IHS) algorithm, developed by [18] and the traditional HS method is in the way of adjusting the PAR. In order to improve the performance of the HS algorithm and to eliminate the drawbacks which are associated with the fixed values of PAR, IHS algorithm uses variable PAR in the improvisation step (Step 3). PAR changes dynamically with the generation number as follow:

\[
PAR(g_n) = PAR_{\text{min}} + \frac{PAR_{\text{max}} - PAR_{\text{min}}}{NI} \times g_n
\]  

Where
PAR: the pitch adjusting rate for each generation;
PAR_{\text{min}}: the minimum pitch adjusting rate;
PAR_{\text{max}}: the maximum pitch adjusting rate;
NI: the number of solution vector generations;
g_n: the generation number.

7. A numerical example

The process of optimization is described in Section 3. For this purpose a computer program is written in Matlab for analysis, design and optimization. The analysis and design are in the form of a function which is called by the optimization program.

Two types of backfills are considered as defined in Table 1, and design is based on 1.0m wide strip of the retaining wall. Ground water level is assumed to be below the foundation level of the wall and therefore not affecting the soil properties. The total height of stem is constant and equal to 6.1m. Surcharge load is 10kN/m². The 28 days concrete cylinder strength is 25 MPa, Rebar yield stress is 300 MPa, and the allowable soil pressure is taken as q_s=300 kN/m² (3kg/cm²). The clear concrete cover is 50 mm. The hp is equal to zero. Upper and lower bounds for the design variable are shown in Table 2. A schematic view of a concrete retaining wall is illustrated in Fig. 2. The improved harmony search algorithm parameters are taken as:

HMS=30, HMCR=0.85, PAR_{\text{min}}=0.35 and PAR_{\text{max}}=0.99.

The design histories for two types of back files (F1 and F2) are shown in Fig. 3 and Fig. 4, respectively.

Numerical results reveal that the HIS can find better solutions compared to the standard HS.

8. Sensitivity analysis for the improved harmony search parameters

In this section, a sensitivity analysis is performed for the improved harmony search parameters involved in the present study. The results of the sensitivity analyses carried out to determine the appropriate values of the improved harmony search parameters are given in Table 4. The design histories are illustrated in Fig. 5.

<table>
<thead>
<tr>
<th>Type of back fill</th>
<th>Description</th>
<th>Density (kN/m³)</th>
<th>Internal friction angle (°)</th>
</tr>
</thead>
<tbody>
<tr>
<td>F1</td>
<td>Coarse granular fills (GW, GP)</td>
<td>22</td>
<td>35</td>
</tr>
<tr>
<td>F2</td>
<td>Granular soils with more than 12% of fines (GW, GS, SM, SL) and fine soils with more than 25% of coarse grains (CL–ML)</td>
<td>20</td>
<td>30</td>
</tr>
</tbody>
</table>

Table 2. Upper and lower bound for design variables

<table>
<thead>
<tr>
<th>Design variables</th>
<th>Thickness of the top stem</th>
<th>Thickness of the key and bottom stems</th>
<th>Toe width</th>
<th>Heel width</th>
<th>Height of the top stem</th>
<th>Footing thickness</th>
<th>Key depth</th>
</tr>
</thead>
<tbody>
<tr>
<td>Upper bound</td>
<td>0.3 m</td>
<td>0.3 m</td>
<td>0.45 m</td>
<td>1.8 m</td>
<td>1.5 m</td>
<td>0.3 m</td>
<td>0.2 m</td>
</tr>
<tr>
<td>Lower bound</td>
<td>0.6 m</td>
<td>0.6 m</td>
<td>1.2 m</td>
<td>3 m</td>
<td>6.1 m</td>
<td>0.9 m</td>
<td>0.9 m</td>
</tr>
</tbody>
</table>

Table 3. Optimum results for two types of back fills

<table>
<thead>
<tr>
<th>Type of fill</th>
<th>Thickness of top stem</th>
<th>Thickness of key and bottom stem</th>
<th>Toe width</th>
<th>Heel width</th>
<th>Height of top stem</th>
<th>Footing thickness</th>
<th>Key depth</th>
<th>As1</th>
<th>As2</th>
<th>As3</th>
<th>As4</th>
</tr>
</thead>
<tbody>
<tr>
<td>F1</td>
<td>33 cm cm</td>
<td>60 cm cm</td>
<td>120 cm cm</td>
<td>256 cm cm</td>
<td>325 cm cm</td>
<td>57 cm cm</td>
<td>67 cm cm</td>
<td>10.33 cm³</td>
<td>30 cm³</td>
<td>26.53 cm³</td>
<td>10.54 cm³</td>
</tr>
<tr>
<td>F2</td>
<td>34 cm cm</td>
<td>60 cm cm</td>
<td>117 cm cm</td>
<td>213 cm cm</td>
<td>333 cm cm</td>
<td>56 cm cm</td>
<td>35 cm cm</td>
<td>9.26 cm³</td>
<td>26.34 cm³</td>
<td>21.48 cm³</td>
<td>10.34 cm³</td>
</tr>
</tbody>
</table>
9. Concluding remarks

The optimization is performed by the recently developed improved harmony search method. This mathematically simple algorithm sets up harmony search matrix, each row of which consists of randomly selected feasible solutions to the design problem. In each search step, the algorithm searches the entire set rather than a local neighborhood of a current solution vector. It neither needs initial starting values for the design variables nor a population of candidate solutions to the design problem.

In this study, the improved harmony search developed in Ref. [18] is used, where the effects of this improvement on different mathematical functions and optimization problems are illustrated. IHS algorithm like harmony search algorithm is good at finding areas of the global optimum and is as good as mathematical techniques at fine-tuning within those areas. Numerical results reveal that the improved algorithm can find better solutions when compared to the standard HS. The results obtained show that the improved harmony search method is a powerful and an efficient method for finding the optimum solution of structural optimization problems. It is observed that the newly framed algorithm for optimal design of cantilever retaining wall is quite robust and efficient.

The proposed optimum design model enables structural designers to generate and evaluate optimal/near-optimal design solutions. This algorithm consists of the following three modules:

---

Table 4. Improved harmony search parameters used for the sensitivity analysis

<table>
<thead>
<tr>
<th>Case</th>
<th>HMS</th>
<th>HMCR</th>
<th>PARmin</th>
<th>PARmax</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>30</td>
<td>0.85</td>
<td>0.4</td>
<td>0.85</td>
</tr>
<tr>
<td>2</td>
<td>30</td>
<td>0.9</td>
<td>0.45</td>
<td>0.9</td>
</tr>
<tr>
<td>3</td>
<td>30</td>
<td>0.95</td>
<td>0.3</td>
<td>0.95</td>
</tr>
<tr>
<td>4</td>
<td>30</td>
<td>0.95</td>
<td>0.25</td>
<td>0.95</td>
</tr>
<tr>
<td>5</td>
<td>30</td>
<td>0.85</td>
<td>0.35</td>
<td>0.99</td>
</tr>
</tbody>
</table>

---

Fig. 2. Schematic view of a concrete retaining wall

Fig. 3. Design history for back fill F1

Fig. 4. Design history for back fill F2

Fig. 5. Design history for back fill F2 (cases 1, 2, 3, 4 and 5)
(1) A design module that performs the design of cantilever retaining wall;
(2) A cost module that computes the total cost of cantilever retaining wall;
(3) An optimization module that searches for optimal design alternatives.

The main aim of this paper has been to present a simple and efficient algorithm which can be employed in practical engineering problems. Such a simple approach can be utilized in many other engineering design problems to reduce the cost of the construction. Here, Harmony search method has been utilized; however, other meta-heuristics can also be applied to this problem. Examples of such methods can be found in [21-25].

**Notation**

- $tt$: Top stem thickness
- $tb$: Bottom stem & key thickness
- $HT$: Top stem height
- $HB$: Bottom stem height
- $LT$: Toe length
- $LH$: Heel length
- $L$: Total length of the base of the footing
- $ht$: Footing thickness
- $yb$: Density of the fill
- $\phi$: Internal friction angle of the fill
- $\beta$: Backfill slope
- $\mu$: Base friction coefficient
- $\gamma_c$: Density of the concrete
- $W_{w,t}$: Weight of the top stem
- $W_{w,b}$: Weight of the bottom stem
- $W_b$: Weight of the fill on the heel
- $W_s$: Surcharge weight
- $hk$: Key depth
- $hp$: Soil over toe
- $C_1$: Cost of the concrete
- $C_2$: Cost of the steel
- $C_3$: Cost of the concreting
- $C_4$: Cost of the erecting reinforcement
- $C_5$: Considered cases for sensitivity study
- $F_1$, $F_2$: Type of the back fills
- $T_1$, ..., $T_7$: The selected variables

**Acknowledgement:** The first author (Fellow of the Iranian Academy of Sciences) is grateful for the support.

**References**


[19] ACI Committee 318, Building Code Requirements for Structural Concrete (ACI 318-05) and Commentary (318R-05), American Concrete Institute, Farmington Hills, Mich., 430 p, 2005.


Appendix: Analysis and design of concrete cantilever retaining wall

The content of this section is based on Ref. [13] and Refs. [19-20].

A1. Active and passive earth pressure coefficients

The active and passive earth pressure coefficients are computed using the Coulomb earth pressure theory. The details of the Coulomb earth pressure are shown in Fig. A1.

\[
K_A = \frac{\sin^2(\alpha + \phi)}{\sin^2(\alpha - \delta) \sin(\alpha - \delta) [1 + \left( \frac{\sin(\phi + \delta) \sin(\phi - \beta)}{\sin(\alpha - \delta) \sin(\alpha + \beta)} \right)^2]}
\]

\[
K_P = \frac{\sin^2(\alpha - \phi)}{\sin^2(\alpha - \delta) \sin(\alpha - \delta) [1 + \left( \frac{\sin(\phi + \delta) \sin(\phi + \beta)}{\sin(\alpha - \delta) \sin(\alpha + \beta)} \right)^2]}
\]

A2. Stability control

All the loads acting on the cantilever retaining wall are shown in Fig. A2.

Check for overturning:

\[
FS_{overturning} = \frac{\Sigma M_R}{\Sigma M_O}
\]

Where

\[
\Sigma M_O = \text{sum of the moments of forces that tends to overturn about point C}
\]

\[
\Sigma M_R = \text{sum of the moments of forces that tends to resist overturning about C}
\]

\[
FS_{overturning} = \frac{\Sigma Wx + M_{HP}}{\Sigma Hy}
\]

Check for sliding along the base:

\[
FS_{sliding} = \frac{F_x}{F_d}
\]

Where

\[
\Sigma F_R = \text{sum of the horizontal resisting forces}
\]

\[
\Sigma F_d = \text{sum of the horizontal driving forces}
\]

\[
FS_{sliding} = \frac{Hb + Hs}{Hp + \mu \Sigma W}
\]

Check for bearing capacity failure:

\[
FS_{bearing\ capacity} = \frac{q_u}{q_{max}}
\]

Where:

\[
q_u = \text{Ultimate bearing capacity}
\]

\[
q_{max} = \text{Maximum bearing pressure}
\]

\[
q_{max} = \begin{cases} \frac{\Sigma W(1 + 6e)}{L}, & \text{for } e \leq \frac{L}{6} \\ \frac{BL}{2 \Sigma W}, & \text{for } e > \frac{L}{6} \end{cases}
\]

\[
e = \frac{L}{2} - \frac{\Sigma Wx - \Sigma Hy - M_{HP}}{\Sigma W}
\]

A3. Check for the strength

Checks the stem flexure capacity ( is the load factor):

\[
M_u/\phi_p M_h \leq l
\]

\[
M_u = \begin{cases} 1.6 \left( \frac{Pa \times HT^4}{6} + \frac{Pa \times HT^2 \times w}{2\gamma_b} \right), & \text{for top stem} \\ 1.6 \left( \frac{Pa \times (HT + HB)^4}{6} + \frac{Pa \times (HT + HB)^2 \times W}{2\gamma_b} \right), & \text{for bottom stem} \end{cases}
\]

\[
P_u = \begin{cases} 1.2 W_{wa}, & \text{for top stem} \\ 1.2(W_{wa} + W_{wb}), & \text{for bottom stem} \end{cases}
\]

Check the heel flexure capacity:

The soil pressure under footing is shown in Fig. A3

\[
e_a = \frac{L}{2} - \frac{\Sigma \gamma W \times x - \Sigma \gamma H \times y \times MHP}{\Sigma \gamma W}
\]
Check the toe flexure capacity:

\[ S = \frac{\sum \gamma W}{0.5 q_{u, toe}} - LT + tb \]  
(A-15)

\[ M_{u,3} = \begin{cases} 
\frac{LH}{2} \left( \frac{w_s + \gamma w_b}{L} \right) \frac{q_{u,3} + 2q_{u,heel}L}{6} \times L, & \text{for } e_u \leq \frac{L}{6} \\
\frac{LH}{2} \left( \frac{w_s + \gamma w_b + \gamma w_f}{L} \right) \frac{q_{u,3}b}{6}, & \text{for } e_u > \frac{L}{6} 
\end{cases} \]  
(A-16)

\[ \rho_{req,3} = \frac{0.85 f' c \left( 1 - \frac{M_{u,3}}{0.383 b d^2 f_c} \right)}{f_y} \]  
(A-17)

Check the shear capacity for the stem:

\[ \rho_{req,4} = \frac{0.85 f' c \left( 1 - \frac{M_{u,4}}{0.383 b d^2 f_c} \right)}{f_y} \]  
(A-19)

Check the shear capacity for the toe:

\[ V_{\text{allowable}} = 2\phi bd \sqrt{f_c} \]  
(A-20)

Check the shear capacity for the heel:

\[ V = \gamma \left( \frac{P_s}{2} + \frac{w, P_s, HT^2}{\gamma_b} \right), \text{for top stem} \]  
(A-21)

\[ V = \left[ \gamma \left( \frac{P_s, (HT + HB)^2}{2} + \frac{w, P_s, (HT + HB)^2}{\gamma_b} \right) \right], \text{for bottom stem} \]  
(A-22)

\[ V = 0.5 (q_{u, toe} + q_{u, heel}) LT \]  
(A-24)

\[ V_{\text{allowable}} = 2\phi bd \sqrt{f_c} \]  
(A-25)