Trends in

Applied Sciences Research

ISSN 1819-3579

Academic
Journals Inc.

www.academicjournals.com

Trends in Applied Sciences Research 6 (4): 375-385, 2011
ISSN 1819-3579 / DOI: 10.3923/tasr.2011.375.385

Searching for the Optimal Design Variables of RC Flat Slab by the Reactive Taboo

Adeola A. Adedeji
Department of Civil Engineering, University of Ilorin, PMB 1515, Ilorin, Nigeria

Key words: Optimum design, Slab, algorithms, reinforced concrete

INTRODUCTION

Design optimization of a Reinforced Concrete (RC) flat slab is addressed in this work using Reactive Taboo Search (RTS). Optimization problems were formulated using design criteria by the British Standards, BS8110, Part I (1997). A modification of reactive taboo search that uses a population-based exploitation of the search history is applied for the design of reinforced concrete flat slab. The flat slab was computed with the aid of a visual basic program, giving the best fit dimensions that met all necessary ultimate and serviceability requirements. Average dimensions obtained for the optimisation is 6200×1000×153 mm and this was compared with the conventional Genetic Algorithms (GAs) of the same structural dimension. The results of this work indicate that RTS is cost effective when compared with the GAs in terms of finding new better optima.

Flat slabs system of construction is one in which the beams used in the conventional methods of construction are done away with. The slab directly rests on the column and load from the slab is directly transferred to the columns and then to the foundation. To support heavy loads the thickness of slab near the support with the column is increased and these are drops, or columns are generally provided with enlarged heads or capitals. Absence of beam gives a plain ceiling, thus giving better architectural appearance and also less vulnerability in case of fire than in usual cases where beams are used. Plain ceiling diffuses light better, easier to construct and requires cheaper form work. In analyzing, detailing and designing of structural elements (of which slab is one), assumptions are made for member sizing (dimensions). This is followed by the analysis of the whole structure and then members' design. The process is always lengthy since it requires so many assumptions that may not be concise at the end of the design. This may call for a re-design of such element and a re-designing process comes into play if the later is not adequate. This process becomes so ambiguous that it takes a considerable amount of time and effort. In order to reduce the process of re-design to the barest minimum and also to minimize cost, optimization methods have been considerable and also led to saving (in terms of time, effort and cost), both in construction materials and in design process. Goldberg and Samtani (1986) appear to have first suggested the use of GA for structural optimization, using continuous variables. However, in this work, the optimum design of flat slab is sought using Reactive Taboo Search. In addition, structural optimization problems are known to have many local optima, which encourage the use of heuristic global optimizers. Some of these optimizers include: Genetic Algorithm (GA), Evolutionary (EA), Reactive taboo search (RTS). Although, RTS has less capability to span the search space than GA, it is capable of efficiently nailing down the local optima by escaping them to find new ones. This work proposes a modification of the escape mechanisms of RTS to further increase its effectiveness at finding new better optima. The obtained results show improved performance when implementing the proposed modifications. In other words this work is concerned mainly about the application of RTS in optimizing the design of flat slab by using a new evolving method of optimization

RC FLAT SLABS SIMPLIFIED ANALYSIS

A reinforced concrete slab as structural element, which more often than not is subjected to bending (tensile or compressive) and in rare cases (such as in bridge decks) could be subjected to shear. In most cases, slabs are horizontal members, but they can be used as vertical members such as walls to infill panels, side walls to drains and sewer appurtenances, etc. Thus, its design centers more on flexure rather than direct shear. Where slabs are thicker than 200 mm or where action of point or live load is predominant, slabs are checked for direct shear and punching shear.

Concrete slabs behave primarily as flexural members and the design may be similar to that of beams, except that it is somewhat simpler because: 1) a width of 1000 mm is generally assumed for design simplicity, 2) the section is usually rectangular and hence the flanges, 3) shear is generally not considered unless where concentrated or live loads predominate and the slab is thicker than 200 mm and design considered for singly reinforced section, that is, compression reinforcement is hardly used. British Standards BS8110 Part1 (1997) permits the use of simplified load arrangement for all slabs of maximum ultimate design load throughout all spans or panels provided that the following conditions are met: (1) In a one way slab, the area of each bay 430 m^2 (2) Live load $Q_k \le 1.25$ dead G_k and (3) Live load $Q_k \le 5$ kN m^{-2} excluding partitions. If an analysis is based on this single load case, all support moments (except at a cantilever) would be reduced by 20% and span moments increased accordingly.

DESIGN CONSIDERATIONS

Shear in reinforced concrete slabs: The shear stress at section in a solid slab is computed mathematically as:

$$v = \frac{V}{bd} \tag{1}$$

where, V is shear force due to the ultimate load, d is effective depth of the slab and b is width of section considered. Calculations are usually based on a strip of slab 1m wide. The BS 8110 code requires that for a solid slab (1) v ≤ the lesser of $0.8\sqrt{f_{cu}}$ or 5 N mm^{-2}, (2) $v \le V_c$ for a slab thickness less than 200 mm and (3) If $v \le V_c$, the shear reinforcement must be provided in slabs more than 200 mm thick. If shear reinforcement is required, than nominal steels as for beams should be provided when $v<(V_c+0.4)$ and designed reinforcement provided for higher values of v.

Span-effective depth ratio: Excessive deflection of slabs will cause damage to ceiling, floor finishes and other architectural details. To avoid this, limits are set on the span-depth ratio. These limits are exactly the same as those for beams. As a slab is usually a slender member, the restrictions on the span-depth ratio become more important and this can often control the depth of slab required. In terms of span-effective depth ratio, the depth of the slab is given by:

$$\text{Minimum effective depth} = \frac{\text{Span}}{\text{Basic ratio} \times F_m} \tag{2}$$

The modification factor (F_m) is based on the area of tension steel in the shorter span when a slab is singly reinforced at mid-span, but if a slab has both top and bottom steel at mid-span, the modification factors for the areas of tension and compression steel are used. Although, a slab should be designed to withstand unfavourable arrangements of design loads, a single load case of maximum design load will suffice provided the following conditions are met:

- In a one-way spanning slab, the area of each bay exceeds 30.0 m^2
- Theratio of the characteristic imposed load to the characteristic dead load does not exceed 1.25
- The characteristic imposed load does not exceed 5 kN m^{-2} excluding partitions

The shear stress of the slab should be calculated from v = V/bd and compared with d when v <V_c, no shear reinforcement is required and in which d = deflection of the slab due to load. When V_c<V<(V_c+0.4); minimum shear reinforcement is required in areas where v>V_c and the spacing of such links is calculated from:

$$s_v = 0.95\ (f_{yv} A_{sv}/0.4b) \tag{3}$$

where, (V_c + 0.4) < V <$0.8vf_{cu}$ or 5 N mm^{-2}, links and/or bent-up bars in any combination (but the spacing between links or bent-up bars need not be less than d) is needed; then the spacing is calculated (for beams) from:

$$s_v = 0.95\ (F_{yv} A_{sv}/b\ (V\text{-}V_c)) \tag{4}$$

Slab links should be avoided in slabs less than 200 mm and bent-up bars avoided in slabs as much as possible. Details are shown in sections 3.7.6.2 to 3.7.7.5. of British Standards BS8110, 1997. In practical cases, deflection in slabs is controlled when the span-effective depth ratio given below and modified by the modification factor for tension reinforcement given in the code is adhered to. The slab effective depth ratios for: (1) cantilever slab = 7, (2) Simply supported slab = 20 and (3) Continuous slab = 26.

GAS, EA AND RTS

GA is an optimization technique invented by John Holland at University of Michigan in the United States in 1975 (Goldberg, 1989). He introduced GA to mimic some of the processes of natural evolution and selection. It is an iterative that involves reproduction, crossover and mutation. Some works have been introduced recently by the modifying mutation in GA and genetic programme (GP) (Adedeji, 2008, 2009) The entropy-based searching techniques with a multi-population and the quasi-exactness penalty function (Wang and Jiye, 2002; Jekayinfa, 2006) has been developed to ensure rapid and steady convergence results. The two categories are (1) GA_1 implements variable storage as integer variables, four crossover operators, 12 mutation operators, fitness scaling, population distribution, roulette wheel selection along with elitist selection. Using an integer storage, the efficiency of storage of variables, are stored directly as integers rather than binary strings and are translated to their equivalent binary strings when needed during some of the crossover and mutation operators and (2) GA_2 is the same as GA_1, but all evaluations of objective function are stored or it could have all, for its members, the initial population normally distributed about the initial feasible design. Objective function caching is inherent in RTS and is one of the strong points in favor of it. In topology optimization, the connectivity is the objective of the optimization (Bendose and Kikuchi, 1988). Therefore, history storage is implemented into GA in order to even up the advantage RTS has and allow for a better comparison.

EA is a search process. It searches a very large space of genotypes producing structures that are efficient at carrying out functions desirable for survival in their environment. EA consists of two operators and it depends on the two operators for effective performance. They are crossover and mutation. EA for designs can be considered as manipulations. It manipulates low-level tools playing with their arrangements until it finds the required variables. Encoding consists of finding a set of low-level tools for EA to manipulate.

Reactive taboo search and its reaction to search behaviour: Reactive taboo search is a heuristic global optimization technique that has less stochastic content than genetic algorithm. In fact, save for a small portion of the algorithm, it is almost completely deterministic. The basic idea in taboo search is to make use of previously evaluated points within the search space to direct the future sampling and prevent entrapment at a local minimum by applying taboo conditions (Glover, 1989, 1990). RTS proposes a scheme for adaptively varying the way the taboo conditions are applied based on the objective function history, thus the search reacts to the objective function behaviour. At each move (iteration), RTS places a taboo condition on the previous point to prevent moving into it again until some other moves are completed. The taboo condition lasts a number of iterations equal to the current taboo list length. RTS also keeps track of when each point was visited and the number of visits. If a point is visited twice, the taboo list length is increased. Thus, near a local minimum, the taboo list length keeps increasing until it is enough to explore regions further away. If a number of iterations pass without any cycles occurring (visiting the same point several times), the taboo list length is decreased. Typically, a maximum taboo list length is specified. It is generally not beneficial to have the maximum taboo list length greater than the number of points in the neighborhood, because it can lead to a situation when all the points in the neighborhood are tabooed. When such a situation arises, the taboo conditions are relaxed and the new current point is chosen as the last visited point in the neighborhood. Thus, the tabooing does not always prevent cycling back into the domain of attraction of a local optimum, which occasionally calls for executing the quick escape mechanism.

Modified quick escape mechanism: Quick escape is performed by randomly changing the values of some of the variables of the current point. It is simply like re-starting the search at new starting point, which may or may not have been visited before, by mutating the current design. The main reason for using an escape mechanism in RTS is to escape large basins of attraction and to diversify the search into unexplored regions of the search space. The escape mechanism described was proposed by Battiti and Tecchiolli (1994) and Caldas and Norford (2003) can be viewed as a mutation of the local optimum point. Such mechanism introduces an element of randomness into RTS, which is otherwise completely deterministic. RTS performs a complete neighborhood evaluation. Unlike the version of RTS proposed by Battiti and Tecchiolli (1994) where all variables were in binary, the implemented version in this work uses integer values for the variables. The neighborhood is defined as the set of points that have all their variables equal to those of the current point except for one variable, which is different by a value of ±1. Thus, the number of points in the neighborhood is twice the number of variables (or less for points touching the upper and lower limits of the variable ranges). The modified escape mechanism follows the following steps:

- By forming a population using some of the already explored points
- By evolving the population in a manner encouraging an up-hill climb
- To Select one point of the up-hill population that is as far as possible from the mean of the explored space (analogous to speciation in GA)
- By performing the same sort of mutation as in normal escape mechanism on the selected point (instead of on the local optimum)

The underlying assumptions that justify the use of the modified escape mechanism are as follows:

- The new optima to be found are pretty far from the currently found ones, so moving away from the found local optima before making the jump is better than making the random jump from the local optima themselves
- The actual objective function evaluation is expensive in terms of computational time, so the population search among the previously explored points (which does not require new objective function evaluations) is not too much of an added computational expense on the overall search process whose dominant computational expense is the evaluation of new points
- Mixing both normal and modified escape mechanisms is also an interesting possibility to make use of the advantages of both. The mixed escape mechanism simply uses the normal escape mechanism to perform the first escape, then the modified mechanism in the following escape, then the normal for the next escape and keeps alternating the escape mechanism

METHOD OF STUDY

The method adopted in this project is the application of Reactive Taboo Search in optimizing design of the flat slab. This method is adopted because it has less stochastic content compared to other evolutionary search methods. The basic idea in RTS is to make use of previously evaluated points within the search space to direct the future sampling and prevent entrapment at a local minimum by applying taboo conditions. RTS proposes a scheme for adaptively varying the way the taboo conditions are applied based on the objective function (OF) history, thus the search reacts to the OF behaviour.

Test: A single starting point in the search space is set as the current point. RTS then evaluates the entire neighborhood of the current point and moves to the best point in it and then becomes the new current point. An important feature in RTS is that all the previously evaluated points are stored in the memory, this leads to lots of savings in computational time when evaluating the neighborhood of the new point. At the start of the search RTS, simply behaves like a steepest descent search until it hits a local minimum. Whereas steepest descent stops upon reaching a local minimum, RTS continues to search the neighborhood of the current point and move to best point within it even if it is worse than the current point. To prevent infinite cycling back and forth around a local minimum, TS imposes a taboo condition upon the last visited point, that is, a previously visited point cannot be visited again until a certain number of iterations is completed and such number of iterations is typically referred to as the taboo list length. In RTS, the taboo list length is adaptively changed according to the search behavior within a minimum and a maximum value. If the search still gets stuck in a large basin of attraction of the objective function, which the maximum taboo list length is not enough to overcome; a quick escape is performed. The search is typically stopped after performing a specified number of moves or objective function evaluations. The best point encountered is returned.

Problem formulation

Fitness (cost) function: An important step in the proper formulation of the problem is identifying the design variables for the structure. In the case of optimization of trusses there can be three main categories i) sizing, ii) configuration and iii) topology (Adedeji and Ejeh, 2006; Adedeji, 2007) . This classification is slightly different from that of continuum structures, given in (Chapman *et al.*, 1993). In this work, the only design variable considered is a one-way flat slab of thickness (h), width (b) = 1.0 m and length (L) = 6.2 m. To minimize the weight of concrete at a density of 24 kN m^{-3}: involved in the flat slab by minimizing the slab thickness, the fitness (objective) function is formulated as:

$$Z_{\text{fitness function}} = \text{Total weight of slab} = W_t \tag{5}$$

Constraints: Constraints evaluation is the main costly event when one considers time of the use of CPU. It involves solving for different load cases then performing safety checks on slab members. Constraints were formulated as:

- Safety Constraints:
  - Qk = 1.25 Gk
  - Qk = 5 kN m^{-2} excluding partitions
  - Flat slab panels should be assumed to be divided into strips and middle strips
  - In a one-way slab, the area of each bay = 30 m^{-2}

- Design constraints:
  Punching shear constraint

$$\sum A_{sv} \text{ Sin } \alpha \le \frac{0.4 \text{ ud}}{0.95 \text{ f}_{yv}} \tag{6}$$

Where:

u = Perimeter of drop panel
d = Slab effective depth
f_{yv} = Steel yield strength
V = Shear force

In which the shear force:

$$V = F - (\frac{\pi}{4} h_c^2 n) \tag{7}$$

where, F = Force due to load, n = F/L^2. Since, h_c = 1.42 m, n = F/L^2 and p = 3.142, Thus;

$$V = F\ (1\text{-}0.037) = 0.963\ F \tag{8}$$

Shear stress:

$$V = \frac{1.15\ V}{ud} \leq V_c \tag{9a}$$

Thus,

$$V = \frac{1.1075\ V}{ud} \leq V_c \tag{9b}$$

At u = h_c = 3.142×1.42 = 4.46164; d = h-c-D/2 = h-26

Where, D = Steel diameter, so that:

$$\frac{1.1075\ V}{4.4616\ (h-26)} \leq 0 \tag{10}$$

Constraints of the cross section

h > 0; b > 0 and 1.2 < h < 3.00 (m)

RTS PSEUDOCODE

```
Step 1: Begin at a starting point
Step 2: Examine non-tabooed neighboring points and move to the best of them
Step 3: If new point has not been visited before
Step 4: Goto 2
Step 5: Else-If cycling is not excessive
Step 6: Put a taboo condition upon point
Step 7: Goto 2
Step 8: Else perform quick escape and Goto 2
```

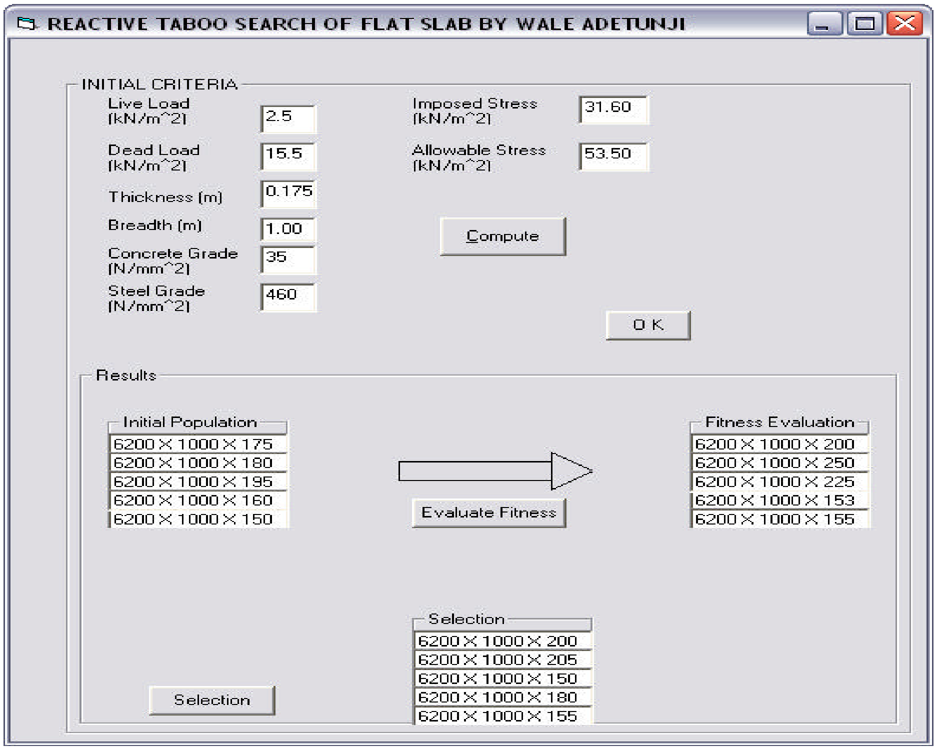
TEST EXAMPLE

A flat slab data is shown in Table 1, while the cost function (Z) considers the unit cost of concrete per m^3 (C_c) and the weight (W_t). Thus, at the value of naira (₦), C_c = N9,310.00 and W_t = 922.56 kN (₦ 150 equivalent = US$1, 2010):

$$Z = C_c\ W_t = 8589033.6\ h \tag{11}$$

Table 1: The slab design data

Parameters	Values
Slab thickness	$0.12\ m<h<0.3\ m$
Effective span (L)	6.20 m
Slab Width (b)	1.0 m
Cross-sectional area (A)	$b \times h$
Concrete self-weight (ω)	24 kN m^{-3}
Slab total weight (W_t)	$\rho\ h\ L^2$
Live load (Q_k)	2.5 k N m^{-2}
Dead load (G_k)	(W_t + 5.1) kN m^{-2}
Ultimate load (F)	1.4 G_k+1.6 Q_k
Equivalent distributed load (n)	F/L^2
Maximum moment (M)	0.063 fL
Maximum deflection	$M/bd^2<1.0$ mm
Concrete strength (F_{cu})	30 N mm^{-2}
Steel strength (F_y)	460 N mm^{-2}
Column diameter (h_c)	1.42 m
Steel diameter (Φ)	12 mm



REACTIVE TABOO SEARCH OF FLAT SLAB BY WALE ADETUNJI

INITIAL CRITERIA

Live Load [kN/m^2] 2.5

Dead Load [kN/m^2] 15.5

Thickness [m] 0.175

Breadth [m] 1.00

Concrete Grade [N/mm^2] 35

Steel Grade [N/mm^2] 460

Imposed Stress [kN/m^2] 31.60

Allowable Stress [kN/m^2] 53.50

Results

Initial Population

6200 × 1000 × 175
6200 × 1000 × 180
6200 × 1000 × 195
6200 × 1000 × 160
6200 × 1000 × 150

Fitness Evaluation

6200 × 1000 × 200
6200 × 1000 × 250
6200 × 1000 × 225
6200 × 1000 × 153
6200 × 1000 × 155

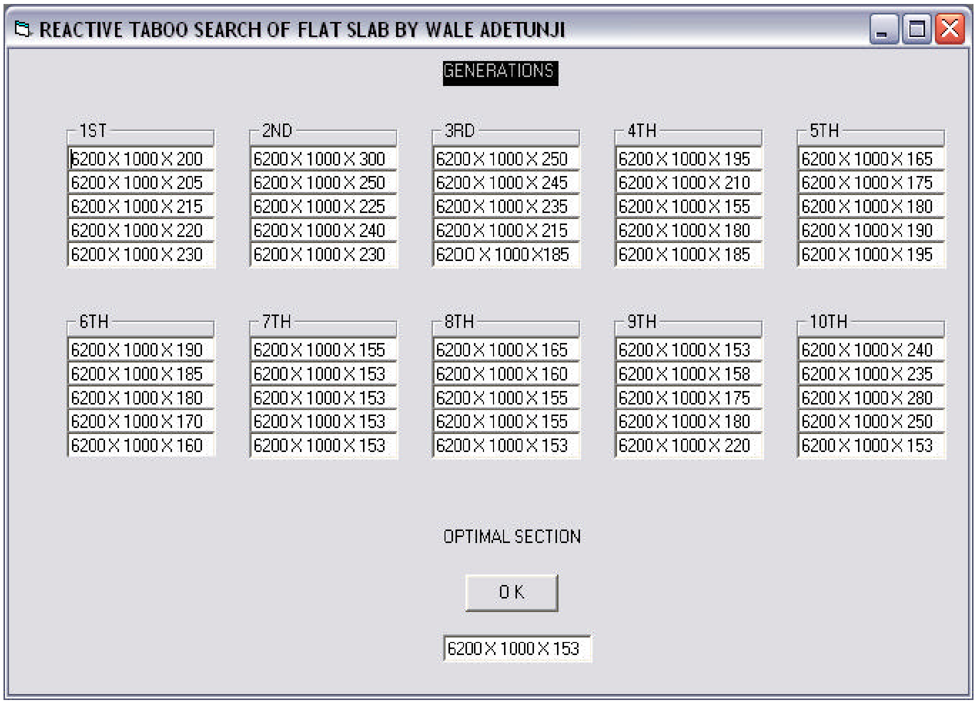
Selection

6200 × 1000 × 200
6200 × 1000 × 205
6200 × 1000 × 150
6200 × 1000 × 180
6200 × 1000 × 155

Fig. 1: Selection of slab parameters

OPTIMAL DESIGN RESULTS

The visual basic interface for the program presented as indicated in Fig. 1 showing the selection and input of the slab parameters, while in Fig. 2, 10 generations were indicated. Shown in Fig. 3



REACTIVE TABOO SEARCH OF FLAT SLAB BY WALE ADETUNJI

GENERATIONS

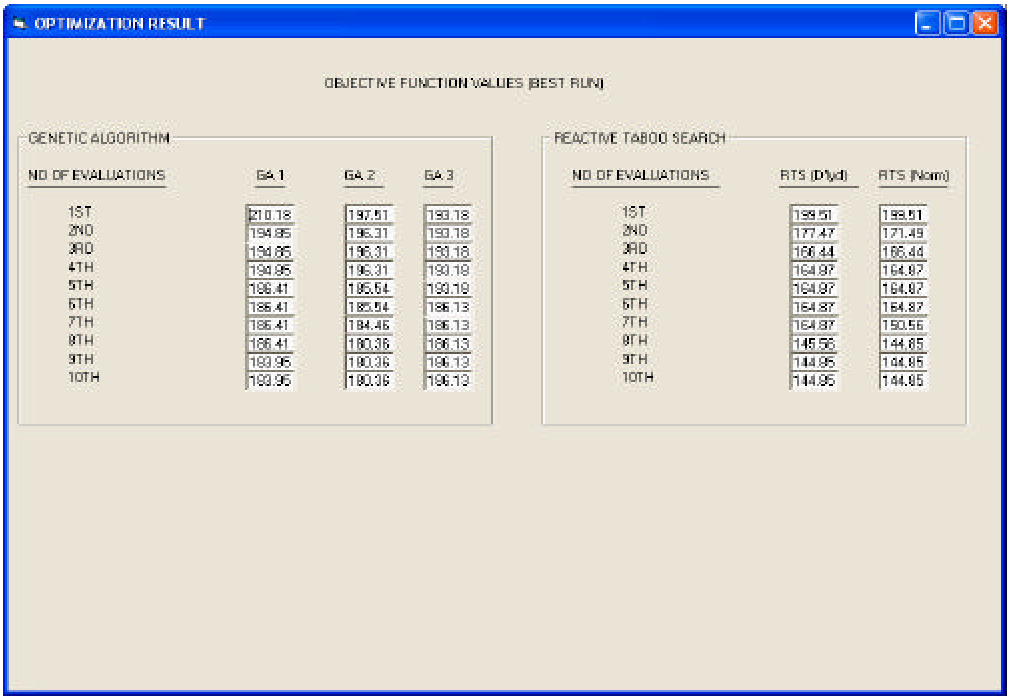
1ST	2ND	3RD	4TH	5TH
6200 X 1000 X 200	6200 X 1000 X 300	6200 X 1000 X 250	6200 X 1000 X 195	6200 X 1000 X 165
6200 X 1000 X 205	6200 X 1000 X 250	6200 X 1000 X 245	6200 X 1000 X 210	6200 X 1000 X 175
6200 X 1000 X 215	6200 X 1000 X 225	6200 X 1000 X 235	6200 X 1000 X 155	6200 X 1000 X 180
6200 X 1000 X 220	6200 X 1000 X 240	6200 X 1000 X 215	6200 X 1000 X 180	6200 X 1000 X 190
6200 X 1000 X 230	6200 X 1000 X 230	6200 X 1000 X185	6200 X 1000 X 185	6200 X 1000 X 195

6TH	7TH	8TH	9TH	10TH
6200 X 1000 X 190	6200 X 1000 X 155	6200 X 1000 X 165	6200 X 1000 X 153	6200 X 1000 X 240
6200 X 1000 X 185	6200 X 1000 X 153	6200 X 1000 X 160	6200 X 1000 X 158	6200 X 1000 X 235
6200 X 1000 X 180	6200 X 1000 X 153	6200 X 1000 X 155	6200 X 1000 X 175	6200 X 1000 X 280
6200 X 1000 X 170	6200 X 1000 X 153	6200 X 1000 X 155	6200 X 1000 X 180	6200 X 1000 X 250
6200 X 1000 X 160	6200 X 1000 X 153	6200 X 1000 X 153	6200 X 1000 X 220	6200 X 1000 X 153

OPTIMAL SECTION

6200 X 1000 X 153

Fig. 2: Generations and the optimal values



OPTIMIZATION RESULT

OBJECTIVE FUNCTION VALUES [BEST RUN]

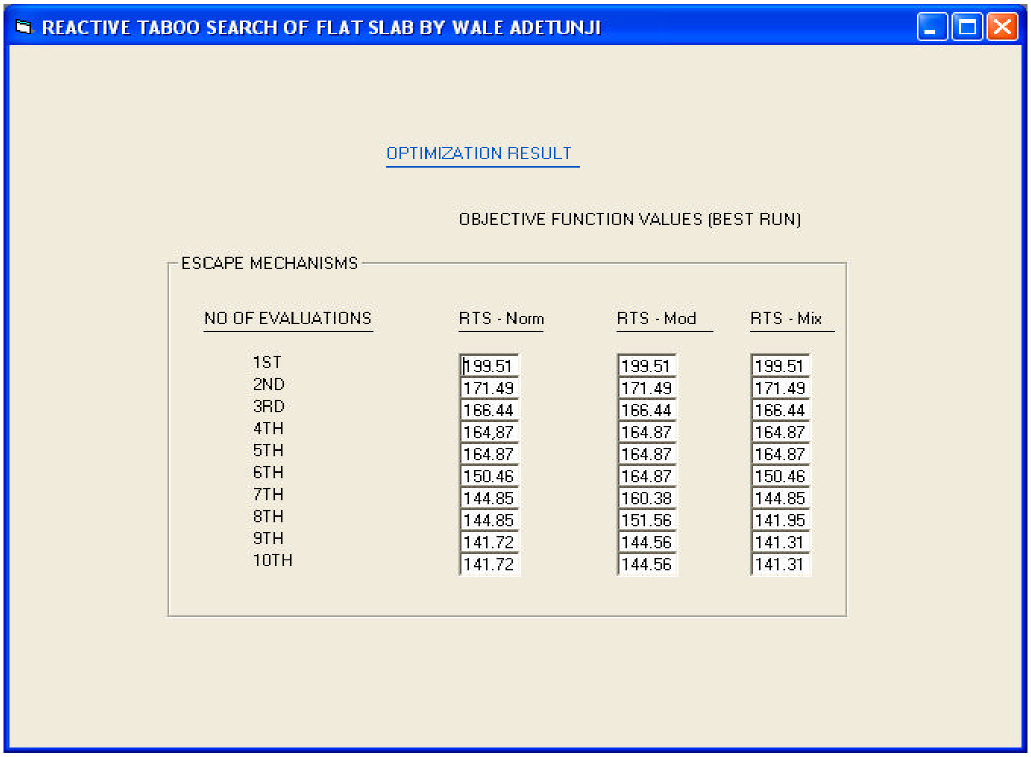
GENETIC ALGORITHM

NO OF EVALUATIONS	GA 1	GA 2	GA 3
1ST	210.18	197.51	193.18
2ND	194.85	196.31	193.18
3RD	194.85	196.31	193.18
4TH	194.95	196.31	193.18
5TH	196.41	185.54	193.18
6TH	186.41	185.54	186.13
7TH	186.41	184.46	186.13
8TH	186.41	180.36	186.13
9TH	193.95	180.36	196.13
10TH	193.95	180.36	196.13

REACTIVE TABOO SEARCH

NO OF EVALUATIONS	RTS (D'yd)	RTS (Norm)
1ST	199.51	199.51
2ND	177.47	171.49
3RD	166.44	166.44
4TH	164.97	164.87
5TH	164.97	164.87
6TH	164.87	164.87
7TH	164.87	150.56
8TH	145.56	144.85
9TH	144.95	144.85
10TH	144.95	144.85

Fig. 3: Optimal values comparison between GAs and RTS



REACTIVE TABOO SEARCH OF FLAT SLAB BY WALE ADETUNJI

OPTIMIZATION RESULT

OBJECTIVE FUNCTION VALUES (BEST RUN)

ESCAPE MECHANISMS

NO OF EVALUATIONS	RTS - Norm	RTS - Mod	RTS - Mix
1ST	199.51	199.51	199.51
2ND	171.49	171.49	171.49
3RD	166.44	166.44	166.44
4TH	164.87	164.87	164.87
5TH	164.87	164.87	164.87
6TH	150.46	164.87	150.46
7TH	144.85	160.38	144.85
8TH	144.85	151.56	141.95
9TH	141.72	144.56	141.31
10TH	141.72	144.56	141.31

Fig. 4: Optimal values with different RTS modes

are the optimal values comparison between GAs and RTS and in Fig. 4, the interface shows the optimal values with different RTS modes.

CONCLUSION

The model used in this work takes into account most of the practical aspects for design applicability. While, utilizing a number of objective function evaluations, which is only a small fraction of the total search space, RTS succeeded in coming up with better designs. RTS, though having less diversification capability, performed better than other search methods, which motivated a modification to the mechanism RTS uses to escape large attraction basins in the objective function. Since the population search is done on pre-computed points, it has no additional cost in terms of objective function evaluation. The performed study shows that alternating between the normal and modified escape mechanism gives an overall better performance. Average dimensions obtained for the optimization is 6200×1000×153 mm and compared with the conventional GAs of the same dimensions. The result shows that RTS is cost effective.

ACKNOWLEDGMENT

I wish to thank Mr. Wale Adetunji, of the Department of Civil Engineering, University of Ilorin, who worked with the author to programme and key-in the data for computations of the results.

REFERENCES

Adedeji, A.A. and S.P. Ejeh, 2006. Simplifying design of truss system using genetic algorithms J. Res. Inform. Civil Eng., 3: 33-54.

Adedeji, A.A., 2007. Genetic (Evolutionary) Algorithm: Introduction and Its Use as an Engineering Design Tool. Olad Publication and Printing Enterprises, Nigeria, ISBN: 978-8115-86-1.

Adedeji, A.A., 2008. Optimising design and maintenance of strawbale masonry prism by genetic algorithm. http://www.unilorin.edu.ng/unilorin/publications/adedeji/ Optimising%20design%20and%20maintenance%20of%20strawbale%20masonry%20prism%20by%20genetic%20algorithm.pdf.

Adedeji, A.A., 2009. Forget it and its application in engineering design. http://www.researchgate.net/profile/Adeola_A_Adedeji/blog/.

Battiti, R. and G. Tecchiolli, 1994. The reactive taboo search. ORSA J. Comput., 6: 126-140.

Bendose, M. and N. Kikuchi, 1988. Generating optimal topologies in structural design using a homogenization method. Comput. Method Appl., Mech. Eng., 7: 197-224.

British Standards, BS8110 Part I, 1997. Flat Slab, Sect. 3.7. BSI., London.

Caldas, L.G. and L.K. Norford, 2003. Genetic algorithms for optimization of building envelopes and design and control of HVAC system. J. Solar Energy Environ., 125: 343-350.

Chapman, C., K. Saitou and M. Jakiela, 1993. Genetic algorithms as an approach to configuration and topology design. Adv. Des. Automaion, 65: 485-488.

Glover, F., 1989. Taboo search-Part I. ORSA J. Comput., 1: 190-206.

Glover, F., 1990. Taboo search-Part II. ORSA J. Comput., 2: 4-32.

Goldberg, D.E. and M. Samtani, 1986. Engineering optimization via genetic algorithms. Proceedings of the 9th Conference on Electronic Computations, (CEC'86), ASCE, Birmingham, pp: 471-482.

Goldberg, D.E., 1989. Genetic Algorithms in Search Optimization and Machine Learning. Addison-Wesley, New York.

Jekayinfa, S.O., 2006. Energy consumption pattern of selected mechanical farms in South-Western Nigeria. CIGRE J., 3: 23-28.

Wang, X. and T. Jiye, 2002. Parallel multi-frontal algorithm for structural shakedown analysis. J. North-Easten Univ., 23: 115-117.