Galactic Swarm Optimization Applied to Reinforcement of Bridges by Conversion in Cable-Stayed Arch

Camilo Vásquez[1], Broderick Crawford[1], Ricardo Soto[1], José Lemus-Romani[1](✉), Gino Astorga[2], Sanjay Misra[3], Agustín Salas-Fernández[1], and José-Miguel Rubio[4]

[1] Pontificia Universidad Católica de Valparaíso, Valparaíso, Chile
{camilo.Vasquez.e,jose.lemus.r,juan.salas.f}@mail.pucv.cl, {broderick.crawford,ricardo.soto}@pucv.cl

[2] Universidad de Valparaíso, Valparaíso, Chile
gino.astorga@uv.cl

[3] Covenant University, Ota, Nigeria
sanjay.misra@covenantuniversity.edu.ng

[4] Universidad Tecnológica de Chile INACAP, Santiago, Chile
jrubiol@inacap.cl

Abstract. The scouring of piers bridges, caused by hydraulic action, is one of the main risks that the structure suffer over the years. One of the methods in development is to change the structure of the bridge incorporating a upper cable-stayed arch, which allows to implement vertical and network hangers in charge of lifting the original bridge board. For this it is necessary to optimize the order and the adjustment magnitudes tension of the hangers. To solve this problem we implemented a software for optimization which uses Galactic Swarms Optimization, which is inspired by the movement of the stars, which is inspired by the movement of the stars, galaxies and superclusters under the influence of gravity. When comparing the results obtained with other approximate techniques, we can observe from the diagrams of distribution of instances that level two of the algorithm does not have the necessary and expected capacity to solve or leave from a local optimum.

Keywords: Reinforcement of bridges · Metaheuristics · Galactic Swarm Optimization · Combinatorial optimization

1 Introduction

The collapse of the beam bridge is a great threat. 70% of the bridges collapse by hydraulic action, of which 35% collapse by scouring their piers [7]. One of the possibilities to avoid the collapse of the bridges is to use reinforcement techniques, such as the one proposed by Matías Valenzuela in his doctoral thesis [10], the bridging method by means of cable-stayed arch conversion. This method allows

S. Misra et al. (Eds.): ICCSA 2019, LNCS 11623, pp. 108–119, 2019.
https://doi.org/10.1007/978-3-030-24308-1_10

the change of the structure making a new configuration of the bridge, implementing hangers active and passive, in charge of lifted the original bridge board. The tension of the hangers must be done sequentially.

For this, it is necessary to optimize the order and the magnitudes of adjustment of the tension of the hangers. To solve the problem raised, the Galactic Swarm Optimization (GSO) algorithm will be used. The GSO algorithm simulates the movement of stars, galaxies and superclusters in the cosmos. The stars are not evenly distributed in the cosmos, but are grouped into galaxies that in turn are not evenly distributed. On a sufficiently large scale, the individual galaxies appear as point masses. The attraction of stars within a galaxy to large masses and galaxies to other large masses is emulated in the GSO algorithm. The results obtained by GSO will be compared with other optimization techniques.

The metaheuristics implemented in very complex problems is not something new [2,3,9], while the use of these techniques for problems related to bridges is not [4,6,11].

This paper is ordered as follows, in Sect. 2 the problem to be solved is presented, in Sect. 3 the metaheuristic to be implemented, in Sect. 4 the results obtained together with the corresponding statistical analysis to compare performance, ending with the Sect. 5 with the conclusions of the work.

2 Problem

One of the most important problems presented by bridges crossing river channels is the undermining of their piers. The most important consequence is the total collapse of the structure, generating high human and economic costs. As a result of this, several systems of inspection, monitoring and maintenance of the submerged bridge infrastructure have been implemented.

There are several statistics worldwide that confirm this fact. The work developed by [7] states that 70% of bridge collapses have a cause in hydraulic action, where scouring reaches 35%.

The reinforcement of bridges is proposed as a methodology, which directly attacks the problem of undermining. The reinforcement consists of a arch that goes over the bridge, from end to end, which by means of hangers supports the board. The proposed method has advantages of cost and time, also presents problems from the engineering point of view. For constructive reasons, the tension of the hangers can not be performed simultaneously, so they must be done in sequential order. In addition, excessive stress can cause damage to the structure, instability or collapse of it. This is why we are facing a direct problem of optimization. What is the proper tension order? And with what magnitude should the hangers hold?

A bridge before reinforcing it is functional and is constructed in such a way that it supports both the board and the effort it makes in the passage of vehicles along it, that is, the bending of it. Because of this we will use the original bridge as a reference, so when installing the arc and tensioning the hangers, the objective is to minimize the tension difference. It is expected that the tension in each longitudinal fraction of the bridge will vary depending on the order and magnitude of tension of the upper hangers, so a full search does not seem to be a good option. It is worth mentioning that a structural system of cable-stayed arch is hyperstatic and interdependent, that is, the modification of a tension redistributes the efforts in the whole structure.

To solve problems of high computational complexity like this one, several algorithms have been developed inspired by the behavior of nature, such as genetics (GA), swarms, flowers, among others. In this research it will be use Galactic Swarm Optimization (GSO) [8] to find order and magnitude of the stress, which is a metaheuristic algorithm based in population and inspired by the movement of stars, galaxies and superclusters of galaxies in the cosmos.

For the modeling of the bridge and the problem itself, we will use SAP2000, a software for the analysis and design of structures, which allows the API to pass information from a metaheuristic technique to the bridge, as well as request from them properties and relevant information of the structure [1]. The API contains pre-defined functions that can be invoked from a library in different programming languages, allowing a realistic modeling of the bridge. The bridges, meanwhile, are pre-designed in the software for optimization to separate their design and structuring of the optimization algorithm.

2.1 Objetive Function

The objective function is defined as the summation of the difference tense of both top and low, for each one of K cuts and each one of the 2 beams, as described by the equation:

$$min \sum_{i=1}^{2} \sum_{k=1}^{k} |\sigma o_{i,k} - \sigma m_{i,k}| \tag{1}$$

Where $\sigma o_{i,k}$ is the tension of the original bridge on beam i and on the cut k. Meanwhile, $\sigma m_{i,k}$ it is the modified bridge tension in beam i and in cut k. This function is evaluated for both the lower and higher tensions of the board, and the objective is to minimize the differences. In the ideal and utopian case, the optimum is 0, since it would represent absolute equality between the efforts of the original bridge and the modified one, maintaining all of its properties and completely eliminating the problem of scour.

2.2 Constraints

The problem have constraints that must be met to satisfy the objective function

- The hangers cannot be jacking simultaneously.

$$ord_1, ord_2, .., ord_n \in \{1, 2, ..., n\} \tag{2}$$

$$ord_w \neq ord_j \ ; \forall j, \omega \ con \ j \neq \omega; \ j, \omega \in \{1, 2, ..., n\} \tag{3}$$

- The effort of the modified bridge deck should not pass the limits of the Band Admissible Modified (BAM):

$$\sigma m \geqslant \sigma o \tag{4}$$

$$\sigma m \geqslant fct \tag{5}$$

$$\sigma m \leqslant fcmax2 \text{ (in intermediate stages)} \tag{6}$$

$$\sigma m \leqslant fcmax \text{ (in final stages)} \tag{7}$$

Where:
- σm is the tension (top or bottom) of the modified bridge.
- σo is the tension (top or bottom) of the original bridge.
- fct is the maximum tension to traction admissible for the concrete.
- fcmax is the maximum tension to admissible compression for the concrete.
- fcmax2 is the maximum tension to compression for the concrete, extended.

Any tension on the modified bridge deck that is not inside the BAM described from the original model is discarded because can generate damage to the bridge.

3 Galactic Swarm Optimization

GSO is based on PSO, the original algorithm is inspired by swarms like the behavior of flocks of birds and fish, which is a defense mechanism to confuse predators. The GSO algorithm simulates the movement of stars, galaxies and superclusters of galaxies in the cosmos, the distribution of stars in the universe is not done uniformly, however they are grouped into galaxies that in turn are not distributed homogeneously, the attraction of stars within a galaxy to large masses and galaxies to other large masses are emulated in the GSO algorithm as follows:

- Individuals in each subpopulation who are attracted to better solutions in the subpopulation according to the PSO algorithm
- Each subpopulation is represented by a better solution found by the subpopulation and treated as a superswarm
- Superswarm comprises the best solutions found in each subpopulation moving to the PSO algorithm.

The swarm and superswarm movement can be achieved since it is population-based, providing multiple exploration and exploitation cycles by dividing the search in terms of offers, providing the algorithm with more opportunities to accurately locate a local minimum, in the first level it is considered the exploratory phase where potential local minimums are identified, the second level of the GSO algorithm is the exploratory phase which uses the best solutions already calculated by the sub swarms considering the information already calculated in the first level.

The swarm is a set X of D-Tuples that contains ($\chi_j^{(x)} \in R^D$) that consists of M partitions, called subswarms X_i, each of size N, X is randomly initialized within the search space $[x_{min}, x_{max}]^D$.

Each subswarm independently explores the search space, the declaration for updating the velocity and the position are:

$$V_j^{(x)} \leftarrow \omega_1 + C_1 R_1 (P_j^{(x)} - \chi_j^{(x)}) + C_2 R_2 (g^{(i)} - \chi_j^{(x)}) \quad (8)$$

$$\chi_j^{(x)} \leftarrow \chi_j^{(x)} + V_j^{(x)} \quad (9)$$

Algorithm 1. GSO

```
Level 1 Initialization: χ_j^(i), v_j^(i), p_j^(i), g^(i) within [x_min, x_max]^D randomly.
Level 2 Initialization: v^(i), p^(i), g within [x_min, x_max]^D randomly.
for EP ← 1 to EP_max do
    Begin PSO: Level 1
    for i ← 1 to M do
        for k ← 0 to L_1 do
            for j ← 1 to N do
                v_j^(i) ← ω_i v_j^(i) + c_1 r_1 (p_j^(i) - χ_j^(i)) + c_2 r_2 (g^(i) - χ_j^(i));
                χ_j^(i) ← χ_j^(i) + v_j^(i);
                if f(χ_j^(i)) < f(p_j^(i)) then
                    p_j^(i) ← χ_j^(i)
                    if f(p_j^(i)) < f(g^(i)) then
                        g^(i) ← p_j^(i);
                        if f(g^(i)) < f(g) then
                            g ← g^(i);
                        end
                    end
                end
            end
        end
    end
    Begin PSO: Level 2
    Initialize Swarm y^(i) = g^(i) : 1,2,...,M;
    for k ← 0 to L_2 do
        for 1 ← 1 to M do
            v^(i) ← ω_2 v^(i) + c_3 r_3 (p^(i) - y^(i)) + c_4 r_4 (g - y^(i));
            y^(i) ← y^(i) + v^(i);
            if f(y^(i)) < f(p^(i)) then
                p^(i) ← y^(i);
                if f(p^(i)) < f(g) then
                    g ← p^(i);
                end
            end
        end
    end
end
```

The best solutions participate in the next stage of clustering creating a new superswarm

$$y^{(i)} \in Y : i = 1, 2, ..., M \tag{10}$$

$$y^{(i)} = g^{(i)} \tag{11}$$

In this second stage of clustering the velocity and the position are updated according to the following expression.

$$v^{(i)} \leftarrow \omega_2 v^{(i)} + C_3 R_3 (p^{(i)} + y^{(i)}) + C_4 R_4 (g - y^{(y)}) \tag{12}$$

$$y^{(i)} \leftarrow y^{(i)} + v^{(i)} \tag{13}$$

where $p^{(i)}$ is the best staff in relation to the vector $y^{(i)}$, is defined ω_2, r_3 and r_4 in a similar way in the equations. In the first level, g indicates us as the best global and is not updated unless the search finds us one better and this is indicated as a global best of the subswarm.

4 Computational Results

As a first step we must carry out an implementation of the GSO algorithm, the solution vector has the positions of hangers and the magnitude of tights that must be applied, Table 1 show us an example how to represent a solution vector, this representation was proposed by Valenzuela for the reinforcement of bridges by cable-stayed arch.

Table 1. Example of solution vector

Position 1	Position 2	Position 3	Magnitude 1	Magnitude 2	Magnitude 3
3	1	2	0,99	0,87	0,28

In the previous example, the solution indicates that hanger 3 will first test with a magnitude of 99% of its total capacity. Then, hanger 1 with a magnitude of 28% of its total capacity will be tested and so on, remember that if we have N hangers we will have N magnitudes, for this example we choose a N = 3.

4.1 Parameter Settings Used in Experiments

It is necessary to clarify that all instances where executed with the same settings that show us Table 2 for GSO, which was obtained through the parametric scanning technique.

Table 2. GSO parameters

Population	Particles	M	N	L1	L2	EPmax	c1	c2	c3,c4
16	16	3	3	10	5	20	1	2	2,05

4.2 Computational Results

GSO for Bridges reinforcement was implemented in Python 3.6 and executed in a Personal Computer running Windows 10 on Intel core i5-2450M CPU (2.5 Ghz), 4 GB of RAM. Each instance was executes 15 times. After 28 days we got the results proposed by GSO, Table 3 show us the order of the hangers and the magnitude of each instance that must be tensed.

4.3 Comparing Ressults

We proceed to compare the Fitness obtained by GSO against Black Hole (BH). In Table 3, the first comparison of the Fitness obtained in GSO compared with BH is made, in this first comparison we can see that BH obtains better results than GSO, instances AB-TCV, CC-TCV and CR-AA10.

Table 3. Fitness comparation of AB-TCV, CC-TCV and CR-AA10

#	AB-TCV		CC-TCV		CR-AA10	
	BH	GSO	BH	GSO	BH	GSO
1	517068,78	529200,60	520682,20	527060,41	517761,81	519494,37
2	524023,77	530211,73	520212,17	527303,87	514470,31	519266,95
3	524821,68	530061,40	520538,00	527834,13	517102,87	520307,33
4	520848,21	528982,32	524706,86	526450,57	517088,90	521410,82
5	520622,78	529420,80	522918,41	527208,89	514502,45	518775,94
6	520152,44	528410,12	518616,05	527399,37	511024,81	520978,36
7	517564,07	530047,57	519413,56	527084,65	516154,93	521187,18
8	523623,83	530784,11	524143,33	528874,63	512943,62	522132,29
9	519373,56	529621,11	522562,15	528226,65	515740,37	525596,65
10	524246,51	531688,83	515930,44	528306,50	512584,97	523154,52
11	523785,40	528014,20	520447,93	528675,52	517126,96	520018,54
12	520203,82	530353,34	523472,34	531062,87	516430,95	520474,21
13	520872,27	530180,50	517694,83	528173,91	514210,37	523051,55
14	518973,97	531445,54	519115,91	528109,73	513743,60	522499,51
15	522447,22	530585,53	515608,06	527803,39	512808,82	519816,41

In Table 4, the BH aptitude is compared with the GSO. In the instances HW-TCV, PT-TCV and PV-TCV, we can see that BH obtains better results compared with GSO.

Table 4. Fitness comparation of HW-TCV, PT-TCV and PV-TCV

#	HW-TCV		PT-TCV		PV-TCV	
	BH	GSO	BH	GSO	BH	GSO
1	518227,34	527961,52	526676,11	530629,71	521950,83	531314,18
2	518554,72	528067,86	519411,92	532261,71	523427,99	532787,87
3	521443,88	527742,47	525110,87	530873,89	523381,04	532459,42
4	518566,59	528248,59	520610,73	530893,66	524362,17	530547,87
5	516990,68	527836,18	524252,03	531887,35	520927,57	531665,44
6	522992,53	527892,94	520560,47	530988,44	525788,70	531611,54
7	522006,56	528272,14	523880,06	532110,56	518788,67	532980,62
8	519225,68	528719,51	525204,20	532565,11	521114,47	533992,99
9	520271,78	528395,84	523880,06	531777,87	523983,25	533430,95
10	523204,08	529802,91	520863,04	534283,32	522809,15	532294,35
11	520515,56	529823,21	522029,40	532478,93	522648,28	532266,30
12	519526,11	529434,44	522617,22	533405,46	521351,71	531424,12
13	517752,37	528505,20	518340,35	533742,99	523941,25	531446,67
14	521967,48	529975,94	517173,90	531376,60	517407,41	533571,92
15	521974,26	527659,89	518891,31	532841,54	520202,02	531187,13

In Table 5, the BH aptitude is compared with the GSO. In the instances RC-AA10, RD-AA10 and TC-TCV, we can see that again BH obtains better results in comparison with the GSO.

Table 5. Fitness comparation of RC-AA10, RD-AA10 and TC-TCV

#	RC-AA10		RD-AA10		TC-TCV	
	BH	GSO	BH	GSO	BH	GSO
1	514019,85	518737,96	517038,97	523745,78	519824,40	536659,43
2	514605,63	518378,62	519226,40	522347,77	519571,13	537237,85
3	517856,01	520511,92	513572,50	524090,89	524103,15	537710,92
4	517790,49	521976,81	512678,64	522025,96	521301,71	536943,11
5	514457,07	520494,80	511665,24	522714,47	522386,88	537360,60
6	514508,11	522145,48	517200,65	523148,52	522834,91	536770,49
7	510072,74	520155,20	515799,79	521575,51	522191,19	536218,72
8	514272,05	519629,90	516494,40	523663,98	519926,61	539349,44
9	515619,91	520425,45	511107,49	525045,23	522591,31	538048,60
10	513847,14	520283,77	514930,92	523703,71	521634,82	537776,72
11	514603,89	522717,67	509176,30	522197,22	523052,42	538490,51
12	513776,52	521214,74	512632,11	523920,74	524989,00	536447,97
13	514564,78	520679,18	508556,02	523448,38	525268,02	536403,85
14	511927,54	522722,21	521000,00	524393,51	520641,95	536528,36
15	513593,24	521267,85	513408,64	521249,26	519598,17	537071,14

Finally, in Table 6, BH is compared with GSO in the instances VC-TCV and WR-TCV, observing similar results to the instances previously compared.

Table 6. Fitness comparation of VC-TCV and WR-TCV

#	VC-TCV		WR-TCV	
	BH	GSO	BH	GSO
1	517454,84	527373,52	521210,45	528830,37
2	516764,75	527174,76	519448,38	528816,50
3	526420,20	528192,99	519880,23	529534,30
4	523586,26	527802,89	519204,81	529416,28
5	517878,92	528666,35	526646,98	529143,40
6	521676,41	528720,57	523896,15	530172,24
7	515963,27	529040,68	519162,74	529313,30
8	524091,12	528679,67	522611,37	529544,89
9	520710,80	530326,24	522236,03	529608,69
10	520958,23	528896,13	518685,87	527664,26
11	522523,26	528854,60	521471,35	529841,96
12	521104,33	529145,63	523417,26	528941,97
13	521939,46	528667,78	525384,95	530628,27
14	521888,84	527765,76	520876,22	530696,92
15	520548,39	526734,62	522051,32	530893,56

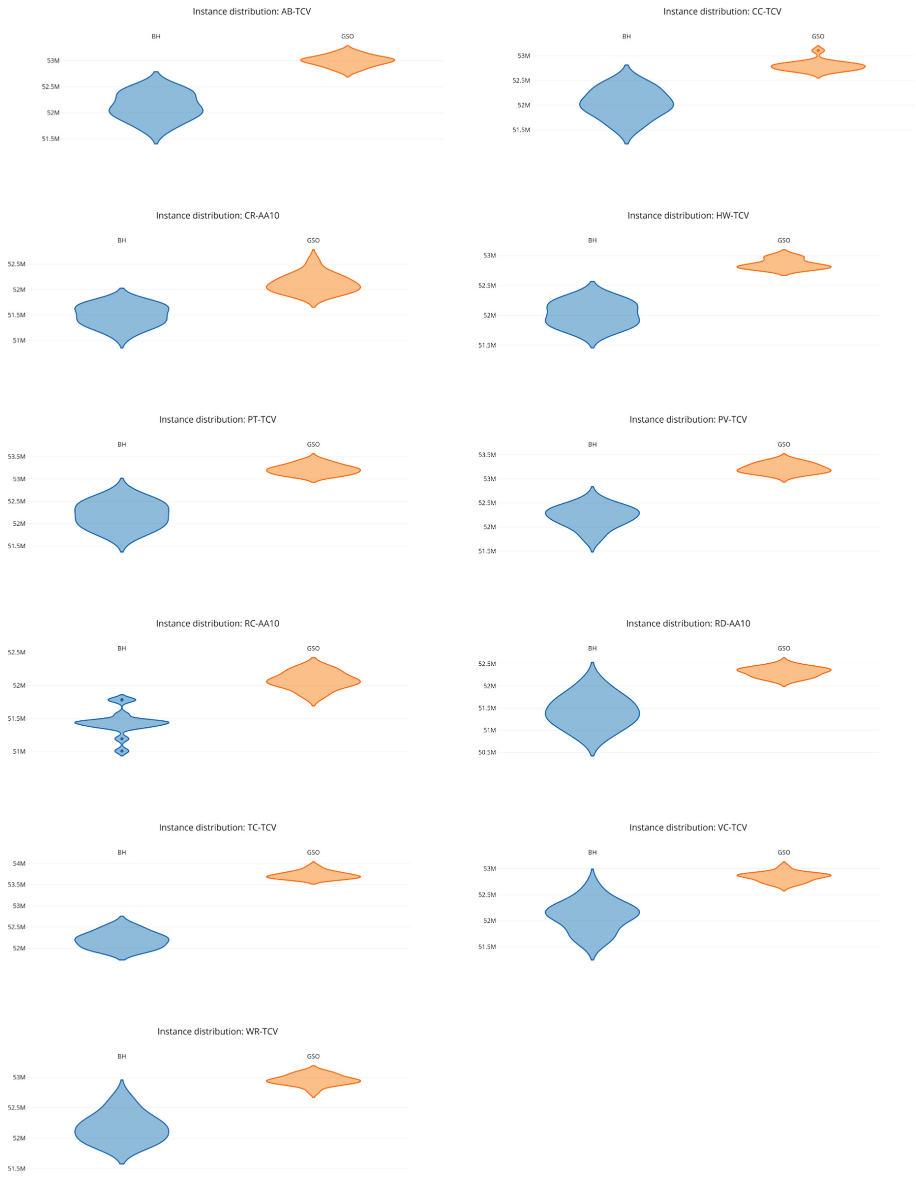
The Table 7 describe us each instance with the minimum value of fitness of GSO and BH.

Table 7. Fitness differences

Instance	Min GSO	Min BH	Difference
AB-TCV	528014,2050	517068,779	10945,4260
CC-TCV	526450,57	515608,056	10842,514
CR-AA10	518775,9410	511024,809	7751,1320
HW-TCV	527659,8893	516990,684	10669,2053
PT-TCV	530629,7060	517173,901	13455,8050
PV-TCV	530547,8720	517407,41	13140,4620
RC-AA10	518378,6210	510072,744	8305,8770
RD-AA10	521249,2605	508556,024	12693,2365
TC-TCV	536218,7240	519571,134	16647,5900
VC-TCV	526734,6176	515963,267	10771,3506
WR-TCV	527664,2601	518685,865	8978,3951

4.4 Instance Distribution

We will compare the distribution of the samples of each instance through a violin plot that shows the full distribution of the data.



Although GSO privileges exploration as exploitation, we can deduce that according to all the cases studied, the second level of PSO is not enough to be able to leave a local optimum, and it is less effective when converging compared to BH.

4.5 Statistical Tests

We perform the statistical tests between the mentioned algorithms BH and GSO. Where Kolmogorov-Smirnov-Lilliefors test allows us to analyze the normality of our 15 executions of each instance, obtaining a non-parametric distribution. While Mann-Whitney-Wilcoxon test [5] It is used to buy the performance of each algorithm for this particular problem.

The test carried a p-value of less than 0.05, therefore H0 cannot be assumed, so the samples are independent of each other. To evaluate the heterogeneity of samples and compare all the results of each instance we used a non-parametric test called Mann-Whitney-Wilcoxon. We propose the following hypotheses:

- H_0: GSO is better than BH
- H_1: States the opposite.

Table 8. p-value Mann-Whitney-Wilcoxon test

Instance	GSO vs BH	BH vs GSO
AB-TCV	0.999998467	1.53348888e-006
CC-TCV	0.999998467	1.53348888e-006
CR-AA10	0.999998467	1.53348888e-006
HW-TCV	0.999998467	1.53348888e-006
PT-TCV	0.99999847	1.5296211e-006
PV-TCV	0.999998467	1.53348888e-006
RC-AA10	0.999998467	1.53348888e-006
RD-AA10	0.999998467	1.53348888e-006
TC-TCV	0.999998467	1.53348888e-006
VC-TCV	0.999998467	1.53348888e-006
WR-TCV	0.999998467	1.53348888e-006

As we can see in Table 8, BH is better than GSO in all instances and GSO is better than BH with an error of 99.99% in all instances.

5 Conclusion

According to the statistical analysis performed we can concluded that the GSO is not better than BH to solve the problem of bridge reinforcement through conversion of cable-stayed arch. We can observe in the distribution diagrams of instances that level two of the GSO algorithm does not have the necessary and expected capacity to solve or leave from a local optimum. In order to improve the search capacity of the algorithm, it is necessary that the exploration capabilities of the second level must be enhanced, as well as a better adjustment of the parameters in the applied algorithm. This it can be discussed in a future work.

References

1. Sap2000. http://www.csiespana.com/software/2/sap2000
2. Crawford, B., Soto, R., Berrios, N., Olguín, E., Misra, S.: Cat swarm optimization with different transfer functions for solving set covering problems. In: Gervasi, O., et al. (eds.) ICCSA 2016. LNCS, vol. 9790, pp. 228–240. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-42092-9_18
3. Crawford, B., Soto, R., Johnson, F., Misra, S., Paredes, F.: The use of metaheuristics to software project scheduling problem. In: Murgant, B., et al. (eds.) ICCSA 2014. LNCS, vol. 8583, pp. 215–226. Springer, Cham (2014). https://doi.org/10.1007/978-3-319-09156-3_16
4. García-Segura, T., Yepes, V.: Multiobjective optimization of post-tensioned concrete box-girder road bridges considering cost, CO2 emissions, and safety. Eng. Struct. **125**, 325–336 (2016)
5. Mann, H.B., Whitney, D.R.: On a test of whether one of two random variables is stochastically larger than the other. Ann. Math. Stat. **18**, 50–60 (1947)
6. Martí, J.V., García-Segura, T., Yepes, V.: Structural of precast-prestressed concrete U-beam road bridges based on embodied energy. J. Clean. Prod. **120**, 231–240 (2016)
7. Muñoz, E., Valbuena, E.: Los problemas de la socavación en los puentes de colombia. Revista de Infraestructura Vial, numeral, p. 15 (2006)
8. Muthiah-Nakarajan, V., Noel, M.M.: Galactic swarm optimization: a new global optimization metaheuristic inspired by galactic motion. Appl. Soft Comput. **38**, 771–787 (2016)
9. Soto, R., et al.: Autonomous tuning for constraint programming via artificial bee colony optimization. In: Gervasi, O., et al. (eds.) ICCSA 2015. LNCS, vol. 9155, pp. 159–171. Springer, Cham (2015). https://doi.org/10.1007/978-3-319-21404-7_12
10. Valenzuela, M.: Refuerzo de puentes de luces medias por conversión en arco atirantado tipo network. PhD thesis, Universitat Politècnica de Catalunya, Barcelona, España, February 2012
11. Yepes, V., Martí, J., García-Segura, T., González-Vidosa, F.: Heuristics in optimal detailed design of precast road bridges. Arch. Civil Mech. Eng. **17**(4), 738–749 (2017)