STRUCTURAL OPTIMIZATION DESIGN OF WIND TURBINE TOWER USING GAME THEORY

Andreas E. Kampitsis[1], Gerasimos D. Panagiotakopoulos[2], Vlasis K. Koumousis[3], Evangelos J. Sapountzakis[4]

[1-4]Institute of Structural Analysis and Antiseismic Research
School of Civil Engineering
National Technical University of Athens, Greece
Athens, GR-157 80, Greece

[1]e-mail: cvakamb@gmail.com [2]e-mail: gerasimospan@gmail.com
[3]e-mail: vkoum@central.ntua.gr; web page: http://users.ntua.gr/vkoum
[4]e-mail: cvsapoun@central.ntua.gr; web page: http://users.ntua.gr/cvsapoun

Keywords: Structural Optimization, Game Theory, Nash Equilibrium, Wind Turbine Tower, Weight Minimization, Budget Limitation, Static Analysis, Finite Element Method.

Abstract. *The objective of this paper is to present a Structural Optimization technique based on Game Theory for the design of the Wind Turbine Tower. In order to surpass the limits of the classic optimization problem, the Maximization of Minimum Utility Method is introduced. According to this method, the discrete problem of structural optimization is simulated as a game, played by the defending player, who defends the structure, managing the stiffness of the materials and cross sections at hand, and the attacking player, who decides upon the way to attack the structure, by choosing among some predefined load combinations, usually described by code regulations. The optimization algorithm is based on statically imposed loading, while the structure is analyzed using Finite Element Method.*

1 INTRODUCTION

Soft types of energy or "renewable energy sources" have received a great amount of interest and considerable growth over the last years. Wind energy is considered as one of the most attractive solutions to the problem of electric power production, while the economic benefits generated by wind turbines can be further raised by decreasing construction costs (weight saving) while increasing the wind turbine efficiency. To this end, extensive research efforts have been presented in the literature concerning the detailed modeling of the tower [1-5], while the structural optimization theory has been successfully applied to determine the optimum tower and/or blade properties [6-9].

Over the years, optimization methods [10] have successfully been applied to several financial and structural models, as well as to industrial problems. Structural optimization was first introduced by Michell [11] in 1904 while little attention was paid until the 1950s. At first the optimization field was dominated by gradient based methods, such as "Optimality Criteria" [12], however in recent years there has been a rapid development of non-gradient based methods, namely evolutionary algorithms such as Genetic Algorithms [13,14] and Simulated Annealing [15, 16]. Non-gradient-based methods have the critical advantage of being able to handle both continuous and discrete optimization problems.

In this work, a Structural Optimization technique based on Game Theory is proposed, for the design of the Wind Turbine Tower. In order to surpass the limits of the classic optimization problem, the Maximization of Minimum Utility Method is introduced. According to this method, the discrete problem of structural optimization is simulated as a game, played by the defending player, who defends the structure, managing the stiffness of the materials and cross sections at hand, and the attacking player, who decides upon the way to attack the structure, by choosing among some predefined load combinations, usually described by code regulations. The optimization algorithm is based on statically imposed loading, while the structure is analyzed using Finite

Element Method. Even though, the game theory approach on structural optimization has been presented in the past [17], however game theory and structural optimization are still widely thought to be unrelated.

2 STRUCTURAL OPTIMIZATION BASED ON GAME THEORY

2.1 Classic Structure Optimization

According to the classic structural optimization techniques a well defined problem has the following properties:

- *A given structure to be optimized*. Some of the properties of the structure may be already known, such as the rough geometry, the materials, the support conditions, loadings, etc. However, it is worth noting that not everything stated above is always known or constant. For example, changing the geometry or the materials may often come into consideration in some optimization problems.
- *An objective function to be optimized*. In most problems, this corresponds to weight or cost minimization. In this paper, other objective functions, such as minimizing a certain displacement is also going to be examined.
- *A set of design variables*, which will optimize the objective function. Usually, these variables are the properties of each group of elements of the structure (sizing problem), while connectivity and geometry can also be design variables. In this paper, it is assumed that the design variables refer to properties of the elements and that the values of the design variables are discrete.
- *A set of constraints*. Usually, these constraints refer to the stresses or displacements of a structure. In this paper the cost will also be included.

2.2 Maximization of Minimum Utility Method

Within the framework of this paper, the Maximization of Minimum Utility Method (MMUM) is proposed. MMUM approaches the discrete structure optimization problem by substituting it into an equivalent game situation, and then solving the problem by using the game theory. In order to deal with the discrete structure optimization problem, MMUM shifts its focus from optimizing the objective function to maximizing the utility function.

According to game theory, the players, the strategies, the rules and the players' payoffs have to be fully defined in order to set a game. To this end, the properties of the game described as follows:

2.2.1 Players

It is assumed that the game is played by two players, the defending player, or player A, and the attacking player, or player B. In this game, the engineer is going to play the role of player A, whereas a fictitious player is going to play the role of player B. Since the incentives of player B are not well defined, it is going to be assumed that player B has exactly opposite wishes to those of player A. This will turn the game into a "non-cooperative game", thus encouraging player A to maximize his own payoffs given that player B wishes to do the exact opposite. According to the rules of the game, player A is going to defend the structure by choosing one of the possible designs for the given structure. Player B is then going to attack the structure, by choosing one of the possible loading cases. One very important rule is that player A is going to choose his strategy before player B. This gives player B an advantage, since he is going to have to choose his strategy after he has observed the strategy of player A. Thus, since both players are rational, player A is going to embed the above fact in his decision making.

The correct formulation of the players' payoffs is the most critical part of the method. Since the payoffs are the players' motives for playing optimally, an accurate formulation of the payoffs is essential for the total equivalence of the game situation to the optimization problem. It is only logical to assume that the payoffs of the defending player will decrease when each of the stresses, displacements and cost increase. In eqn. (1) a realistic suggestion for the utility that the defending player enjoys due to the degree of freedom d_i is presented, while in Figure *1* the same is depicted through a graphic representation. Eqn. (1) is described by two parts, depending whether a certain value of limit displacement has been exceeded. Coefficient a_{di1} describes the linear part, coefficients a_{di2} and k_d describe the non-linear part, whereas P_{di} corresponds to the gap (penalty) between the two parts.

$$u_{d,i}\left(d_i\right)=\left\{\begin{array}{cc} -a_{di1}\cdot\left|d_i\right| & ,\left|d_i\right|\leq d_{i,\lim} \\ -a_{di1}\cdot d_{i,\lim}-P_{di}-a_{di2}\cdot\left\|d_i\right|-d_{i,\lim}\right|^{k_d} & ,\left|d_i\right|>d_{i,\lim} \end{array}\right\} \tag{1}$$

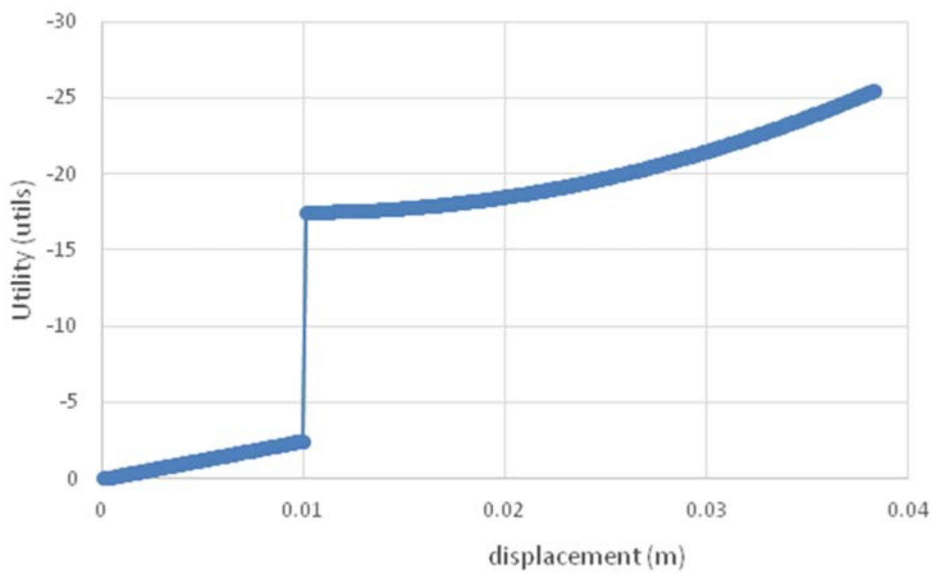

Figure 1: Utility function of the defending player for a single degree of freedom di.

Similar relationships hold between the defending player's utility and the stress σ of an element i (eqn. (2)) or the cost of the structure (eqn. (3)). For these equations the same assumptions hold. That is, there exists a value ($\sigma_{i,\lim}$ corresponds to the yield stress of the element i in eqn. (2), and B, corresponds to the budget in eqn. (3)) that should not be exceeded, and an appropriate penalty is imposed in order to discourage such a concession.

$$u_{\sigma,i}\left(\sigma_i\right)=\left\{\begin{array}{cc} -a_{\sigma i1}\cdot\left|\sigma_i\right| & ,\left|\sigma_i\right|\leq \sigma_{i,\lim} \\ -a_{\sigma i1}\cdot \sigma_{i,\lim}-P_{\sigma i}-a_{\sigma i2}\cdot\left\|\sigma_i\right|-\sigma_{i,\lim}\right|^{k_\sigma} & ,\left|\sigma_i\right|>\sigma_{i,\lim} \end{array}\right\} \tag{2}$$

$$u_{\text{cost}}(c)=\left\{\begin{array}{cc} -a_{\text{cost},1}\cdot c & ,c\leq B \\ -a_{\text{cost},1}\cdot B-P_{\text{cost}}-a_{\text{cost},2}\cdot\left(c-B\right)^{k_c} & ,c>B \end{array}\right\} \tag{3}$$

Having evaluated the partial utility of player A for every displacement, stress and taken into account the cost of the structure, all the partial utilities should be summed in the appropriate way. This is done in two steps. Firstly, the partial utilities for displacements and stresses in general are calculated by grouping together all $u_{d,i}\left(d_i\right)$ and $u_{\sigma,i}\left(\sigma_i\right)$ in eqns.(4),(5). Secondly, utilities of displacements, stresses and cost are summed up, using appropriate weight coefficients, as presented in eqn.(6).Finally, having calculated the utility of player A, player B's utility is the exact opposite quantity (eqn.(7)).

$$u_d = w_d \cdot u_{d,i} \quad u_\sigma = w_\sigma \cdot u_{\sigma,i} \tag{4,5}$$

$$u_{\text{A}} = \left\{ w_{\text{cost}} \quad w_d \quad w_\sigma \right\} \cdot \left\{ \begin{array}{c} u_{\text{cost}} \\ u_d \\ u_\sigma \end{array} \right\} \tag{6}$$

$$u_{\text{B}} = -u_{\text{A}} \tag{7}$$

2.2.2 Solving the game

Having defined the game explicitly, game theory will determine the solution. The simplest way to solve this game is by first representing the game in its tree form and then solving it by using backward induction. The biggest advantage of the tree form representation is that it can allow sequential decisions, which is necessary in this game, since player B decides after player A. In Figure 2 the tree form of the game is depicted. The tree contains nodes and branches. Each node contains the name of the player who has to make a decision, while each branch corresponds to a strategy available to the player. Depending on the players' decisions, different routes of the tree may be followed. At the end of each route there is a single number, corresponding to player A's utility (of course the utility of player B is also explicitly stated in the game tree, due to eqn.(7)). Since the actual number of strategies available to each player is not the same in every problem, each decision node of the player will have 3 branches, corresponding to the first, last and one general strategy. On further analysis, M is going to represent the number of strategies (designs) available to player A and N is going to represent the number of strategies (load cases) available to player B.

The easiest method to approach and solve a game in its tree form is backward induction. According to backward induction, the tree is traversed starting from the nodes in the bottom. At each node, the best strategy for the deciding player is going to be found and the node is going to be replaced by the value of the game (i.e. the utilities for both players), if the tree ever reaches the node. When this procedure is finished for the whole tree, then both the value of the game and the best strategies will be known.

When applying backward induction to this particular game the following conclusions arise. Firstly, in every decision node of player B, it is obvious that the attacking player will choose the strategy that maximizes his utility, simultaneously minimizing the utility of the defending player. It is therefore a certainty that should the game reach the i^th decision node of player B, the value of the game will be $(u_i, -u_i) : u_i = \min(u_{ij}),\ j = 1,...,N$. After substituting all decision nodes of player B with the corresponding value of the game, given that player B will always choose his optimal strategy, the game is simplified, as shown in **Σφάλμα! Το αρχείο προέλευσης της αναφοράς δεν βρέθηκε.**. After establishing player B's decision under all possible circumstances, player A is left with a simple decision. Since player B will minimize his utility, no matter which design player A chooses, he is going to choose the design with the maximum minimal utility $\max(u_i),\ i = 1,...,M$.

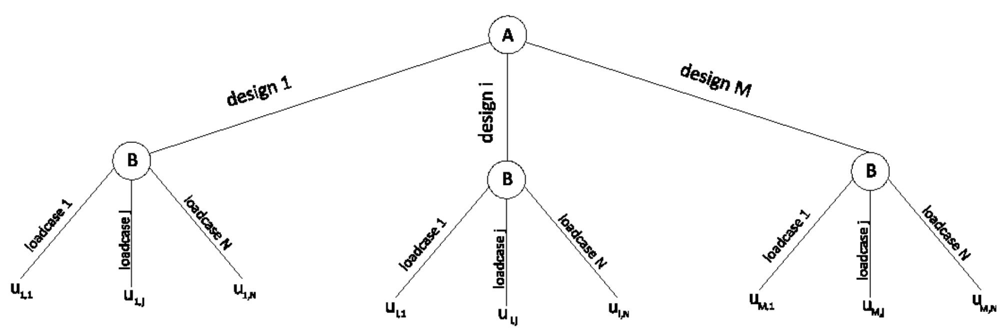

Figure 2: Tree Form of the game

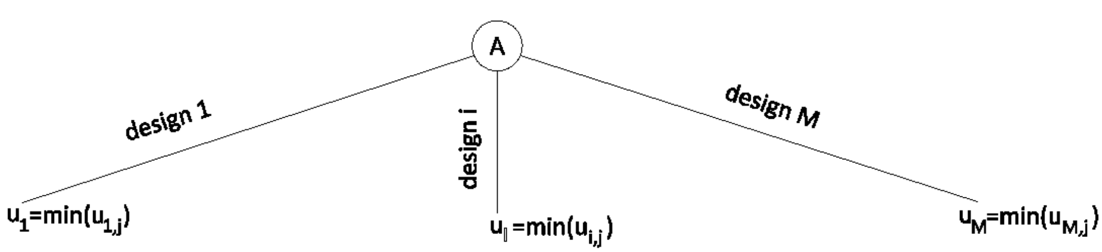

Figure 3: Tree Form of the game after substitutions

2.3 Heuristics

While the correctness of the method is guaranteed, the running time of MMUM is not appealing. This is due to the overwhelmingly large number of possible designs available to player A. Assuming that there are m distinct element groups, each of which can take n distinct values, the number of possible designs (combinations) is n^m. Since such running time is unacceptable, two heuristic methods were used in order to accelerate the solution. The key idea behind heuristic algorithms is to avoid searching a huge solution space and make educated guesses about where the optimal solution would lie. Although a small piece of accuracy is sacrificed, there is a huge computational bonus. However, in order to go into further detail, the design space has first to be specified.

2.3.1 Design Space

The design space of a structure is the set of all possible designs. A single design of the structure is a vector consisting of as many elements as the number of design variables. Therefore, each element of the design corresponds to a single property of one element group of the structure. The value of the element determines the ID of the corresponding property, which can be looked up in an ID table constructed at the beginning. The design space is a matrix containing all these vectors. Thus, if d is the design space, then an element $d_{i,j}$ represents the ID of the j^{th} property of the i^{th} design.

2.3.2 Simulated Annealing:

Simulated Annealing (SA) is one of the most common heuristic algorithms, known since 1983 [18]. SA is applied to this problem as follows: One design d_i is selected at random and its minimum utility u_{d_i} is evaluated. Afterwards a new design $d_{i'}$ is created by randomly mutating d_i. The new utility $u_{d_{i'}}$ is evaluated and compared to u_{d_i}. Then the new design is selected as the current best design with probability $p = \exp\left(\frac{u_{d_{i'}} - u_{d_i}}{T}\right)$, where T is a parameter corresponding to the inverse of algorithm experience. This procedure is repeated for a number of iterations for every T, for descending values of T.

2.3.3 Maximum Descent Algorithm

The other heuristic algorithm used is called Maximum Descent Algorithm, and was developed for the purposes of accelerating MMUM. It is faster than SA, although less accurate and can only work for minimum weight problems. In order to work, a certain pre-processing step is required. All property values need to be sorted in descending ID order. The algorithm is applied as follows: Firstly the design with the maximum feasible IDs, $d_{\max}$ is selected as the current design, d_i. Afterwards, a neighborhood of d_i is created. A neighborhood $N(d_i)$ of size (m^-, m^+) is the design subspace which contains all feasible designs $d_{i'}$ for which eqn. (8) holds. After the creation of $N(d_i)$ this new subspace is thoroughly searched for the local optimum $d \in N(d_i) : u(d) = \max$. The procedure is repeated until convergence, $d = d_i$.

$$-m^- \le d_{ij} - d'_{ij} \le m^+ ,\ \forall i = 1,2,...,n \tag{8}$$

3 NUMERICAL EXAMPLE

In this study a structural optimization methodology is developed in order to design the wind turbine tower under certain constraints. To this end, let us consider a 120m high wind turbine fixed end tower, as presented in Figure 4. The tower is divided into 4 cross circular beam elements, each of which is allowed to get its own diameter and thickness from a variety of predetermined values presented in Table 1. Three different loading scenarios are investigated as presented in Table 2. Two problems are going to be solved:
Problem 1: The minimum weight of the wind turbine, considering the following constraints:

- Maximum node deflections according to the first mode shape, with maximum deflection at the top 2.0m
- The stress at the base cannot exceed the yield stress $f_y = 355\text{MPa}$.

Problem 2: Minimize the deflection at the top of the tower, with a given budget 500.000€. Again, the stress at the base cannot exceed the yield stress $f_y = 355\text{MPa}$ .Assume that cost of steel is $c = 1.30€ / kg$.

To solve each problem, the utility coefficients need to be defined appropriately. In Tables 3 and 4 the values of these coefficients are available for each problem separately. In the sizing problem, stresses and displacements are constraints, therefore their respective utility functions are characterized by very small linear part coefficients and very large penalties. On the other hand, the cost of the structure is the objective function of the sizing problem, therefore the linear part coefficient is significantly larger than its counterparts, but not large enough to surpass a possible penalty. Of course these coefficients characterize the problem itself and do not change depending on the heuristic algorithm used to accelerate the process.

In Table 5 the collective results of all problems are shown. In all problems the critical loading scenario turned out to be the third one. It is also obvious that in all problems, some of the constraints were exhausted (the top displacement constraint in the Problem 1, the budget constraint in Problem 2) in order to reach an optimal solution. Both algorithms were used to solve Problem 1 and both of them converged on the same solution. The Maximum Descent Algorithm converged a lot faster than SA, however the SA algorithm predicts with greater accuracy. However, in the sizing problem, both algorithms search only a small part of a relatively big design space $\left(size(D) = 19^4 \cdot 16^4 = 8.5 \cdot 10^9\right)$. In Problem 2 only SA was used, since the Maximum Descent Algorithm only works for minimum weight problems.

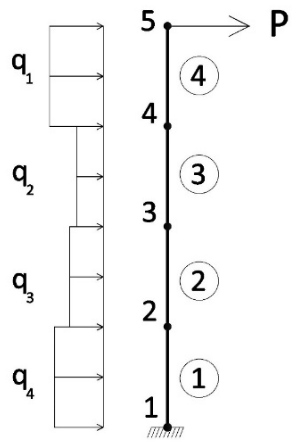

Figure 4. Wind turbine tower simulated as 4 beam finite elements.

ID	Diameter (m)	ID	Diameter (m)	ID	Thickness (m)	ID	Thickness (m)
1	1.00	12	6.50	20	0.02	31	0.042
2	1.50	13	7.00	21	0.022	32	0.044
3	2.00	14	7.50	22	0.024	33	0.046
4	2.50	15	8.00	23	0.026	34	0.048
5	3.00	16	8.50	24	0.028	35	0.05
6	3.50	17	9.00	25	0.03		
7	4.00	18	9.50	26	0.032		
8	4.50	19	10.00	27	0.034		
9	5.00			28	0.036		
10	5.50			29	0.038		
11	6.00			30	0.04		

Table 1. Design space for the Diameter and Thickness values of the tower.

	EC 1-1-4 [19] and Lavassas et al. [20]		Time instant for artificial time history [21-23]
	Scenario 1	*Scenario 2*	*Scenario 3*
Concentrated force at the top of the tower (kN)	700	720	1000
Moment at the top (kNm)	825	0	825
Distributed force along the tower (kN/m)	5	Stem loading	4

Table 2. Loading scenarios.

$u(\text{DOF}_i)$		$u(\sigma_i)$		$u(\text{Cost})$	
a_{di1}	10^{-5}	$a_{\sigma i1}$	10^{-9}	$a_{\text{cost},1}$	0.05
a_{di2}	10^{-5}	$a_{\sigma i2}$	10^{-9}	$a_{\text{cost},2}$	0.05
k_d	1	k_σ	1	k_c	1
P_{di}	10^5	$P_{\sigma i}$	10^5	P_{cost}	0

Table 3. Utility Function Coefficients for Problem 1.

$u(\text{DOF}_i)$		$u(\sigma_i)$		$u(\text{Cost})$	
a_{di1}	0.1	$a_{\sigma i1}$	10^{-9}	$a_{\text{cost},1}$	10^{-5}
a_{di2}	0.1	$a_{\sigma i2}$	10^{-9}	$a_{\text{cost},2}$	10^{-5}
k_d	1	k_σ	1	k_c	1
P_{di}	0	$P_{\sigma i}$	10^5	P_{cost}	10^5

Table 4. Utility Function Coefficients for Problem 2.

Results	Problem 1		Problem 2
Design Variables	SA	MDA	SA
R_1 (m)	3.00	3.00	4.75
R_2 (m)	2.75	2.75	3.75
R_3 (m)	2.00	2.00	2.75
R_4 (m)	1.50	1.50	1.75
t_1 (m)	0.20	0.20	0.20
t_2 (m)	0.20	0.20	0.20
t_3 (m)	0.20	0.20	0.20
t_4 (m)	0.20	0.20	0.20
Cost (€)	354327	354327	498596
u_{top} (m)	1.9871	1.9871	0.7155
% of design space analyzed	0.0117	0.000029	0.0117

Table 5. Collective results in both problems

4 CONCLUDING REMARKS

In this work, a Structural Optimization technique based on Game Theory is proposed, for the design of the Wind Turbine Tower. In order to surpass the limits of the classic optimization problem, the Maximization of Minimum Utility Method is introduced. The main conclusions that can be drawn from this investigation are:

a) The proposed technique is capable of obtaining the optimum design for various structural optimization problems, handling both problems with many loading cases and problems with different objectives with the same ease.
b) The accompanying heuristic algorithms enable the MMUM to find an optimum, or near optimum solution while only searching a very small fraction of the very big design space, thus saving computational time.

5 ACKNOWLEDGMENT

The first and fourth author acknowledge the support of the European Union (European Social Fund - ESF) and Greek national funds through the Operational Program" Education and Lifelong Learning" of the National Strategic Reference Framework (NSRF) – Research Funding Program: THALES: Reinforcement of the interdisciplinary and/or inter-institutional research and innovation.

REFERENCES

[1] Bazeos, N., Hatzigeorgiou, G. D., Hondros, I. D., Karamaneas, H., Karabalis, D. L., Beskos, D. E. (2002), “Static, seismic and stability analyses of a prototype wind turbine steel tower,” *Eng. Struct*., Vol. 24, No. 8, pp. 1015–1025.

[2] Lavassas, I., Nikolaidis, G., Zervas, P., Efthimiou, E., Doudoumis, I. N., and Baniotopoulos, C. C. (2003), “Analysis and design of the prototype of a steel 1- MW wind turbine tower,” *Eng. Struct*., Vol. 25, No. 8, pp. 1097–1106.

[3] Arasu, P. V., Sagayaraj, D., Gowrishankar, J. (2011), “Seismic analysis of a wind turbine steel tower”, *Simulation Driven Innovation, L&T Integrated Engineering Services, HTC, Altair,* HyperWorks Technology Conference, Pune, India, Aug. 4–5.

[4] Al Hamaydeh, M., Hussain, S. (2011), “Optimized frequency-based foundation design for wind turbine towers utilizing soil–structure Interaction,” *J. Franklin Inst*., Vol. 348, No. 7, pp. 1470–1487.

[5] Dimopoulos, C. A., Gantes, C. J. (2013), “Comparison of stiffening types of the cutout in tubular wind turbine towers,” *J. Constr. Steel Res*., Vol. 83, pp. 62–74.

[6] Hani M. Negm, Karam Y. Maalawi (2000), “Structural design optimization of wind turbine towers,” *Computers and Structures*, Vol. 74, pp. 649-666.

[7] Uys, P.E., Farkasb, J., Jarmai, K., Van Tonder F. (2007), Optimisation of a steel tower for a wind turbine structure, Engineering Structures Vol, 29, pp. 1337–1342.

[8] Hezhen Y., Yun Z., Qijin L., Jun Z. (2015), “Dynamic reliability based design optimization of the tripod sub-structure of offshore wind turbines”, *Renewable Energy*, Vol. 78, pp. 16-25.

[9] Chehouri, A., Younes, R., Ilinca, A., Perron, J. (2015) "Review of performance optimization techniques applied to wind turbines", *Applied Energy*, Vol. 142, pp. 361–388.

[10] Schoofs, A. (1993), "Structural optimization, history and state-of-the-art," *Topics in Applied Mechanics*, pp. 339-345.

[11] Michell, A. (1904), "The limits of economy of materials in frame," *Philosophical Magazine*, Vol. 47, No. 8, pp. 589-597.

[12] Venkayya, V. B., Khot N. S., Reddy, V. S. (1968), "Optimization of structures based on the study of strain energy distribution," *AFFDL-TR,* pp.68-150.

[13] Rajeev, S., Krishnamoorthy, C. S., (1992), "Discrete optimization of structures using genetic algorithms," *Journal of Structural Engineering*, Vol. 118, pp. 1233-1250.

[14] Koumousis V.K. , Georgiou P.G. (1994) "Genetic algorithms in discrete optimization of steel truss roofs," *ASCE Journal of Computing in Civil Engineering*, Vol. 8 No. 3, pp. 309-325.

[15] Kripka, M. (2004), "Discrete optimization of trusses by simulated annealing," Journal *of the Brazilian Society of Mechanical Sciences and Engineering*, Vol. XXVI, No. 2, pp. 170-173.

[16] Sonmez, F.O.(2008), "Structural optimization using simulated annealing," *Proceedings in Simulated Annealing*, Vienna, InTech, pp. 281-306.

[17] Rao, S.S., (1987), "Game theory approach for multiobjective structural optimization," *Computers & Structures*, Vol. 25, No. 1, pp. 119-127.

[18] Kirkpatrick, S., Gelatt, Jr. C.D., Vecchi, M.P. (1983), "Optimization by simulated annealing," *Science*, Vol. 13, pp. 671-680.

[19] EN 1993-1-4, (2005) Actions on structures – General actions.

[20] Lavassas, I., Nikolaidis, G., Zervas, P., Baniotopoulos, C.C.(2011), "Design of large scale wind turbine towers in seismic areas", *Proceedings in the 7th National Conference of Steel Structures*, Volos, Greece, Vol.1, pp. 272-279.

[21] Vassilopoulou, I., Gantes, C. J., Gkimousis, I., (2011), "Response of cable nets under wind loads", *Proceedings in the* 7th National Conference on Steel Structures, Volos, Greece, Vol. 1, pp. 416-423.

[22] Koulatsou, K., Petrini, F., Vernardos, S., Gantes, C. J. (2013), "Artificial time histories of wind actions for structural analysis of wind turbines," *Proceedings in the 2nd International Balkans Conference on Challenges of Civil Engineering, BCCCE*, EPOKA University, Tirana, Albania, May 23–25.

[23] Sapountzakis, E.J., Dikaros, I.C., Kampitsis, A.E., Koroneou, A.D. (2015) "Nonlinear response of wind turbines under wind and seismic excitations with soil–structure interaction", *Journal of Computational and Nonlinear Dynamics,* Vol. 10, pp.1-16.