Yielding novel k-factor formula according to the aisc standard by machine learning

Nguyen Thanh Tung(1)

Abstract

The results of using machine learning via genetic programming (GP) to automatically generate novel effective length factor formula in accordance with AISC standard are presented in this article. The data points obtained from applying the numerical method equation solving for the transcendental equation for the effective length of the braced frame were fed into the machine learning algorithm. The the formula was compared to the AISC standard's numerical solution method for the equation. As a result, the error in the formula is negligible. Therefore, for greater convenience in practice, the the formula can completely replace the AISC standard's chart.

Key words: *Genetic Programming, Symbolic Regression, Machine Learning, Numeric Analysis Method, Effective Length Factor*

(1) MS, Lecturer, Faculty of civil engineering, Hanoi Architectural University,
Email: nguyenthanhtungb@gmail.com

Date of receipt: 15/4/2022
Editing date: 6/5/2022
Post approval date: 5/9/2022

1. Introduction

In stability analysis, the AISC standard [1] requires determining the effective length for columns in frames. The AISC standard included the concept of effective length factor for frame column design in 1961, and it is still used today.

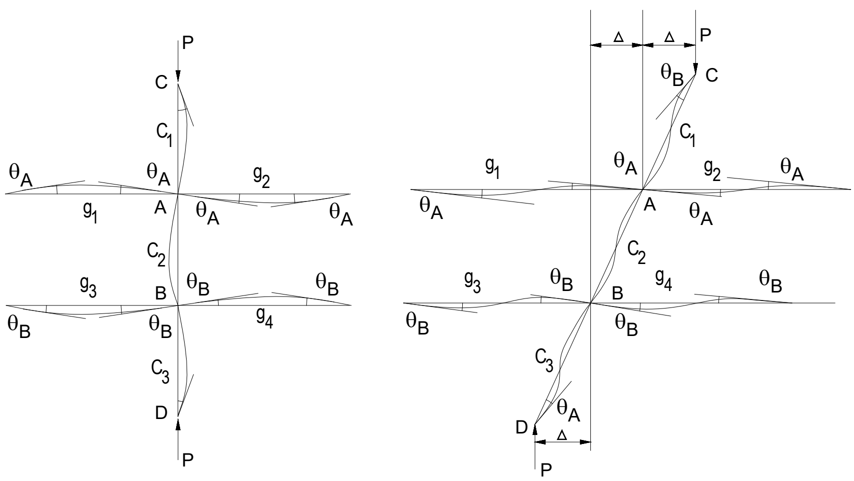
When design for multi-storey frame columns, the effective length factor (K) greatly affects the critical buckling load. Intuitively, this concept is merely a mathematical method to alleviate the problem of calculating the critical stress for a column whose two ends are connected to the frame. The bending moment in the column due to the beam's gravity load does not significantly affect the overall stability of the frame in the elastic range, and only the axial force will have significant effect.

The AISC standard only has one method for calculating the effective length, which is depicted in Figure 2 [1]. The chart makes it possible to obtain the elastic solution of the K-factor without performing an actual stability analysis (which is rather complex). However, if engineers use software such as spreadsheets to automate calculations, charts are no longer valid. As a result, an analytic formula is required to facilitate practical application.

Many engineering problems require solutions to be derived from transcendental equations, experimental data, or numerical simulation data. But most experimental formulae are frequently derived from human experience and performed manually. This has the disadvantage of not providing an optimal formula and a good fit to the data.

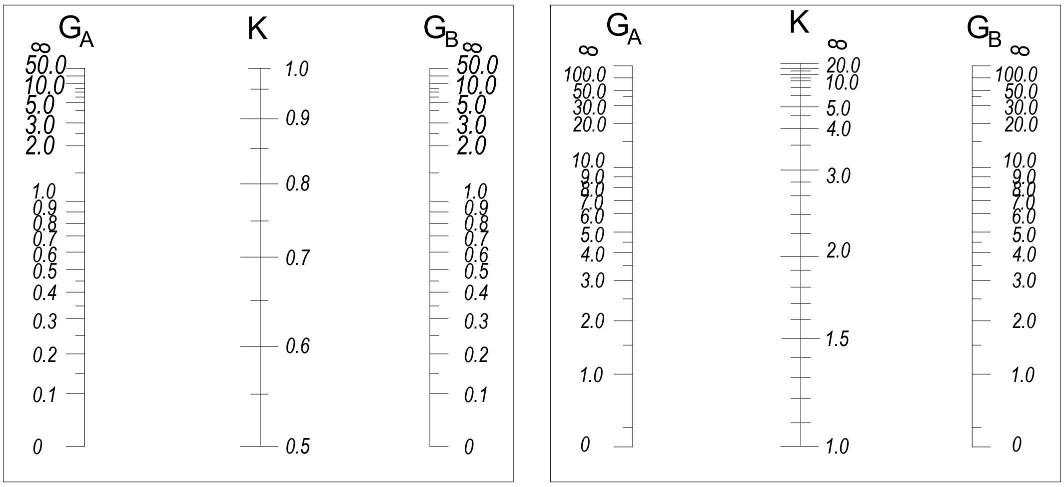
A great difficulty is to find the analytic solution of a general equation that is impossible. Even polynomial equations with degrees greater than 5 do not have algebraic solutions (Abel–Ruffini theorem of 1813 [2]). Richardson's theorem [3], introduced in 1968, states that there is no general analytic solution for algebraic or transcendental equations.

As a result, using machine learning to automatically generate approximate formulas from data collected by numerical or experimental methods is a feasible and effective method. The machine learning method based on genetic programming (GP, John Koza 1990[4]) is popular among the methods to find the formula, also known as symbolic regression (SR)[4]. It has been used in



(a) Braced frame **(b) Unbraced frame**

Figure 1: Models for the K-factor of frame columns



(a) Braced frame (b) Unbraced frame

Figure 2: Design chart for determining the effective length of the column in the frame

a variety of engineering disciplines, producing results that can be considered "inventions" that outperform humans[4]. However, the use of machine learning methods to create design formulas based on empirical data is still limited in the construction industry.

This paper presents a genetic programming-based machine learning method for determining the effective length of a braced frame column using data from numerical analysis. From there, a the formula is convenient in practice with high accuracy is proposed.

In published papers, the authors proposed the K-factor formula of frame columns that relate to the AISC's alignment chart method as following: Newmark 1949 [10]; Julian and Lawrence, 1959 [11]; Kavanagh, 1962 [12]; Johnston, 1976 [13]; LeMessurier, 1977 [14,15,16]; Lui, 1992 [17] ; Duan, King, Chen, 1993[18]; White and Hajjar, 1997 [19,20]. The Standards of steel structure involve formulas for K-factors including: European (prestandard-1992) [21], German, 2008 [22], France, 1966 [23], Russia, 2011 [24].

The K-factor formulas for frame columns in the above material do not coincide with the formula (10) found by GP in the article. The interpolation method is used in all of the K-factor formulas above. Therefore, they differ from the method described in the article in that knowing the form of the formula in advance (based on the builder's experience and knowledge) is required before identifying the formula's coefficients. In this paper, on the other hand machine learning method does not know the formula form in advance, it will automatically determine the formula form and coefficients (symbolic regression).

2. Effective length factor based on theoretical of stability

Frames are classified as braced or unbraced in AISC structural steel design standards[1]. When the stability of the structure is generally provided by walls, braces, or struts that are designed to carry all lateral forces in that direction, the column may be braced in that direction. When the resistance to lateral loads is caused by the bending of the columns, the column is not fully braced in that plane. There are no fully braced frames in practice, and there is no apparent distinction between braced and unbraced frames.

In the AISC [1] steel structure design standard, the interaction between a compression member and an adjacent member or a part of the structure is modeled as shown below.

The elastic stiffness of joints A and B is given by[1]

$$G_A = \frac{\sum\left(E_c I_c / L_c\right)_A}{\sum\left(E_g I_g / L_g\right)_A} \quad (1)$$

$$G_B = \frac{\sum\left(E_c I_c / L_c\right)_B}{\sum\left(E_g I_g / L_g\right)_B} \quad (2)$$

In which, the ∑ means the total stiffness of all elements connected to the joint on the instability plane of the column being considered. I_c is the moment of inertia, L_c is the length between the supports of the column. I_g is the moment of inertia, L_g is the length between the beam supports or other supporting members. I_c and I_g are in axis perpendicular to the buckling plane.

Galambos[5], 1968 solved this problem and gave the following transcendental equation to determine the effective length of the column in the frame.

Unbraced frame[1]:

$$\frac{G_A G_B (\pi / K)^2 - 36}{6(G_A + G_B)} = \frac{\pi}{K}\cot\left(\frac{\pi}{K}\right) \quad (3a)$$

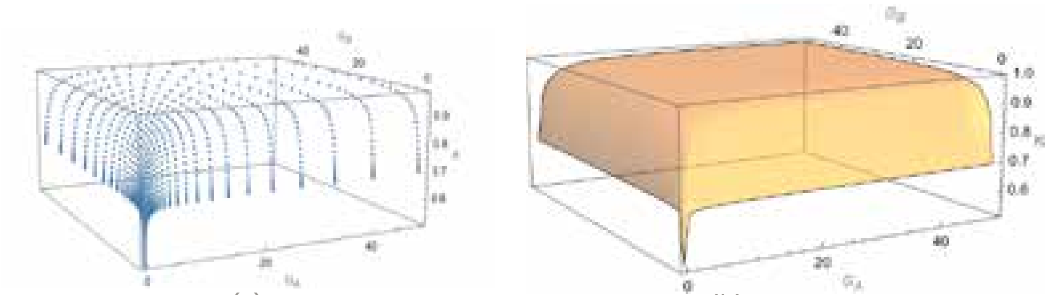
Braced frame[1]:

$$\frac{G_A G_B}{4}\left(\frac{\pi}{K}\right)^2 + \left(\frac{G_A + G_B}{2}\right)\left[1 - \frac{\pi}{K}\cot\left(\frac{\pi}{K}\right)\right] + 2\frac{\tan\left(\pi / 2K\right)}{\pi / K} - 1 = 0 \quad (3b)$$

3. Method of calculating effective length factor according to AISC

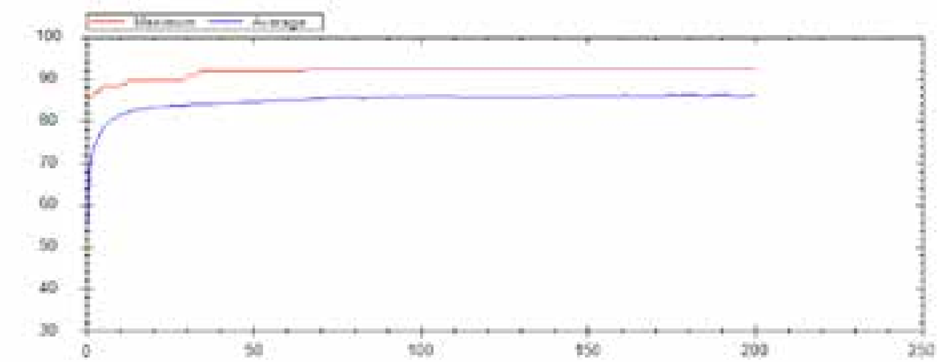
The AISC standard [1] relies on (3a) and (3b) to provide charts for convenient apply in practice. However, this leads to difficulties for applying in spreadsheet software

Where GA, GB is the relative stiffness ratio between the column and the beam at the ends A and B as shown in Figure 2 and is taken from (1) and (2).



(a) (b)

Figure 3 : (a) Plot of the data set obtained from the numerical method for the equation (b) for learning and (b) Plot of learned K-factor formula (10)



i-th generation

Figure 4: Graphs of maximum and average fitness values in evolution generations.

4. Application of Machine learning based on genetic programming to solve the problem of finding K-factor formulas for brace frames from numerical analysis data

4.1. Overview of machine learning by genetic programming

Machine learning has long been used in research [8], but it has exploded in popularity in recent years, thanks to researchers Yoshua Bengio, Geoffrey Hinton, Yann LeCun who won the Turing Award (Nobel Prize in IT) in 2018 [9] for developing a deep learning method. Deep learning, on the other hand, does not allow for the solution of the symbolic regression problem because it relies on an artificial neural network (ANN) and the learning process is just modifying the network's weights. As a result, in the domain of symbolic regression, the genetic programming method remains the most advantageous method.

In 1975, John Holland [6] published a genetic algorithm (GA) that approximates solving the combinatorial global optimization problem. This is an NP-hard problem [7], which is the most difficult class of problems for which there is currently no general solution for all problem instances. GA is used in a variety of fields, including machine learning. However, it does not allow for the solution of the symbolic regression problem. The symbolic regression problem could not be solved until the advent of genetic programming (in 1988, John Koza [4]). Genetic programming is based on genetic algorithms, but instead of data encoded in the form of string genome, it works on tree data structures genome.

4.2. Application of machine learning algorithms to learn the K-factor formula

The application of GP to learn the K-factor formula is described in this section as following.

Let

- $KN:\{G_A,G_B\}\rightarrow K$, $K\in\mathcal{R}+$ where K_N is K-factor value from the numerical solution to equation (3b).
- P is a sample (data point) for learning,
- $P=\{G_A,G_B,K_N(G_A,G_B)\}$, $G_A,G_B\in\mathcal{R}+$.
- T is the data set (data table) which is the set of samples $T=\{P_i\}$, i=1,…,n; n – number of samples.
- T_L is a data set for learning TL $=\{P_j\}\subset T$, j=1,…,l, l- the number of samples to be learned.
- T_T is the data set for evaluation (testing) $T_T=\{P_k\}\subset T$, k=1,…,t, t- the number of samples to be tested.

Two sets T_L and T_T satisfy the following constraint:

$T=T_L\cup TT$, $TL\cap T_T=\emptyset$, from $T=T_L\cup T_T\rightarrow n=l+t$. Typically, there is 80% learning data and 20% testing data i.e. l=0.8n and t=0.2n.

- $K_f^{i,j}:\{G_A,G_B\}\rightarrow K$, $K\in\mathcal{R}+$; where $K_f^{i,j}$ is i-th individual K-factor formula of j-th generation.
- $K_{GP}:\{T_L,B,P_r\}\rightarrow K_f^{best}$, where K_{GP} is a Genetic Programming learner that outputs as an explicit expression of K-factor formula; B – set of basic functions; P_r – set of parameters of a GP learner.
- $K_f^{best}:\{G_A,G_B\}\rightarrow K$, $K\in\mathcal{R}+$, where K_f^{best} is the best outputting K-factor formula,
- $\epsilon_k^{i,j}$ is the error in percentage between $K^{i,j}_f(G^k_A,G^k_B)$ and $K_N(G^k_A,G^k_B)$,

$\epsilon_k^{i,j}=100\times(K^{i,j}_f(G^k_A,G^k_B)-K_N(G^k_A,G^k_B))/K_N(G^k_A,G^k_B)$; (4)

where i=1,…,m; m- the cardinality of the set $\{\epsilon ki,j\}$, i is i-th individual, j is j-th generation.

- ϵ is a member of the set of ϵ_k, $\epsilon\in\{\epsilon_k\}$, i=1,…,m,k=1,…,N,
- $Var[\epsilon]$ is the variance of ϵ, $Var[\epsilon]=E[(\epsilon-\mu)]$, where μ is expected value of ϵ, $\mu=E[\epsilon]$, E is mean of ϵ
- ϵ_{max}, ϵ_{min} is the maximum and minimum absolute errors between the value calculated by the learned formula and

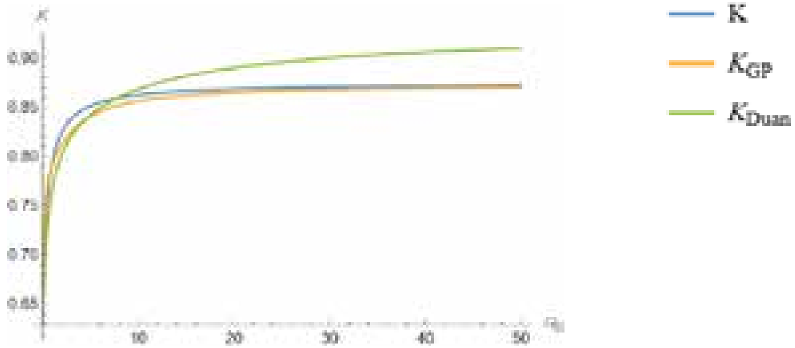


Figure 5: Graph of K(exact),K_{GP}(10),K_{Duan} (11) [18] with G_A=1, $G_B \in$ [0…50]

the numerical solution are given by ϵ_{max}, ϵ_{min} as following: $\epsilon max=max\{|\epsilon i|\}$, i=1,…,m; $\epsilon min=min\{|\epsilon i|\}$, i=1,…,m

- $\epsilon_k^{L,i,j}$, ϵ_k^{T} is learn and test error, i=1,…,m,k=1,…,N, k is k-th sample in T_L.
- $\epsilon^{L,i,j}$, ϵ^T is a member of the set of { $\epsilon_k^{L,i,j}$ },{ $\epsilon_k T$ }.
- $\epsilon^L_{ma}x$, ϵ^L_{min} and ϵ^T_{max}, ϵ^T_{min} is the maximum and minimum absolute errors for learn and test sets.

From above definitions, the fitness function F is implemented as follows:

$$F(K^{i,j}_f (G_A,G_B))=(100-Var[\epsilon^{L,i,j}]) \quad (5)$$

Where i is i-th individual, j is j-th generation.

Convergence condition[4]:

$$Max(F(K_f^{i,j} (G_A,G_B)) - F(K_f^{i,j-1} (G_A,G_B)))\rightarrow 0 \quad (6)$$

The GP learning stage with fitness function F, by input T_L,B and output K_{GP}:{T_L,B,P_r}→K^f_{best};

$$K_f^{best} = arg_{(i)} \max(F(K^{i,j}_f (G_A,G_B))). \quad (7)$$

The GP evaluation stage is to score the learned model based on statistics variables: Var[ϵ^T], ϵ^T_{max}, ϵ^T_{min}, the lower the values, the higher the quality of the learned model.

4.3. Data set for training and evaluation

The data set for the machine learning algorithm to learn the bracing effective length formula is based on the numerical method of solving equations (3b). After extensive testing, it is clear that the function of calculated length increases rapidly when the stiffness G_A,G_B is low and slowly as the stiffness increases (figure 3a). As a result, the final learning data set contains 2500 data points with increasing distances, as determined by the square rule. This achieves the required accuracy without necessitating the use of an excessive number of data points to learn.

$$G^{i+1}_A= G^i_A +\Delta^2 , G^{i+1}_B= G^i_B +\Delta^2, i=1..n \quad (8)$$

Where: Δ is the basic step size Δ = 0.1, n- number of data points of variable G_A, G_B, n=50.

The data used to train machine learning is divided into two sets: learning data set (80%) and testing data set (20%). Overfitting can be avoided by dividing the data set into two parts. Overfitting causes the learned model to be less generalizable, lowering prediction accuracy. This means that some range the the accurary of will be high while others will be low, which should always be avoided when using machine learning.

4.4. Parameters of the genetic programming algorithm

Viewing the plot, one can see that the shape of the data obtained from the numerical method is a monotonically increasing function that is not quite rapidly increasing, as shown in the figure 3, indicating that exponential functions are unnecessary. On the other hand, because the plot is not acyclic, trigonometric functions are unnecessary. The following operators are used from there:

B={+(Plus),-(Minus),×(Times),/(Divide),

^(Power),√ (Square Root), $\tan^{-1}$ (Arctan)} (9)

The following are the ideal parameter values for the problem under examination, as determined by a series of trials with various parameters:

Table 1: Parameters for the algorithm GP

Parameters	Values
Population size	1000
Generations	200
Crossover	0.9
Mutation	0.05
Reproduction	0.2
Maximum initial level	5
Maximum operation level	6

The algorithm starts to converge with number of generations > 100, then the objective function value cannot be improved further. After a number of different runs, the best fitness K-factor formula of braced frame column formula was obtained (Fig. 3b):

$$K = \tan^{-1}\left(\tan^{-1}\left(\begin{array}{l} 1.49\tan^{-1}(0.26G_A)\tan^{-1}(0.26G_B) \\ +0.23\left(\tan^{-1}(4.58G_A)+\tan^{-1}(4.58G_B)\right) \\ +0.55 \end{array} \right)\right) + 0.055 \quad (10)$$

4.5. Result evaluation

The statistical parameters of the machine-learning-discovered formula are listed in the table below:

Table 2: Statistical parameters of the learned formula

Parameters	Values
Var[ϵ^T]	0.15%
ϵ^T_{max}	2 %
ϵ^T_{min}	2.83×10^{-7}%

According to table 2, the maximum absolute error value is only 2%, showing that the given formula is not overfit. The variance throughout the range is 0.15 %, which is a tiny error. The current best formula by Duan (11) [18] has a maximum absolute error value of 5%. A comparison of exact solutions obtained by numerical approach (K), machine learning formula (10) (K_{GP}), and Duan (K_{Duan}) is shown in the graph below:

Where, the KDuan [18] is

$$K_{Duan} = 1 - \frac{1}{5+9G_A} - \frac{1}{5+9G_A} - \frac{1}{10+G_A G_B} \quad (11)$$

5. Conclusion

The research findings demonstrate the advantages of using machine learning to find practical formulas based on data from experiments or numerical methods. It enables formulas with tiny errors across the entire data domain and differs from other methods for its automability. Furthermore, machine learning enables the successful learning of a wide variety of data types and problems./.

References

1. *ANSI/AISC 360-16 An American National Standard Specification for Structural Steel Buildings*
2. *Ruffini, Paolo (1813). Riflessioni intorno alla soluzione delle equazioni algebraiche generali opuscolo del cav. dott. Paolo Ruffini ... (in Italian). presso la Societa Tipografica.*
3. *Richardson, Daniel (1968). "Some Undecidable Problems Involving Elementary Functions of a Real Variable". Journal of Symbolic Logic. 33 (4): 514–520. JSTOR 2271358. Zbl 0175.27404*
4. *Koza, J.R. (1990). Genetic Programming: A Paradigm for Genetically Breeding Populations of Computer Programs to Solve Problems, Stanford University Computer Science Department technical report STAN-CS-90-1314..*
5. *Theodore V. Galambos. Guide to Stability Design Criteria for Metal Structures. John Wiley & Sons, 1988.*
6. *John Holland. Adaptation in Natural and Artificial Systems (1975, MIT Press)*
7. *Knuth, Donald (1974). "Postscript about NP-hard problems". ACM SIGACT Novels. 6 (2): 15–16. doi:10.1145/1008304.1008305. S2CID 46480926.*
8. *Samuel, Arthur (1959). "Some Studies in Machine Learning Using the Game of Checkers". IBM Journal of Research and Development. 3 (3): 210–229. CiteSeerX 10.1.1.368.2254. doi:10.1147/rd.33.0210.*
9. *Fathers of the Deep Learning Revolution Receive ACM A.M. Turing Award Bengio, Hinton and LeCun Ushered in Major Breakthroughs in Artificial Intelligence*
10. *Newmark NM. A simple approximate formula for effective end-fixity of columns. J Aeronaut Sci 1949;16(2)*
11. *Julian, O.G. and Lawrence, L.S. (1959) Notes on J and L Nomographs for Determination of Effective Lengths.*
12. *Kavanagh, T.C. (1962), "Effective Length of Framed Columns," Transactions, Part II, ASCE, Vol. 127, pp. 81–101.*
13. *Johnston, B.G. (ed.) (1976), Guide to Stability Design for Metal Structures, 3rd Ed., SSRC, John Wiley & Sons, Inc., New York, NY.*
14. *LeMessurier, W.J. (1976), "A Practical Method of Second Order Analysis, Part 1—PinJointed Frames," Engineering Journal, AISC, Vol. 13, No. 4, pp. 89–96.*
15. *LeMessurier, W.J. (1977), "A Practical Method of Second Order Analysis, Part 2—Rigid Frames," Engineering Journal, AISC, Vol. 14, No. 2, pp. 49–67.*
16. *LeMessurier, W.J. (1995), "Simplified K Factors for Stiffness Controlled Designs," Re structuring: America and Beyond, Proceedings of ASCE Structures Congress XIII, Boston, MA, ASCE, New York, NY, pp. 1,797–1,812.*
17. *Lui, E.M. 1992. A Novel Approach for K-Factor Determination. AISC Eng. J., 29(4):150-159.*
18. *Duan L, King WS, Chen WF. K-factor equation to alignment charts for column design. ACI Struct J 1993;90(3):242–8.*
19. *White, D.W. and Hajjar, J.F. (1997a), "Design of Steel Frames without Consideration of Effective Length," Engineering Structures, Elsevier, Vol. 19, No. 10, pp. 797–810.*
20. *White, D.W. and Hajjar, J.F. (1997b), "Buckling Models and Stability Design of Steel Frames: a Unified Approach," Journal of Constructional Steel Research, Elsevier, Vol. 42, No. 3, pp. 171–207.*
21. *Eurocode 3, Design of steel structures – part 1.1: general rules and rules for buildings (European prestandard ENV 1993-1-1:1992),*
22. *DIN 18800-2: Stahlbauten – Teil 2: Stabilitätsfälle – Knicken von Stäben und Stabwerken*
23. *Regles de calcul des constructions en acier CM66. Editions Eyrolles, Paris, France; 1966.*
24. *СВОД ПРАВИЛ СП 16.13330.2011 СТАЛЬНЫЕ КОНСТРУКЦИИ Актуализированная редакция СНиП 11-23-81 Издание официальное*