A strategy for the optimal design of steel portal frames

G. S. Brasil[1], F. S. Almeida[2], H. M. Gomes[3]

[1,2,3] *Graduate Program in Civil Engineering (PPGEC), Federal University of Rio Grande do Sul (UFRGS)*
Av. Osvaldo Aranha, 99, 3o. andar, 90035-190, Porto Alegre, RS, Brazil
giovannisbrasil@gmail.com, *felipe.almeida@ufrgs.com*, *herbert@mecanica.ufrgs.br*

Abstract. This article presents a strategy for the optimal design of single-span steel portal frames used in the primary framework of single-story buildings. A methodology that integrates a structural analysis program and a heuristic optimization algorithm for the design process is developed and evaluated. The methodology is applied to the design of single-span symmetric pitched roof portal frames fabricated from built-up welded I-shaped sections with rigid connections. Geometric properties of beams and columns cross-sections are optimized for the minimization of the weight of the portal frame. The flange width and the total section depth are taken as continuous design variables, while web and flange thicknesses are limited to the discrete standard values. The portal frame is subjected to self-weight, the secondary elements weight, the service load and wind load. The optimization constraint functions include checks of serviceability and ultimate limit states given by Brazilian standards. Numerical experiments indicated that the proposed strategy produce high quality design with reasonable computational cost. Additional tests showed that increasing the number of iterations of the optimization algorithms or reducing the range of the design variables based on engineering judgment may improve the efficiency of the design strategy.

Keywords: Steel Structures, Metaheuristic Optimization, Portal Frames, Wind Loads.

1 Introduction

Steel structures are widely used in commercial constructions, such as shopping malls, supermarkets, stores, sports courts, and industrial constructions. The choice of this material is related to economy, flexibility of use, architecture, quick execution, and ease of maintenance. An important aspect in architecture is the requirement to overcome large spans, such as industrial sheds, which usually have span dimensions ranging from 15 to 50 meters. According to Steel Construction Institute (SCI) [1], industrial building structures can be formed by rigid or flexible connections, by beams and columns or latticework, with or without a constant section.

This article focuses on the design of single-span symmetric pitched roof portal frames fabricated from built-up welded I-shaped continuous sections with rigid connections. A strategy for the optimization assisted design of steel portal frames is proposes and its efficiency is evaluated based on the quality and reliability of the solution, as well as the computational cost. The main objective of the optimization is the reduction of the total mass of the structure, while respecting all design constraints. It was developed a code in Python 3 [2] capable of integrating a genetic algorithm (GA), the commercial structural analysis software SAP2000 [3] and all calculation routines for checking serviceability and ultimate limit states given by ABNT NBR 8800:2008 [4]. The parametric representation of the portal frame, shown in Fig. 1, is employed for the automatic construction of the structural model and the evaluation of loads, including the wind load obtained according to ABNT NBR 6123:1998 [5].

Many works addressed the efficiency of metaheuristics algorithms in nonlinear optimization problems formulated for the design of steel structures. Oliveira and Fálcon [6] integrated a MATLAB [7] implementation of (GA) with the commercial structural analysis software ANSYS Mechanical APDL [8] for the optimization of a steel frame with 3 spans and two floors formed by rolled I-shaped section. Advanced aspects such as the connections rotation stiffness and geometric nonliterary were included in the design process for the evaluation of constraints based on displacement and strength given by ABNT NBR 8800:2008[4]. Phan et. al. [9] used a real coded niching genetic algorithm (RC-NGA) for the optimization of steel portal frames with 15 to 50 meters long single-span and 5 to 10 meters height single-story. This work showed that serviceability limits are more critical than strength constraints for the portal frames considered in the study. Mckinstray et. al. [10] used the RC-NGA to optimize a rigid portal frame with built-up welded I-shaped sections considering the ultimate and serviceability limit state based on the recommendations of the SCI [1].

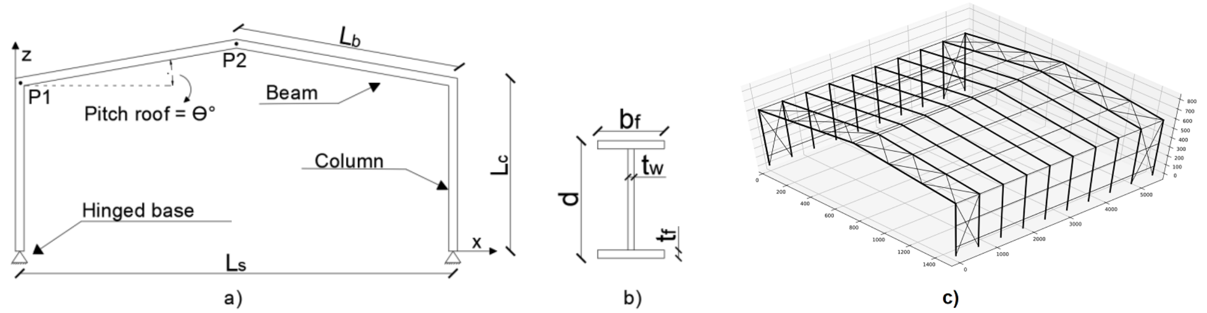

Figure 1. a) portal frame b) Beam and Column cross-sections c) Shed structural global view.

2 Optimization of Steel Portal Frames

The optimum design of the structure shown in Fig. 1 is obtained by the minimization of the objective function given by eq. 1, including the structure mass plus a penalty P for infeasible designs, allowing the solution by methods for unconstrained optimization.

$$f(x) = 2\rho(L_b A_b + L_c A_c) + P, \tag{1}$$

where the parameters L_b and L_c are the beam and column lengths, respectively, and the steel density is ρ = 7850 kg/m³. The beam and column cross-section areas, A_b and A_c, respectively, are calculated based on eight design variables of the optimization problem (x), representing the dimensions of the beam and the columns cross-sections of the symmetric frame (see Fig. 1b)), namely: total height d, web thickness t_w and flange width and thickness b_f and t_f, respectively. The cross-section height and the flange width are continuous variables, but the thicknesses are defined by discrete variables, with values limited by the steel plate gauges. The penalty for infeasible designs is given by the sum of the product each violated constraint ($g_i > 1$) and the structural mass given by the maximum value of the design variables.

The optimization problem proposed for the design of the steel frame includes serviceability and ultimate limit states checks as constraint functions. Equations 2 gives the constraint on shear strength, while eq. 3 represents the limit for the interaction of bending and compression. The terms V_{sd}, N_{sd} and M_{sdx} represent the required shear strength, compression strength and flexural strength for bending about the cross-section major axis, respectively. The subscript Rd indicate the corresponding available strength, which are calculated according to the design criteria given by NBR8800:2008 [4]. These constraint functions are evaluated on beams and column for each ultimate load combination and the maximum value of each function is used for the computation of the penalty in eq.1.

$$g_1 = \frac{V_{sd}}{V_{Rd}} \leq 1.0, \tag{1}$$

$$g_2 = \frac{N_{sd}}{N_{Rd}} + \frac{M_{sdx}}{M_{Rdx}} \leq 1.0. \tag{3}$$

Serviceability limit states are including in the optimal deign formulation by the constraints functions g3 and g4, given in eq. 4 and 5, respectively. These constraint functions reflect the limitation on lateral δ_x and vertical displacements δ_z of the portal frame eave and apex, indicated by points P1 and P2 in Fig.1, respectively. The displacements are obtained by structural analysis for serviceability load combinations and the maximum admissible displacements (δ_{xlimt} and δ_{zlimt}) may be assigned for each particular design.

$$g_3 = \frac{\delta x}{\delta x_{limt}} \leq 1.0, \tag{4}$$

$$g_4 = \frac{\delta z}{\delta z_{limt}} \leq 1.0. \tag{5}$$

Finally, the flange width of the beam cross-section b_{fb} is enforced to not exceed the flange width of the column cross-section b_{fb} by the inclusion of the following constructional constraint function

$$g_5 = \frac{bf_b}{bf_c} \leq 1.0. \tag{6}$$

The formulation given by eq. 1–6 represents a nonlinear constrained optimization problem with mixed integer design variables and non-smooth constraint functions. These characteristics makes metaheuristic algorithms more suitable than classical gradient based methods for solving the optimization problems.

2.1 Genetic Algorithm

Developed by John Holland and collaborators in the 1960s, the genetic algorithm (GA) is a metaheuristic optimization method based on Charles Darwin's theory of evolution. It is an algorithm that combines evolutionary strategies using three genetic operators, chromosome crossover, recombination and gene mutation, to evolve to an improved population (Yang [11]). Traditional genetic algorithm has binary encoding, that is, the design variables are stored in strings of bits, which makes the algorithm effective for solving problems with discrete design variables. On the other hand, there is the real encoding genetic algorithm, where the design variables are stored in decimal encoding, making it more efficient for continuous design variables. Another important desirable feature is that of elite individuals. In this case, the algorithm is able to store in an extra position the best individual of that population. This avoids the inconvenience of losing a good individual from one generation to another due to the genetic operators.

The code used in this work was elaborated by Solgi [12] in Python 3 [2] programming language, and is available in the geneticalgorithm library and distributed by Pypi. The referred code is a real encoding, one that allows the inclusion of elitism. The main parameters and arguments are the usual ones present in this kind of metaheuristic algorithm, such as, maximum number of iterations, the population size, crossover probability, mutation probability, parents-portion and the elitism rate. The genetic algorithm initializes the first iteration forming a population with the chosen number of individuals, in the other iterations the number of individuals generated will be equal to the difference of the population size to the number of remaining “parents” defined by the parents-portion parameter. These new individuals are called “children”. Each “child” is produced by a crossover between two parents, and receives part of the design variables from each parent. In this work, a modified version of the uniform crossover was implemented to improve performance. Each design variable of the child is a random combination of the corresponding values of parents.

The link between the Python 3 [2] code and the commercial software SAP2000 [3] is performed internally to the GA code following CSI API Documentation [13]. This allows the evaluation of the objective function and constraints (structural analysis and constraints verification). The constraints checks and calculations are also performed in the Python 3 [2] environment following ABNT NBR 8800:2008 [4] recommendations.

2.2 Case Study

This section presents an example of the application of the optimal design strategy proposed in this work. The symmetric single-span portal frame considered for this study was based on the model introduced by Pravia et al. [14]. The welded I-shaped beam and columns are made of ASTM A572 G50 steel and main dimensions shown in Fig. 1 are: span length Ls = 15 m, eaves height Lc = 6 m and pitch roof θ_p=10º. The shed is 54 m long with 6 m between each portal frame. Purlins with X plan bracing provide lateral restraint for the rafters, avoiding failure by lateral buckling, but the lateral or torsional buckling of columns may take place, since no lateral restraints are provided along the length of these elements. Besides the structure self-weight, which is automatically computed based on the design variables, a dead load of 0.25 kN/m², and service loads of 0.25 kN/m² acting on the roof are considered.

The wind load is determined according to the ABNT NBR6123:1988 [5] considering São Paulo (Brazil) region, with characteristic wind speed of 40 m/s, topography factor S1 of 1.0, probabilistic factor S3 of 0.95 and geometry factor S2 calculated at each 3 meters for a terrain of category 3 and class C building. The distributions values of the pressure’s coefficients are shown in Fig. 2 for wind direction parallel and perpendicular to the frame.

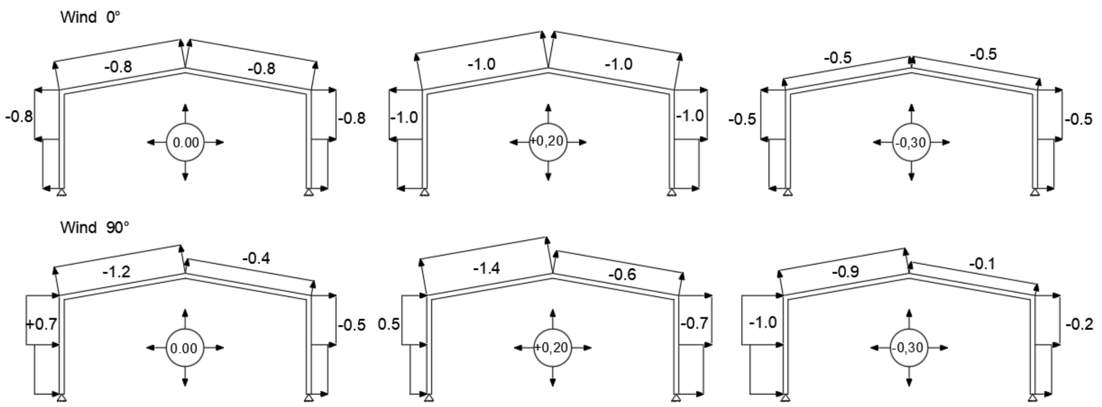

Figure 2 Wind pressure's coefficients

The load combinations used for the evaluation of the ULS or SLS given by the constraint functions g_1 and g_2 or g_3 g_4, respectively, are defined in Tab. 1. A notional force acting on the eave (see point P1 in Fig. 1a)) with magnitude 0.3% of all gravity loads, is included in the combination ULS-1 to account for geometric imperfections.

Table 1. Load Combinations

ULS – 1	1.25 (Self-Weight + Dead) + 1.5 (Service) + 1.4 Notional
ULS – 2	1.25 (Self-weight + Dead) + 1.5x0.8 (Service) + 1.40 (Wind)
ULS – 3	1.0 (Self-weight + Dead) + 1.40 (Wind)
SLS – 1	Self-weight + Dead + 0.70 (Service)
SLS – 2	Self-weight + Dead + 0.60 (Service) + 0.30 (Wind)
SLS – 3	Self-Weight + Dead + 0.30 (Wind)

2.3 Optimization parameters and results

The evaluation of the performance of the strategy for optimal design of the steel portal frame proposed in this paper is conducted by three test cases, designated by cases A, B and C. The evaluation of each case was performed based on 10 independent runs with stopping criteria given by the maximum number of iterations and the following GA parameters: population size equal to 16; mutation probability equal to 10%; uniform crossover with probability equal to 85%; portion of parents equal to 50%; and elitism rate equal to 10%. Table 2 gives the design variables range (or admissible discrete values) adopted for each case.

Case A is the taken as the reference optimization test with 80 iterations, which provided a reasonable average processing time of 40 minutes in a current laptop. The design variables range for this case were defined based on the dimensions of standard I-shaped welded profiles. Case B was included to investigate the possible improvement of the solution with the increase of the computational cost. The maximum number of iterations was increased to 160 in this case, with all other parameters unchanged. Finally, case C is introduced to evaluate the benefit of reducing the range of the design variables based on engineering judgment. In case C, all parameters are kept equal to case A, except those given in Tab. 2, which were defined according to the recommendations for sheds without an overhead crane given by Bellei [15]. The column and beam cross-section height (d_c and d_b) where taken in the range $L_c/30 \leq d_c \leq L_c/20$ and $L_s/70 \leq d_b \leq L_s/50$, respectively. Based on previous experience, it is known that thinner profiles are lighter, and that the thickness of the web is usually less than or equal to the thickness of the flange, thus, the largest thicknesses of steel plate were removed from the list of possible values. So, continuous variables d and bf turn out to have lower boundary of 15 and 10 cm, respectively, and an upper boundary of 40 cm. The discrete variable tw is limited to a maximum thickness of 0.8 cm, and the variable tf is limited to 1.60 cm, as indicated in Tab. 2.

Table 2. Design variables range

Design variable	Case A and B (cm)	Case C (cm)	Type
d	[15, 100]	[15, 40]	Continuous
t_w	{0.475, 0.63, 0.8, 0.95, 1.25, 1.6, 1.9, 2.24, 2.5}	{0.475, 0.63, 0.8}	Discrete
t_f	{0.475, 0.63, 0.8, 0.95, 1.25, 1.6, 1.9, 2.24, 2.5}	{0.475, 0.63, 0.8, 0.95, 1.25, 1.6}	Discrete
b_f	[10, 70]	[10, 40]	Continuous

The best design solution was obtained in the case B, resulting in a total structural mass of 729.30 kg with the design variables values given in Tab. 3. All runs in the three cases resulted in feasible solutions. The constraints related to the lateral displacement, obtained with load case SLS-2 and wind internal pressure coefficient equal to -0.30, was active or nearly active in all optimization solutions, being the decisive factor for the sizing of the cross-sections. The constraint associated with vertical displacement had no significant influence on the best design obtained by the optimizations, but the limit on combined compression and flexure (evaluated by constrain function g2) was important in some cases.

Table 1. Optimal solution

Design variable	Beam cross-section (cm)	Column cross-section (cm)
d	29.64	50.33
tw	0.475	0.475
tf	0.63	0.475
bf	10.84	20.58

If we use this solution as the best solution known of the optimization problem, we can compare the efficiency of the cases A, B and C. So that, it is necessary to know some statistics values of each case, like the best and worst results, the mean values, standard deviation and coefficient variation. All these statistic parameters are shown in Tab. 4.

Table 4. Objective function statistics values

Objective function statistics	Case A	Case B	Case C
Maximum value	1231.14 kg	984.87 kg	1000.33 kg
Minimum value	788.33 kg	729.65 kg	768.64 kg
Mean value	988.05 kg	819.83 kg	833.59 kg
Standard deviation	120.41 kg	78.61 kg	67.25 kg
Coefficient variation	12.20 %	9.60 %	8.10 %

It is interesting to notice Pravia et. al. [14] utilizes the same steel profile for the beam and column, with mass equal to 38,7 kg/m, what results an approximate total mass of the portal frame equal to 1048 kg, therefore the case A has the tendency in finding worse results than non-optimized solutions. So, the case A is not good enough, and to solve this problem the genetic algorithm needs more time or a reduced search space. The increase of iterations, case B, and the reduce of range of design variables, case C, proved be able to find out better solutions and to reduce the variation between the results.

In order to better analyze and compare the tests cases, it was evaluated the 95% confidence intervals for each test case, it means that there is a 95% probability of the result, on an independent run, falls within the interval range. The confidence intervals of the three tests cases are $A = [901.93\ to\ 1074.18]\ kg$, $B = B = [763.60\ to\ 876.07]\ kg$, $C = [785.49\ to\ 881.70]\ kg$. Using a hypothesis test it was possible to statistically check the relations of the mean values of objective function. A T-test comparison of sample means, firstly assuming a null hypothesis of H_0: $\overline{f}_A = \overline{f}_B$ and secondly assuming a null hypothesis of H_0: $\overline{f}_A < \overline{f}_B$, resulted in the values presented in Tab. 4.

Table 4. Statistic results.

Compares Cases	Tests	H_0	Result	H_0	Result
A – B		$\bar{f}_A = \bar{f}_B$	Reject H_0	$\bar{f}_A < \bar{f}_B$	Reject H_0
A – C		$\bar{f}_A = \bar{f}_C$	Reject H_0	$\bar{f}_A < \bar{f}_C$	Reject H_0
B – C		$\bar{f}_B = \bar{f}_C$	Accept H_0	$\bar{f}_B < \bar{f}_C$	Accept H_0

In addition, the database includes the relationship between the processing time in SAP2000 [3] software and the Python 3 [2] 3, and it was discovered that the analyses evaluated in SAP2000 [3] represents 99.7% of all the optimization time. As the final analysis it was compared the sizes for of the column and beam. Search the solution list, the cross-section parameters of the columns used to be bigger or equal to the corresponding values of beams. The design variables values find out for the column, Tab. 3, in the optimal solution at test case B are not included in the design variables range determined at test case C, it means that practical engineering recommendations may miss the optimal solution. The graphic in Fig. 3 shows the area in cm² of the column and beam in relationship to objective function, where it is possible to see that the column is bigger than the beam in almost all the 30 optimizations, the indices A, B and C correspond to the cases A, B and C.

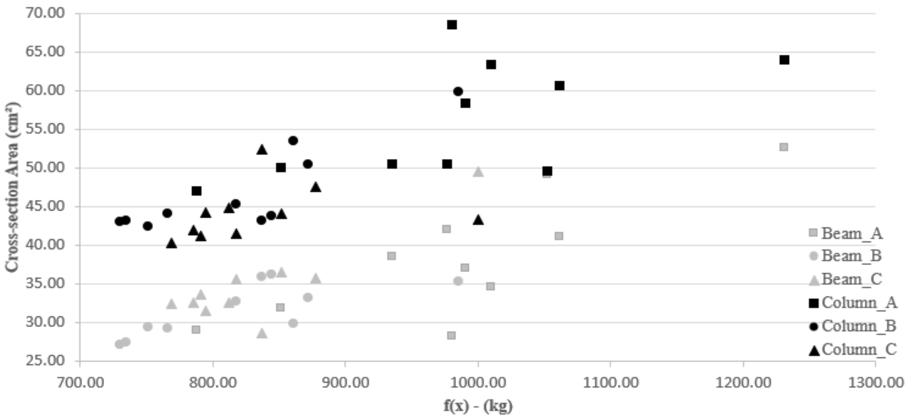

Figure 3. Beam and column cross-section area for the optimal designs in cases A, B and C.

3 Conclusions

The application of the genetic algorithm (GA) as a tool for the optimal design of the structure proved that the stochastic optimization methods combined with programming languages are efficient for solving non-linear problems and non-smooth functions. According to the results presented, it is possible to conclude that in all tests cases the algorithm was successful in satisfying the constraints, find a viable solution. However, for case A, the genetic algorithm obtained very dispersed and poor-quality results, making the test case less efficient. According to the results in Tab. 4, it can be said that cases B and C have lower average objective function than case A, and it does allow assert test case B has the same average than case C. With this information and with the standard deviation values, it is clear that increasing the number of iterations and reducing the feasible search space improves the performance of the algorithm. The use of practical engineering recommendation, used in test case C, improved the convergence of results, reducing randomness, but considering that the best solution found of 729.30 kg has a column with height $d = 50.33$ cm, the case C eliminated the possibility in optimal solutions. So, for the frame model studied in this work, the best solution is to use the increment of the number of iterations to assure the final result are indeed the best ones.

As far as concerned in this study, the genetic algorithm proves to be able to reach a feasible local optimal solution, but it is strongly dependent on the number of iterations. While the increase in computational processing time helps to obtain good results, it can become a limiting factor for the practical application of engineering designs.

According to Fig. 3, the best results obtained are characterized by columns larger than beams. This happens, because the total length of columns is smaller than the total length of beams, and unlike the beams, that were completely laterally contained, the columns were mostly limited to lateral buckling with torsion. So, it is concluded that the consideration of lateral buckling with torsion is crucial in the design of the industrial building portal frames. The use of thin steel plates provides more material savings, as seen, the optimal reference solution has 4.75 mm and 6.3 mm thickness sheets. The most relevant constraints were obtained by the load combination 2 in the ultimate limit state and the load combination 2 for the lateral displacement of the frame, demonstrating that the wind loads were decisive for the design of the structure.

References

[1] The Steel Construction Institute (SCI). "Best Practice in Steel Construction Industrial Buildings Guidance for Architects, Designers and Constructors". 2008.
[2] Van Rossum, G., & Drake, F. L. (2009). Python 3 Reference Manual. Scotts Valley, CA: CreateSpace.
[3] Computers & Structures, INC. CSI Analysis Reference Manual. ETABS, 2016.
[4] Associação Brasileira de Normas Técnicas. NBR 8800: "Projetos de Estrutura de aço e de estrutura mista de aço e concreto de edifícios". Rio de Janeiro, 2008.
[5] Associação Brasileira de Normas Técnicas. NBR 6123: "Forças devidas ao Vento em edificações". Rio de Janeiro, 1988.
[6] P. C. G. P. Oliveira, G. A. S. Falcon. "Dimensionamento ótimo de pórtico de aço considerando ligações semirrígidas e análise não linear geométrica". Centro Brasileiro da Construção em Aço (CBCA). Revista da Estrutura de Aço (REA), 2020.
[7] MATLAB. (2010). version 7.10.0 (R2010a). Natick, Massachusetts: The MathWorks Inc.
[8] ANSYS. Finite Element Program User's Manuals, ANSYS, Inc., Canonsburg, United States, 2015.
[9] D. T. Phan, J. B. P. Lim, T. T. Tanyimboh, R. M. Lawson, Y. Xu, S. Martin, W. Sha. "Effect of serviceability limits on optimal design of steel portal frames". Journal of Constructional Steel Research 86, 2013.
[10] R. Mckinstray, J. B. P. Lim, T. T. Tanyimboh, D. T. Phan, W. Sha. "Optimal design of long-span steel portal frames using fabricated beams". Journal of Constructional Steel Research 104, 2015.
[11] X. S. Yang. "Engineering Optimization. "An Introduction with Metaheuristic Applications". University of Cambridge. John Wiley & Sons, Inc., Hoboken, New Jersey, 2010.
[12] R. M. Solgi. geneticalgorithm · PyPI, 2020.
[13] Computers & Sructures, Inc. CSI API Documentation. ETABS, 2018.
[14] Z. M. C. Pravia, G. A. Drehmer, E. M. Júnior. "Galpões para usos gerais". Instituto Aço Brasil. Rio de Janeiro: IABr/CBCA, 2010.
[15] I. Bellei. "Edifícios Industriais em aço". 5ª Edição. São Paulo, Editora Pini, 2006.