THE PUBLISHING HOUSE
OF THE ROMANIAN ACADEMY

PROCEEDINGS OF THE ROMANIAN ACADEMY, Series A,
Volume 15, Number 2/20, pp. 165–173

OPTIMAL DESIGN OF TRUSS STRUCTURES WITH FREQUENCY CONSTRAINTS USING INTERIOR POINT TRUST REGION METHOD

Zhenglei YU, Tao XU, Peng CHENG, Wenjie ZUO, Xue LIU, Tatsuo YOSHINO

Jilin University, College of Mechanical Science & Engineering, Changchun 130025, PR China
Corresponding author: Tatsuo Yoshino, E-mail: yuzl11@mails.jlu.edu.cn

Nowadays, in the industrial environment of simulation-based design, optimization has been widely applied. This article illustrates that how truss structural frequency optimization problems is being solved by interior point trust region method (IPTRM). As a possible way, primal-dual interior point method could optimize the structure by multiple centrality modifications with trust region sub-problems. Through these series of correction, the algorithm's convergence has been improved progressively. Moreover, IPTRM verifies the classical truss structures with the conclusion that it is more reliable and efficiency than any other algorithm when meet the certainly accuracy level, especially, when time is the constraint, IPTRM seems to be the superior solution for large optimization problems.

Key words: Interior point trust region method (IPTRM), Structural optimization, Frequency optimization.

1. INTRODUCTION

In various structural design and optimization problems it is necessary to repeatedly solve the response of the modified structures. For each intermediate structure the implicit analysis equations must be solved and the multiple repeated analyses usually involve extensively computational effort. A variety of numerical methods have been proposed in literatures in purpose to deal with the complexity and nonlinearity of structural optimization problems. As well known, an optimization problem is defined by an objective function and different constraint equations and a common analytical form is the following:

$$\begin{cases} \min\ f(x_1, x_2, \cdots, x_N) \\ \text{s.t.}\ \ g_k(x_1, x_2, \cdots, x_N) \geq 0 \\ \qquad x_j^l \leq x_j \leq x_j^u \end{cases} \begin{cases} j = 1, \cdots, N \\ k = 1, \cdots, NC \end{cases}, \tag{1}$$

where $(x_1, x_2, \cdots, x_N)$ are the N design variables; $f(x_1, x_2, \cdots, x_N)$ is the objective function; $g_k(x_1, x_2, \cdots, x_N)$ are the NC inequality constraint functions; x_j^l and x_j^u are the lower and upper bounds of the j^{th} design variable.

Despite of the conceptual simplicity of the method, the SLP techniques are not globally convergent [1] and some problems, like convergence to a local or an unfeasible optimum or objective function oscillations, could arise if the method is not well controlled during successive iterations [2]. In practical design, sequential linear programming (SLP) is very popular because of its inherent simplicity and easy availability. In this paper we present an approach to managing the use of approximation models in optimization that is based on the trust region and interior point methods approach from nonlinear programming.

As the origin of the trust region methods (TRM), the work by Levenberg [3] in 1944 were the mostly cited among the papers related to TRM. And a more direct link to TRM was the work by Marquardt in 1963 [4]. Different authors proposed move limits definition criteria for the Sequential Linear Programming method. Schittowsky proved that the SLP technique is very efficient if compared to other methods, from a numerical point of view, when a proper choice of the move limits was adopted [5]. The convergence and regularity of the standard trust region method when applying it to ill-posed problems had also been studied by Wang and Yuan [6].

Interior-point (IP) methods were originally developed for linear optimization as an alternative algorithm to the effective simplex method [7]. IP methods were particularly attractive for problems where a large number of constraints were present. Contrary to the simple method, which searches for the optimum by moving along the constraints' boundaries, the linear IP algorithms force the designs to follow a central path until convergence was achieved [8, 9]. In general, these algorithms do not generate feasible designs at each iteration. Instead, interiority is achieved with respect to the slack and surplus variables used for the inequality constraints.

The trust region framework gives an adaptive method for managing the amount of optimization done with the approximation models before one has recourse to a detailed model to check the validity of the design generated by the approximation model [10]. Perez [11] detailed the development of an interior-point approach for trust region managed sequential approximate optimization. The interior-point approach will ensure that approximate feasibility is maintained throughout the optimization process.

The paper is organized in the following sequence. In Sect. 2 the proposed interior-point trust-region method is presented with a detailed description of the algorithm. Test structural optimization problems and results are presented in Sect. 3 and some final remarks are presented in Sect. 4.

2. INTERIOR POINT TRUST-REGION METHODOLOGY (IPTRM)

In this section, an approximate interior point trust region based SLP algorithm is presented. We use a trust region interior point algorithm to solve the problems [12, 13]. It is the trust region that controls the linear step size and ensures the validity of the linear model.

Generally, many weapons may be utilized to check whether the approximation used in the current iteration is good. Here, IPTRM defines two functions: the linearization error $\tau_{\mathrm{LIM}}(X)$ and the trust region parameter $\rho(X)$. The former quantity evaluates the difference between the actual and approximate models while the latter serves to shrink design space which might yield no significant improvement in cost.

2.1. Linearization error problem

Expressions (1) show that optimization problems usually have very complex and highly non-linear implicit formulations. Therefore, the SLP method is a popular approach to deal with the problems. The original non-linear problem is replaced with a linear problem, build using linear approximations of the objective function and constraints. The original non-linear problem changes into the following linear one:

$$\begin{cases} \min\ f(\mathrm{X}_o^i) + \sum_{j=1}^{N} \frac{\partial f}{\partial x_j}\Big|_{P_o^i} (x_j - x_{o,j}^i) \\ \mathrm{s.t.}\ \ g_k(\mathrm{X}_o^i) + \sum_{j=1}^{N} \frac{\partial g_k}{\partial x_j}\Big|_{P_o^i} (x_j - x_{o,j}^i) \geq 0 \quad \begin{cases} j = 1, \cdots, N \\ k = 1, \cdots, NC \end{cases} \\ x_j^1 \leq x_j \leq x_j^u\ . \end{cases} \tag{2}$$

The $\mathrm{X}_0^i(x_{0,1}^i, x_{0,2}^i, \cdots, x_{0,N}^i)$ vector defines the linearization point P_0^i in the design space while the superscript *i* refers to the current optimization cycle. The linearization error $\tau_{\mathrm{LIN}}(X)$ at X is computed as follows:

$$\tau_{\mathrm{LIN}}(X) = \underset{k=1,\ldots,NC_{act}}{\mathrm{Max}} \left(\left| \frac{f_{\mathrm{NLIN}} - f_{\mathrm{LIN}}}{f_{\mathrm{NLIN}}} \right|, \left| \frac{g_{\mathrm{NLIN}}^k - g_{\mathrm{LIN}}^k}{g_{\mathrm{NLIN}}^k} \right| \right), \tag{3}$$

where NC_{act} is the number of active constraints in the current iteration; f_{NLIN} and f_{LIN}, respectively, are the values of the original and linear objective function, evaluated at any generic point X of the design space distinct from the linearization point P_0^i; g_{NLIN}^k and g_{LIN}^k, respectively, are the original and linear k th active constraints, evaluated at X. In the every iteration, the linearization error must satisfy the function as follows:

$$\tau^{i}_{LIN}(X) \le \tau^{LIM\,i}_{LIN} \,. \tag{4}$$

The $\tau^{i}_{\mathrm{LIN}}(X)$ and $\tau^{i}_{\mathrm{LIN}}(X)$ terms, respectively, indicate the linearization error at an arbitrary point X of the design space and the linearization error is limited in the current iteration.

The initial value of $\tau_{\mathrm{LIN}}(X)$ is chosen in the range of 1-10%, which requires the user to specify the allowable value. Such values are suitable for weight minimization of truss structures under displacement, stress and buckling constraints. Hence, care should be taken in updating the value of $\tau_{\mathrm{LIN}}(X)$ at the end of each design cycle.

2.2. Trust region methodology

In this research the move limits will be managed by a trust region approach. Trust region managed frameworks for optimization have been widely adopted due to their provable convergence properties. In the every iteration, the trust region approach manages the local variable bounds based on the performance of a merit function.

It is clear that the trust region might be very sensitive to the value of $\tau^{LIM\,i}_{LIN}$. Of course, this fact is not desirable, especially in the first optimization cycle. For this reason, the trust region approach is based on the use of a trust region ratio ρ to check whether the value of $\tau^{LIM\,i}_{LIN}$ is suitable or not. The trust region parameter function is defined as follows:

$$\rho^{i} = \frac{f(x^{i}) - f(x^{i+1}_{*})}{\tilde{f}^{\,i}(x^{i}) - \tilde{f}^{\,i}(x^{i+1}_{*})}, \tag{5}$$

where $\tilde{f}^{\,i}$ is an approximation to the function f around the point x^{i} at the i th iteration. The quantity ρ is simply the ratio of the actual change in the function to the change predicted by the approximation. The closer the value of ρ is to 1, the better the approximation $\tilde{f}^{\,i}$ mimics the behavior in the descent direction of f. The function (5) will be satisfied if the trust region radius Δ is sufficiently small.

If one step is accepted then the trust region is increased as follows:

$$\Delta^{i+1} = \begin{cases} c_1\Delta^{i} & \text{if } \rho^{i} < 0.25 \\ \min\{c_2\Delta^{i}, \overline{\Delta}\} & \text{if } \rho^{i} > 0.75 \\ \Delta^{i} & \text{otherwise.} \end{cases} \tag{6}$$

The trust region multiplication factors c_1 and c_2 are commonly set to be 0.25 and 2 respectively. $\overline{\Delta}$ is the maximum value of region radius.

2.3. Trust region interior point algorithm

The algorithm assumes that the starting point is a (surrogate) feasible design. This assumption will help to develop the general framework.

The main idea of the algorithm is simple. If a (surrogate) feasible starting point is given, the algorithm should give a (surrogate) feasible point in each iteration. Though this sounds like a simple requirement to comply with, it is a very tough problem for general nonlinear optimizers. Instead, approximations of the objective function and constraints are built, and then an optimization is performed over the approximations. The step size depends on the quality of the approximations.

The main ingredients of the algorithm have been discussed and the overall procedure for solving structural optimization problems can be described as follows:

(A): Initialize parameters. Set the trust region radius Δ_0, τ_0, ρ_0.

(B): A starting point x^{0} that satisfies (2) can be obtained.

(C): Solve the trust region sub-problem to obtain linear step size Δ^i. The value of x is updated, $\tilde{x}^{i+1} = x^i + \Delta x^i$. The problem is executed at the updated operating point $\tilde{x}^{i+1}$. Thus a new updated point x^{i+1} that satisfies equality constraints strictly is obtained.

(D): If convergence is achieved: $\|x^{i+1} - x^i\| \leq \varepsilon$, then stop, otherwise go on to Step (E).

(E): Compute ratio ρ. If $\tilde{f}^i(x^i) - \tilde{f}^i(x_*^{i+1}) > 0$, then go on to Step (F). Otherwise go on to Step (G).

(F): Update the trust region radius Δ^{i+1} with (6).

(G): Let $x^{i+1} + \Delta x^i$, $\Delta^{i+1} = \Delta^i / 2$. Set i=i+1. Go to Step (C).

2.4. Optimization model for truss structure

For truss structures in this paper, a design optimization problem usually aims at seeking a least-weight structure while satisfying all the stress and displacement constraints. Mathematically, it may be stated as

$$\text{Min} \quad f(\text{X}) = \sum_{i=1}^{N_e} x_i L_i \rho_i \tag{7}$$

and constraints are chosen to be natural frequencies of trial structures:

$$g_i(\text{X}) = \frac{\omega_i}{\omega_{all}} - 1 \geq 0, \quad i = 1, 2, \ldots, m, \tag{8}$$

where ρ_i and L_i are weight of unit volume and length of i th element, respectively. N_e is the number of the structural elements. ω_i and ω_{all} are the i^{th} frequency and allowable frequency, respectively.

The main concepts of the algorithm are briefly explained in Section 2.3. The flow chart of the IPTRM truss structural optimization is shown in Fig. 1.

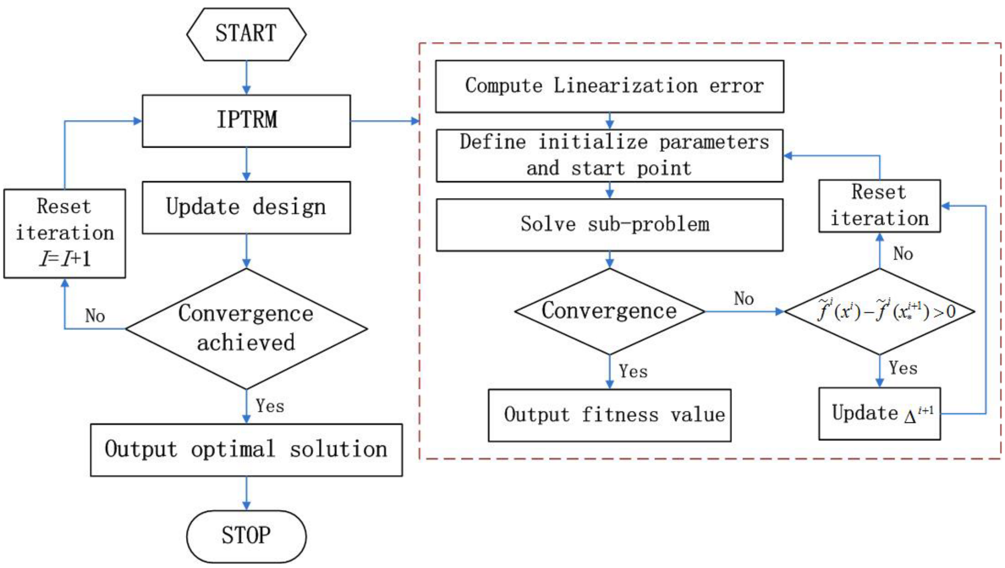

Fig. 1 – Flowchart of the IPTRIM optimization.

3. NUMERICAL RESULTS AND DISCUSSIONS

Above IPTRM optimization method is programmed and embedded into our software platform [14], which is developed in VB.NET framework. In this section, three examples are static design of structural frequency optimization of trusses and discussed to verify the proposed optimization described before [15-

18]. These examples are all carried out on a 2.8 GHz Pentium (R) Dual-Core E5500 server with 2 GB memory, running under WindowsXP. The value E_p = 0.001 used in the numerical examples is suggested according to reference [12].

3.1. Example 1: 10-bar truss

The 10-bar truss shown in Fig. 2 is a classical example in structural frequency optimization, where l = 914.4 cm (360 in) and P = 45.400 kg (105 lb). The material density is 2,770 kg/m^3 and the modulus of elasticity is 69.8 GPa. All the lower bounds for all cross-sectional areas are 0.65 cm^2. The value 454 kg of concentrated masses are added in node 2, 3, 4 and 5, respectively. Frequency constraints are $\omega_1 \geq 7\,\text{Hz}$, $\omega_2 \geq 15\,\text{Hz}$, $\omega_3 \geq 20\,\text{Hz}$.

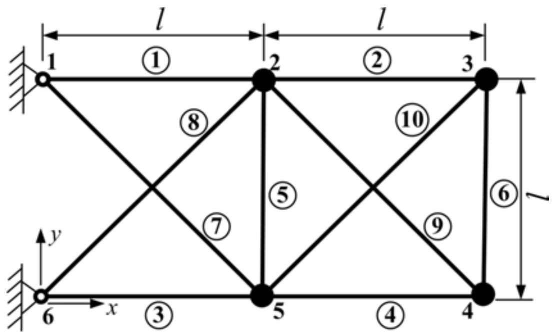

Fig. 2 – Structure of 10-bar truss.

The results are compared in Table 1 and Table 2. The methods achieved the good solution as those reported in references. The optimal weight is 544.7 kg, which is well coincide with the references [15-18] values. The first three frequency constraints are all satisfied. The results prove that the IPTRM is highly accurate and practical for truss optimization. However, the computing time of IPTRM is less prominent in Table 2. This because it consumes lots of time in each iteration.

Table 1

Optimal designs of the natural frequencies from various methods

Frequency No.	Wang [15]	Grandhi [16]	Sedaghati [17]	Wei [18]	SLP	D-SLP [19]	IPTRM
1	7.011	7.059	6.992	7.008	7.022	7.003	7.001
2	17.302	15.895	17.599	18.148	16.376	16.543	18.077
3	20.001	20.425	19.973	20.000	20.199	20.154	20.025

Table 2

Optimal designs comparison for the 10- bar truss

Member No.	Optimal cross-sectional area (cm^2)						
	Wang [15]	Grandhi [16]	Sedaghati [17]	Wei [18]	SLP	D-SLP [19]	IPTRM
1	32.456	36.584	38.245	42.234	31.836	35.148	36.380
2	16.577	24.658	9.916	18.555	14.080	13.169	12.941
3	32.456	36.584	38.619	38.851	30.835	37.690	35.764
4	16.577	24.658	18.232	11.222	14.589	19.556	18.314
5	2.115	4.167	4.419	1.783	1.234	1.087	3.002
6	4.467	2.070	4.419	4.451	4.844	4.844	5.433
7	22.810	27.032	20.097	21.049	23.663	18.314	20.989
8	22.810	27.032	24.097	20.949	26.381	27.415	24.140
9	17.490	10.346	13.890	10.257	19.848	12.562	9.753
10	17.490	10.346	11.452	14.342	14.231	12.106	18.102
Weight (kg)	553.8	594.0	537.0	542.75	548.08	534.57	544.7
CPU time (s)	0.56	0.67	0.58	0.55	0.52	0.57	0.63

3.2. Example 2: 72-bar truss

The second example considers the weight minimization of a 72-bar truss shown in Fig. 3. The structure is designed 16 groups of the truss elements (Table 3). Material properties are: E = 6.86 GPa and ρ = 2 770 kg/m^3. The natural frequencies of truss are $\omega_1 > 4\text{Hz}$ and $\omega_2 > 6\text{Hz}$. The lower and upper bounds for all cross-sectional areas are 0.64 and 10 cm^2, respectively. The value 2 270 kg of concentrated masses are added from node 1 to node 4, respectively. The results are compared in Table 4. Clearly, the IPTR method yield excellent solutions and not violated frequency constraints. The optimize computational results by the IPTRM are similar same with references [20, 21]. The strategy and precision control of the IPTR method is accurate. Compare with the computation times, the effectiveness of the IPTRM is almost same with each method.

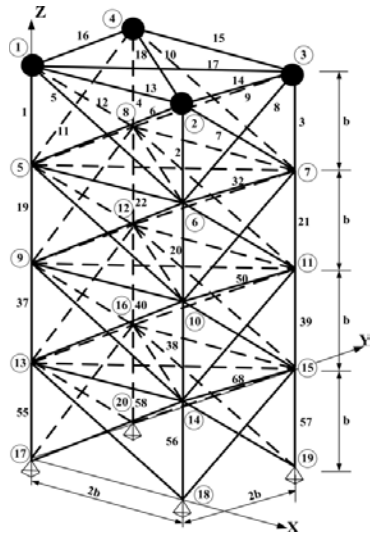

Fig. 3 – A sketch of a 72-bar truss.

Table 3

Groups of the 72-bar truss elements

Group	Elements	Group	Elements
1	1-4	9	37-40
2	5-12	10	41-48
3	13-16	11	49-52
4	17-18	12	53-54
5	19-22	13	55-58
6	23-30	14	59-66
7	31-34	15	67-70
8	35-36	16	71-72

Table 4

Optimum designs of the 72-bar truss

Variable No.	Optimum design (cm^2)		
	Dede [20]	Salajegheh [21]	IPTRM
1	3.500	3.169	2.667
2	7.900	10.302	10.540
3	0.645	0.686	0.650
4	0.645	0.768	0.724
5	13.000	14.553	11.021
6	7.900	6.598	6.432
7	0.645	0.753	0.810
8	0.645	1.012	0.945
9	12.500	12.036	11.313
10	7.900	7.684	8.48
11	0.645	0.856	0.733
12	0.645	0.718	0.642
13	13.000	13.054	11.217
14	7.800	6.846	6.984
15	0.645	0.718	0.738
16	0.645	0.979	0.851
Weight (kg)	324.67	325.7	322.67
CPU time (min)	8.4	8.2	7.9

3.3. Example 3: 200-bar truss

The third example considers the weight minimization of a 200-bar truss structure shown in Fig. 4, which is a classical example in structural optimization with 77 nodes. However, the structure is rarely used as the frequency optimization. The structure is designed 29 groups of the 200 truss elements. Material properties are: E = 210 GPa and ρ = 7 800 kg/m^3. The natural frequencies of truss are $\omega_1 \geq 5\text{Hz}$, $\omega_2 \geq 10\text{Hz}$ and $\omega_3 \geq 15\text{Hz}$. The lower and upper bounds for all cross-sectional areas are 1.0 and 10 cm^2, respectively.

The results are also listed in Table 5. The results prove that both of the reanalysis method [22], D-SLP [19] and IPTRM have good results. However, the IPTRM, D-SLP and reanalysis method were costed 45.8, 92.5 and 103.4 mins, respectively. Speed-up ratio of computational time reaches 92.5/45.8 = 2.01 and 103.4/45.8 = 2.26, which is much more efficient than D-SLP and reanalysis method. Short compared with the IPTRM, while the each iteration of the calculation to complete a very heavy task of solving equations. But the solution can be achieved remarkable progress in reaching. However, the D-SLP method may be iterative to complete a very heavy task, and the calculation of each iteration takes less than. Therefor, for large scale problems IPTRM will be beyond the methods in speed, as the result of the third example.

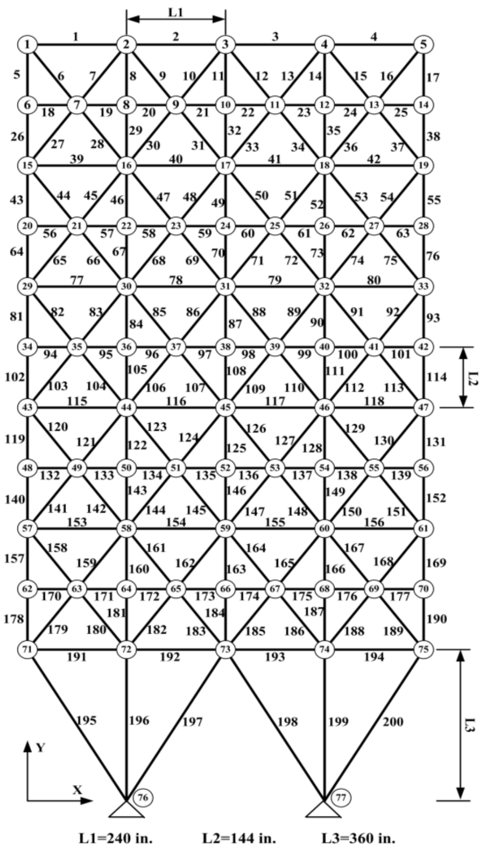

Fig. 4 – Structure of 200-bar truss.

Table 5

Optimal design comparisons for the 200-bar truss

Group	Optimal cross-sectional area (cm^2)			
	Members	Reanalysis [22]	D-SLP [19]	IPTRM
1	1, 2, 3, 4	1.2362	1.1654	1.6378
2	5, 8, 11, 14, 17	1.5984	1.5906	1.1181
3	19, 20, 21, 22, 23, 24	1.4991	1.0176	1.1937
4	18, 25, 56, 63, 94, 101, 132, 139, 170, 177	1.1185	1.0157	1.0394
5	26, 29, 32, 35, 38	1.0558	1.2677	1.1102
6	6, 7, 9, 10, 12, 13, 15, 16, 27, 28, 30, 31, 33, 34, 36, 37	1.0000	1.0157	1.0630

Tabel 5 (continued)

7	39, 40, 41, 42	1.3781	1.1181	1.2913
8	43, 46, 49, 52, 55	1.3622	1.0630	1.0709
9	57, 58, 59, 60, 61, 62	1.0080	1.0709	1.2756
10	64, 67, 70, 73, 76	1.2443	1.3701	1.0315
11	44, 45, 47, 48, 50, 51, 53, 54, 65, 66, 68, 69, 71, 72, 74, 75	1.0870	1.0000	1.0787
12	77, 78, 79, 80	1.3700	1.1102	1.0315
13	81, 84, 87, 90, 93	1.3614	1.0471	1.1176
14	95, 96, 97, 98, 99, 100	1.0000	1.0000	1.0000
15	102, 105, 108, 111, 114	1.9212	1.3465	1.5118
16	82, 83, 85, 86, 88, 89, 91 92, 103, 104, 106, 107, 109, 110, 112, 113	1.1024	1.0866	1.0472
17	115, 116, 117,118	1.0318	1.0496	1.4094
18	119, 122, 125, 128, 131	1.0386	1.3922	1.4471
19	133, 134, 135, 136, 137, 138	1.0589	1.5930	2.5734
20	140, 143, 146, 149, 152	1.3000	1.5354	1.9213
21	120, 121, 123,124, 126, 127, 129, 130, 141, 142, 144, 145, 147, 148, 150, 151	1.7452	1.5882	1.2745
22	153, 154, 155,156	7.8424	2.5161	2.3675
23	157, 160, 163, 166, 169	9.0277	9.8275	9.7804
24	171, 172, 173, 174, 175, 176	3.2000	1.6106	1.3836
25	178, 181, 184, 187, 190	8.2039	8.6039	8.1020
26	158, 159, 161, 162, 164, 165, 167, 168, 179, 180, 182, 183, 185, 186, 188, 189	9.1452	8.3216	8.1569
27	191, 192, 193, 194	9.4591	9.9059	8.2980
28	195, 197, 198, 200	9.3181	9.5765	9.8510
29	196, 199	9.9527	9.8824	9.9608
Weight (Kg)		306.07	292.68	289.11
CPU Time (min)		103.4	92.5	45.8

In the following, we discuss the influence of linearization error ($\tau_{\text{LIM}}(X)^{\text{init}}$) on the truss optimal. The optimal weight and CPU time of 200-bar truss for different initial value of linearization error is list in Table 6. It is observed that initial value of allowable linearization error has large effect on optimal solution and CPU time during the solving process. The number of optimization CPU times changed marginally. In the 200-bar truss problem, increasing the initial value of allowable linearization error resulted in more times: respectively, 13% and 33% for $\tau_{\text{LIM}}(X)^{\text{init}} = 0.5$ and $\tau_{\text{LIM}}(X)^{\text{init}} = 1.0$. This is because increasing $\tau_{\text{LIM}}(X)^{\text{init}}$ results in large perturbations of design in the early optimization cycles. Since IPTRM checks if the changes in design variables are too large, the approximation may be judged no longer accurate at the new design point.

Finally, since Table 6 also shows the optimal weight of various initial values of allowable linearization error approximately the same, the speed reducer was optimized for each starting design. However, the overall performance of IPTRM did not depend on the input data.

Table 6

Optimal weight and CPU time of 200-bar truss for different linearization error

No.	$\tau_{\text{LIM}}(X)^{\text{init}}$	Optimal weight (kg)	CPU time (min)	Frequency no.		
				1	2	3
1	0.2	289.87	45.8	5.272	13.17	19.83
2	0.5	287.25	50.3	5.151	12.94	19.09
3	1.0	292.11	60.2	5.730	14.15	20.97

4. CONCLUSION

In this paper, an interior point trust-region methodology (IPTRM) for structural optimization with frequency constraints is presented. Numerical examples applied to optimal design of trusses are given in this paper to verify the accuracy and efficiency of the proposed method. The algorithm may be suitable for large frequency optimization problems where time is a constraint and the optimization may be stopped before convergence is achieved. The accuracy of optimal solution is acceptable and the design process is speeded up, indeed.

The effect of initial linearization error on optimal solution and computational time are also discussed. In a large variety of optimization problems, the IPTRM is more efficiently. Optimum designs are found substantially insensitive to input parameters such as the initial value. From the results it is proved that IPTRM is a controllable, adaptive and approximate optimization method.

ACKNOWLEDGMENTS

This research is supported by the Technology Development of Jilin Province (No. 20126001) and the National Natural Science Foundation of China (No. 51205159).

REFERENCES

1. Arora J.S., Belegundu A.D., *Structural optimization by mathematical programming methods*, AIAA J., **22**, pp. 854–856, 1984.
2. Haftka R.T., Gurdal Z., *Elements of structural optimization*, 3rd ed., Dordrecht, Kluver Academic Publishers, 1992.
3. Levenberg K., A method for the solution of certain nonlinear problems, Quart. Appl. Mat., **2**, pp. 164–168, 1944.
4. Marquardt D.W., *An algorithm for least-squares estimation of nonlinear parameters*, SIAM J. Appl. Math., **11**, pp. 431–441, 1963.
5. Schittowski K., Zillober C., Zotemantel R., *Numerical comparison of nonlinear programming algorithms for structural optimisation,* Structural Optimization, **7**, pp. 128, 1994.
6. Wang Y., Yuan Y., *Convergence and regularity of trust region methods for nonlinear ill-posed inverse problems*, Inverse Problem, **21**, pp. 821–838, 2005.
7. Wright S.J., *Primal-Dual Interior-Point Methods*, PA: SIAM, 1997.
8. Byrd R.H., Hribar M.E., Nocedal J., *An interior point algorithm for large scale nonlinear programming,* SIAM J. Optim, **9**, pp. 877–900, 2000.
9. Vanderbei R.J., Shanno D.F., *An interior point algorithm for nonconvex nonlinear programmin,* Comput. Optim. Appl., **13**, pp. 231–252, 1999.
10. Luciano L., Carmine P., *Move limits definitionin structural optimization with sequential linear programming. Part I: Optimization algorithm*, Computers and Structures, **81**, pp. 197–213, 2003.
11. Pérez V.M., Renaud J.E., Watson L.T., *An interior-point sequential approximate optimization methodology,* Struct Multidisc Optim., **27**, pp. 360–370, 2004.
12. Xu T., Yu Z.L., Xiu H., Xu T.S., Li Y.W., *Structural Optimization of Truss using Interior point and Trust Region*, Information–An International Interdisciplinary Journal, in press, 2013; Metalurgia International, **9**, pp. 81–85, 2013.
13. Rodríguez J.F., Renaud J.E., Watson L.T., *Convergence of trust region augmented Lagrangian methods using variable fidelity approximation data*, Structural Optimization, **15**, pp. 141–156, 1998.
14. Zuo W.J., Li W.W., Xu T., Xuan S.Y., Na J.X., *A complete development process of finite element software for body-in-white structure with semi-rigid beam in .NET framework*, Adv. Eng. Softw., **45**, *1*, pp. 261–271, 2012.
15. Wang D., Zhang W.H., Jiang J.S., *Truss optimization on shape and sizing with frequency constraints,* AIAA Journal, **42**, pp. 622–630, 2004.
16. Grandhi R., *Structural optimization with frequency constraints-a review*, AIAA Journal, **31**, pp. 2296–2303, 1993.
17. Sedaghati R., Suleman A., Tabarrok B., *Structural optimization with frequency constraints using the finite element force method*, AIAA Journal, **40**, pp. 382–388, 2002.
18. Wei L.Y., Zhao M., Wu G.M., Meng G., *Truss optimization on shape and sizing with frequency constraints based on genetic algorithm*, Comput. Mech., **35**, pp. 361–368, 2005.
19. Luciano L., Carmine P., *Move limits definitionin structural optimization with sequential linear programming. Part II: Numerical examples*, Computers and Structures, **81**, pp. 215–238, 2003.
20. Dede E.M., Hulbert G.M., *Analysis, design and optimization of structures with integral compliant mechanisms for mid-frequency response*, Int. J. Numer. Meth. Eng., **73**, pp. 470–492, 2008.
21. Salajegheh E., Gholizadeh S., Torkzadeh P., *Optimal design of structures with frequency constraints using wavelet back propagation neural,* Asian Journal of Civil Engineering (Building and Housing), **8**, pp. 97–111, 2007.
22. Leu L.J., Chang W.H., *Reanalysis-based optimal design of trusses*, Int. J. Numer Methods Eng., **49**, *8*, pp. 1007–1028, 2000.

Received May 17, 2013