

TỐI ƯU TRỌNG LƯỢNG KHUNG THÉP CỘT ĐẶC DÀN VÌ KÈO SỬ DỤNG THUẬT TOÁN TIẾN HÓA VI PHÂN

Nguyễn Trần Hiếu^{a,*}, Vũ Anh Tuấn^a, Nguyễn Quốc Cường^a

^a*Khoa Xây dựng dân dụng và công nghiệp, Trường Đại học Xây dựng,
số 55 đường Giải Phóng, quận Hai Bà Trưng, Hà Nội, Việt Nam*

Nhận ngày 29/08/2019, Sửa xong 17/09/2019, Chấp nhận đăng 18/09/2019

Tóm tắt

Kết cấu thép thường được sử dụng trong những công trình vượt khẩu độ lớn như nhà công nghiệp, nhà triển lãm, rạp hát, nhà thi đấu, sân vận động nhờ những ưu điểm như cường độ chịu lực cao, trọng lượng nhẹ. Kết cấu khung là dạng kết cấu phổ biến thường được áp dụng trong nhà công nghiệp có nhịp dưới 40 m. Bài báo giới thiệu một nghiên cứu về tối ưu trọng lượng cho khung thép bao gồm cột đặc liên kết với mái dàn vì kèo. Quá trình tối ưu dựa trên thuật toán tiến hóa vi phân được triển khai bằng ngôn ngữ lập trình VBA với các biến thiết kế là tiết diện của cột thép và các thanh dàn, hàm mục tiêu là trọng lượng của cả khung thép và ràng buộc thiết kế gồm thỏa mãn đồng thời cả trạng thái giới hạn về chịu lực và trạng thái giới hạn về điều kiện sử dụng. Việc kiểm tra các ràng buộc thiết kế này được thực hiện qua phân tích kết cấu bằng phần mềm CSI SAP2000. Một ví dụ bằng số được thực hiện để minh họa khả năng của thuật toán tối ưu. Ví dụ khảo sát một số yếu tố như hình dạng dàn, cấu tạo hệ thanh bụng của dàn với mục đích tìm ra sơ đồ hợp lý cho dạng kết cấu trên.

Từ khoá: kết cấu thép; nhà công nghiệp; khung thép dàn vì kèo; tối ưu kết cấu; thuật toán tiến hóa vi phân.

WEIGHT OPTIMIZATION OF STEEL TRUSS FRAMES USING DIFFERENTIAL EVOLUTION ALGORITHM

Abstract

Structural steel is most commonly used in large-span buildings such as industrial buildings, exhibitions, theaters, gymnasiums, and stadiums because of its advantages including high strength, lightweight. The portal frame is the most popular structure which is normally applied to industrial buildings up to 40 m span. This article presents a study on the weight optimization of steel truss frames. The optimization process based on differential evolution algorithm, is implemented by VBA language. The design variables are the cross-section dimensions of column and truss members, the objective function is the weight of the truss frame while the design constraint includes satisfying both of ultimate limit state and serviceability limit state. The design constraints are checked using the software CSI SAP2000. An example is then conducted to demonstrate the applicability of the developed program. Moreover, a survey is carried out by changing the shape of the truss and the configuration of the web members. The purpose of the survey is to find the most suitable shape for this kind of structure.

Keywords: steel structures; industrial buildings; truss frames; structural optimization; differential evolution algorithm.

[https://doi.org/10.31814/stce.nuce2019-13\(5V\)-07](https://doi.org/10.31814/stce.nuce2019-13(5V)-07) © 2019 Trường Đại học Xây dựng (NUCE)

1. Giới thiệu

Trong những năm gần đây, nhu cầu xây dựng nhà xưởng phục vụ sản xuất tăng cao dẫn đến sự bùng nổ về số lượng các công ty chế tạo kết cấu thép. Để tăng sức cạnh tranh dưới áp lực của thị

*Tác giả chính. Địa chỉ e-mail: hieunt2@nuce.edu.vn (Hiếu, N. T.)

trường, các đơn vị sản xuất luôn yêu cầu kỹ sư thiết kế đưa ra phương án kết cấu kinh tế nhất trong khi vẫn phải đảm bảo điều kiện chịu lực. Phương pháp thường được các kỹ sư sử dụng là phương pháp “thử - sai”. Tuy nhiên đối với bài toán có quá nhiều biến số, phương pháp này hiệu quả không cao khi phải thử đi thử lại nhiều trường hợp, kết quả thu được không đảm bảo là kết quả tối ưu nhất mà phụ thuộc nhiều vào kinh nghiệm của kỹ sư thiết kế. Điều này dẫn tới nhu cầu nghiên cứu những thuật toán tìm kiếm phương án tối ưu hiệu quả hơn nhằm giảm thời gian thiết kế cũng như đảm bảo luôn thu được kết quả tốt nhất.

Từ lâu, bài toán tối ưu đã được nhiều nhà khoa học quan tâm nghiên cứu. Một số lý thuyết tối ưu như quy hoạch tuyến tính, quy hoạch phi tuyến, quy hoạch động đã được đưa vào giảng dạy trong chương trình bậc đại học. Những lý thuyết này hứa hẹn có thể giải quyết được nhiều bài toán trong lĩnh vực giao thông, xuất nhập khẩu, đem lại hiệu quả kinh tế lớn. Trong ngành xây dựng, bài toán tối ưu kết cấu công trình cũng được nghiên cứu từ khá sớm [1]. Do bài toán tối ưu kết cấu thường là phi tuyến, rời rạc, nhiều biến số, những thuật toán dựa trên cơ chế tự nhiên như thuật toán tiến hóa (Evolutionary Algorithms – EAs) hoặc thuật toán trí tuệ bầy đàn (Swarm Intelligence – SI) tỏ ra có ưu thế hơn những loại thuật toán khác. Các thuật toán tiến hóa bao gồm: giải thuật di truyền (Genetic Algorithm – GA) [2], quy hoạch di truyền (Genetic Programming – GP), quy hoạch tiến hóa (Evolution Programming – EP), chiến lược tiến hóa (Evolution Strategy – ES) [3], tiến hóa vi phân (Differential Evolution – DE) [4]. Nhóm thuật toán trí tuệ bầy đàn có thể liệt kê một số thuật toán điển hình như thuật toán tối ưu bầy đàn (Particle Swarm Optimization – PSO) [5], thuật toán đàn kiến (Ant Colony Optimization – ACO) [6], thuật toán bầy ong (Artificial Bee Colony – ABC) [7], ...

Đối với kết cấu khung thép nói riêng, một số nghiên cứu về tối ưu dựa trên EA hoặc SI đã được triển khai [8–16]. Thuật toán thông dụng nhất là GA được đề xuất từ những năm 60 của thế kỷ XX [8, 10, 13–15]. Trong [16], các tác giả đã áp dụng thuật toán DE để tối ưu một số dạng kết cấu như khung cột vát - dầm vì kèo, khung cột và xà ngang vát,... Chương trình tối ưu phát triển trong [16] đọc kết quả phân tích kết cấu được xuất ra từ phần mềm SOFISTiK dưới dạng tệp văn bản do SOFISTiK chưa cho phép kết nối trực tiếp với các phần mềm khác. Vấn đề này được giải quyết trong [17] bằng cách sử dụng tính năng oAPI (open Application Programming Interface) của phần mềm CSI SAP2000. Hạn chế của chương trình phát triển trong [17] là yêu cầu người dùng phải cài đặt cả hai phần mềm thương mại là Matlab và CSI SAP2000.

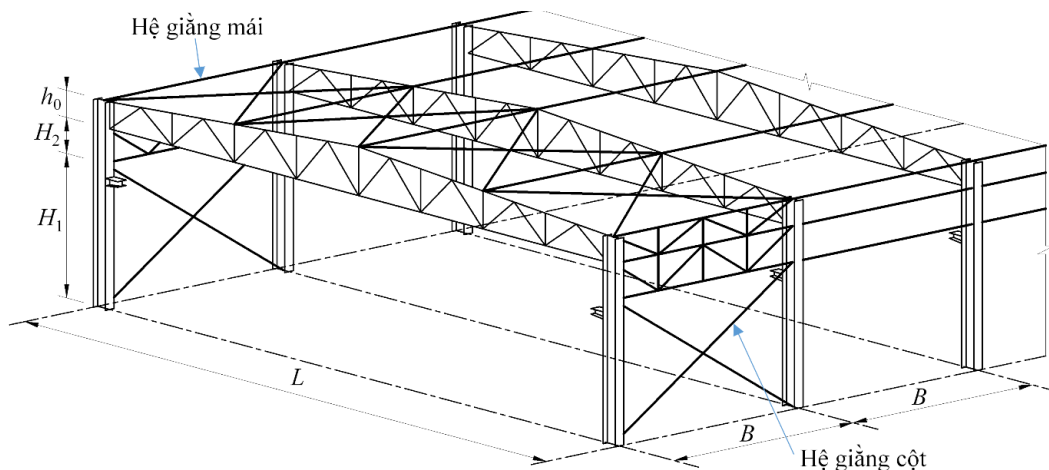
Bài báo trình bày một nghiên cứu về tối ưu trọng lượng cho dạng khung thép thường gặp trong nhà công nghiệp gồm cột đặc tiết diện chữ I cánh rộng liên kết với mái dầm vì kèo. Chương trình tối ưu FrameOpt dựa trên thuật toán DE được phát triển bằng ngôn ngữ lập trình Visual Basic for Application (VBA) trên nền Microsoft Excel và kết nối với phần mềm CSI SAP2000 thông qua oAPI. Đây là những phần mềm thông dụng hiện nay, giúp cho kỹ sư thiết kế dễ dàng tiếp cận kết quả của nghiên cứu. Bên cạnh đó, một khảo sát được thực hiện trong đó thay đổi hình dạng dầm và sơ đồ hệ thanh bụng nhằm xác định sơ đồ hợp lý của dầm vì kèo.

Bài báo được cấu trúc như sau: Mục 2 giới thiệu một số vấn đề cụ thể trong bài toán thiết kế khung thép cột đặc – dầm vì kèo. Thuật toán DE cũng như chương trình tối ưu được trình bày cụ thể trong Mục 3. Mục 4 thực hiện ví dụ bằng số nhằm thể hiện khả năng của thuật toán DE trong việc giải quyết bài toán tối ưu khung thép cũng như khảo sát ảnh hưởng của hình dáng cấu tạo dầm tới trọng lượng khung. Mục 5 rút ra một số kết luận và đề ra hướng phát triển tiếp theo của nghiên cứu.

2. Thiết kế khung thép cột đặc - dàn vì kèo

2.1. Các kích thước cơ bản

Một số kích thước cơ bản của khung thép: nhịp khung L , bước khung B , cao trình đỉnh ray H_1 , khoảng cách từ đỉnh ray đến đáy dàn vì kèo H_2 , chiều cao đầu dàn h_0 (Hình 1). Nhằm đảm bảo cầu trục hoạt động bình thường, chiều cao H_2 phải lớn hơn tổng chiều cao dầm cầu trục và xe con K_1 và khoảng cách an toàn theo phương đứng: $H_2 \geq K_1 + (*)$ trong đó kích thước K_1 và $(*)$ phụ thuộc nhà sản xuất cầu trục. Chiều cao giữa dàn (trong trường hợp dàn hình thang) được xác định từ chiều cao đầu dàn và độ dốc mái: $h_g = h_0 + (L/2)i$ với độ dốc mái thông thường $i = 10\%$ (Hình 3).



Hình 1. Cấu tạo khung thép

Hệ giằng dọc nhà bao gồm hệ giằng mái và giằng cột tạo thành miếng cứng bất biến hình. Hệ giằng được bố trí tại hai khoang đầu hồi và khoang giữa công trình. Để đảm bảo ổn định theo phương ngoài mặt phẳng, các khung trung gian được liên kết với các khoang giằng bằng thanh chống dọc như thể hiện trên Hình 1.

2.2. Tải trọng

Khung ngang được thiết kế với các loại tải trọng bao gồm: tải trọng thường xuyên (TT), hoạt tải sửa chữa mái (HT), áp lực đứng cầu trục (D_{max}), lực hãm cầu trục (T), hoạt tải gió (gió thổi ngang nhà - GN , gió thổi dọc nhà - GD). Hệ số khí động được lấy theo sơ đồ 2 - Bảng 6 tiêu chuẩn TCVN 2737:1995 [18]. Tải trọng động đất không được xét đến trong nghiên cứu. Nguyên tắc tổ hợp tải trọng tuân theo quy định nêu trong [18].

2.3. Thiết kế theo TCVN 5575:2012

a. Chiều dài tính toán cầu kiện

Chiều dài tính toán của cầu kiện bao gồm chiều dài tính toán trong mặt phẳng l_x và chiều dài tính toán ngoài mặt phẳng l_y . Đối với cột, chiều dài tính toán trong mặt phẳng $l_x = \mu l$ trong đó hệ số chiều dài tính toán μ được xác định theo Công thức (52) - Bảng 19 tiêu chuẩn TCVN 5575:2012 [19], l là chiều dài hình học của cột. Chiều dài tính toán ngoài mặt phẳng của cột l_y được lấy bằng khoảng cách giữa các thanh chống dọc. Đối với các thanh dàn vì kèo, chiều dài tính toán được lấy như trong Bảng 1 với l là khoảng cách giữa hai nút dàn.

Bảng 1. Chiều dài tính toán của thanh dàn vì kèo

Loại thanh	Chiều dài tính toán trong mặt phẳng l_x	Chiều dài tính toán ngoài mặt phẳng l_y
Thanh cánh	khoảng cách giữa 2 nút dàn liên tiếp	khoảng cách giữa các thanh chống dọc
Thanh xiên đầu dàn	l	l
Thanh xiên	$0,8l$	l
Thanh đứng	$0,8l$	l

b. Kiểm tra trạng thái giới hạn I

Trong sơ đồ tính, các thanh dàn được mô hình là cấu kiện thanh hai đầu khớp vì thế nội lực trong các thanh dàn chỉ có lực dọc trục. Các thanh dàn được kiểm tra như cấu kiện chịu kéo hoặc nén đúng tâm. Cột được kiểm tra như cấu kiện chịu nén uốn. Trong trường hợp độ lệch tâm tính đối m_e lớn hơn 20, cột được kiểm tra như cấu kiện chịu uốn.

c. Kiểm tra trạng thái giới hạn II

Khung ngang được kiểm tra chuyển vị đứng và chuyển vị ngang theo quy định trong Mục 5.3 - TCVN 5575:2012 [19]. Chuyển vị đứng được kiểm tra tại vị trí giữa dàn với độ võng cho phép lấy theo $[\Delta] = L/250$ trong đó L là nhịp dàn. Chuyển vị ngang được kiểm tra tại hai vị trí: cao trình đỉnh ray và cao trình diềm mái. Trong trường hợp sử dụng cầu trục chế độ làm việc nặng, giá trị giới hạn của chuyển vị ngang của cột lấy theo Bảng 2 - TCVN 5575:2012 [19]. Trường hợp cầu trục chế độ làm việc nhẹ và trung bình, chuyển vị ngang giới hạn không vượt quá $1/300$ chiều cao khung theo điều 5.3.4 - TCVN 5575:2012 [19].

3. Tối ưu bằng thuật toán DE

3.1. Bài toán tối thiểu hóa trọng lượng khung thép

Bài toán tối thiểu hóa trọng lượng khung thép có thể phát biểu tổng quát như sau:

$$\begin{aligned} \text{Tìm } \mathbf{x} \text{ sao cho: } f(\mathbf{x}) &= \sum_{k=1}^n \rho A_k L_k \text{ nhỏ nhất,} \\ \text{thỏa mãn điều kiện } &\begin{cases} \Delta(\mathbf{x}) \leq [\Delta]; \sigma(\mathbf{x}) \leq [\sigma]; \lambda(\mathbf{x}) \leq [\lambda] \\ \mathbf{x} = \{x_j^L \leq x_j \leq x_j^U\}, j = 1, 2, \dots, D \end{cases} \end{aligned} \tag{1}$$

trong đó \mathbf{x} là một véc-tơ D chiều chứa giá trị của D biến số x_j ; $f(\mathbf{x})$ là hàm mục tiêu của bài toán, trong trường hợp cụ thể của nghiên cứu là tổng trọng lượng của khung (đơn vị là kg); n là số lượng cấu kiện trong khung; ρ là trọng lượng riêng của thép (7850 kg/m^3); A_k là diện tích tiết diện các cấu kiện thứ k ; L_k là chiều dài tương ứng của cấu kiện thứ k ; $\Delta(\mathbf{x})$ là chuyển vị của khung tương ứng với véc-tơ \mathbf{x} ; $[\Delta]$ là chuyển vị cho phép, $\sigma(\mathbf{x})$ là ứng suất (hoặc nội lực) trong cấu kiện khung, $[\sigma]$ là ứng suất cho phép (hoặc sức kháng) của cấu kiện tương ứng; $\lambda(\mathbf{x})$ là độ mảnh của cấu kiện; $[\lambda]$ là độ mảnh giới hạn của cấu kiện; x_j là biến thứ j , x_j^L và x_j^U là cận trên và cận dưới của biến x_j ; j là chỉ số của biến có giá trị $1, 2, \dots, D$.

Biến số của bài toán tối ưu trong nghiên cứu là tiết diện của cột và các thanh dàn. Các yếu tố khác như tải trọng (trừ trọng lượng bản thân của cấu kiện), kích thước hình học của khung được coi như

hàng số không thay đổi trong quá trình tối ưu. Các điều kiện ràng buộc được trình bày cụ thể trong Mục 2.4, bao gồm:

- Điều kiện ràng buộc về chịu lực: $\sigma(\mathbf{x}) \leq [\sigma]$.
- Điều kiện ràng buộc về chuyển vị: $\Delta(\mathbf{x}) \leq [\Delta]$.
- Điều kiện ràng buộc về độ mảnh: $\lambda(\mathbf{x}) \leq [\lambda]$.

3.2. Thuật toán DE

Thuật toán DE được lựa chọn để giải quyết bài toán tối ưu trọng lượng khung thép. Trong suốt quá trình phát triển, thuật toán DE đã có rất nhiều biến thể. Nghiên cứu trong bài báo này sử dụng thuật toán DE cổ điển được đề xuất bởi K. Price và R. Storn vào năm 1997, bao gồm bốn bước cơ bản:

- Khởi tạo: một quần thể ban đầu bao gồm N_p cá thể được khởi tạo, mỗi cá thể là một véc-tơ D chiều \mathbf{x}_i đặc trưng cho D biến của bài toán tối ưu.

$$x_{ij} = x_j^L + \text{rnd}(0, 1) \times (x_j^U - x_j^L) \quad (2)$$

trong đó i có giá trị $1, 2, \dots, N_p$; j có giá trị $1, 2, \dots, D$; N_p là số lượng cá thể trong một quần thể; D là số lượng biến trong bài toán tối ưu; x_j^L và x_j^U là cận trên và cận dưới của biến x_j , $\text{rnd}(0, 1)$ là một số ngẫu nhiên trong khoảng từ 0 đến 1.

- Đột biến: cá thể đột biến \mathbf{v}_i được tạo ra từ mỗi cá thể ban đầu \mathbf{x}_i :

$$\mathbf{v}_i = \mathbf{x}_{r1} + F \times (\mathbf{x}_{r2} - \mathbf{x}_{r3}) \quad (3)$$

trong đó $r_1 \neq r_2 \neq r_3$ được lựa chọn ngẫu nhiên trong $1, 2, \dots, N_p$, F là hệ số khuếch đại thường có giá trị trong khoảng từ 0 đến 1.

- Lai ghép: cá thể thử nghiệm \mathbf{u}_i được tạo ra bằng cách thay một số thành phần của cá thể đột biến \mathbf{v}_i bằng các thành phần tương ứng của cá thể ban đầu \mathbf{x}_i :

$$u_{ij} = \begin{cases} v_{ij} & \text{nếu } \text{rnd}(0, 1) \leq Cr \\ x_{ij} & \text{ngược lại} \end{cases} \quad (4)$$

trong đó i có giá trị $1, 2, \dots, N_p$; j có giá trị $1, 2, \dots, D$; v_{ij} là thành phần thứ j của cá thể đột biến \mathbf{v}_i ; x_{ij} là thành phần thứ j của cá thể ban đầu \mathbf{x}_i ; Cr là hệ số lai ghép.

- Chọn lọc: so sánh cá thể thử nghiệm \mathbf{u}_i với cá thể ban đầu \mathbf{x}_i , cá thể nào tốt hơn (có hàm mục tiêu nhỏ) sẽ được giữ lại cho thế hệ tiếp theo:

$$\mathbf{x}_i = \begin{cases} \mathbf{u}_i & \text{nếu } f(\mathbf{u}_i) \leq f(\mathbf{x}_i) \\ \mathbf{x}_i & \text{ngược lại} \end{cases} \quad (5)$$

trong đó $f(\mathbf{u}_i)$ và $f(\mathbf{x}_i)$ là giá trị hàm mục tiêu của cá thể tương ứng.

Quá trình tối ưu lặp đi lặp lại cho tới thế hệ n_{gen}^{\max} được thiết lập từ ban đầu.

3.3. Xử lý điều kiện ràng buộc

Nghiên cứu sử dụng phương pháp được đề xuất bởi Lampien [20] để xử lý điều kiện ràng buộc. Đầu tiên, do bài toán tối ưu kết cấu yêu cầu phải thỏa mãn đồng thời nhiều loại điều kiện ràng buộc, để thuận tiện khi triển khai thuật toán, các điều kiện ràng buộc được chuẩn hóa thành dạng:

$$g_{\Delta}(\mathbf{x}) = \frac{\Delta(\mathbf{x})}{[\Delta]} \leq 1; \quad g_{\sigma}(\mathbf{x}) = \frac{\sigma(\mathbf{x})}{[\sigma]} \leq 1; \quad g_{\lambda}(\mathbf{x}) = \frac{\lambda(\mathbf{x})}{[\lambda]} \leq 1 \quad (6)$$

Một cách tổng quát, giả sử bài toán tối ưu có m điều kiện ràng buộc:

$$g_k(\mathbf{x}) \leq 1 \text{ với } k = 1, \dots, m \quad (7)$$

Tại bước chọn lọc, phương trình (5) được thay thế bằng phương trình:

$$\mathbf{x}_i = \begin{cases} \mathbf{u}_i & \text{nếu} \\ \mathbf{x}_i & \text{các trường hợp còn lại} \end{cases} \left\{ \begin{array}{l} \left\{ \begin{array}{l} \forall k \in \{1, \dots, m\} : g_k(\mathbf{u}_i) \leq 1 \wedge g_k(\mathbf{x}_i) \leq 1 \\ \wedge \\ f(\mathbf{u}_i) \leq f(\mathbf{x}_i) \\ \vee \\ \left\{ \begin{array}{l} \forall k \in \{1, \dots, m\} : g_k(\mathbf{u}_i) \leq 1 \\ \wedge \\ \forall k \in \{1, \dots, m\} : g_k(\mathbf{x}_i) > 1 \end{array} \right. \\ \vee \\ \left\{ \begin{array}{l} \forall k \in \{1, \dots, m\} : g_k(\mathbf{u}_i) > 1 \\ \wedge \\ \forall k \in \{1, \dots, m\} : g'_k(\mathbf{u}_i) \leq g'_k(\mathbf{x}_i) \end{array} \right. \end{array} \right. \end{array} \right. \quad (8)$$

trong đó $g'_k(\mathbf{x}_i) = \max(g_k(\mathbf{x}_i), 1)$; $g'_k(\mathbf{u}_i) = \max(g_k(\mathbf{u}_i), 1)$.

3.4. Xử lý bài toán tối ưu rời rạc

Thuật toán DE ban đầu được đề xuất cho bài toán biến liên tục, không thích hợp để giải quyết dạng bài toán đang nghiên cứu khi tiết diện phải được lựa chọn trong một danh mục thép hình có sẵn. Để xử lý vấn đề này, biến tiết diện thép hình được thay thế bằng biến số nguyên tương ứng với số thứ tự của tiết diện lựa chọn trong danh mục thép hình sử dụng. Kích thước cũng như các đặc trưng hình học của tiết diện được tra bảng theo số thứ tự tương ứng.

Khi đó, quần thể ban đầu được khởi tạo theo công thức sau:

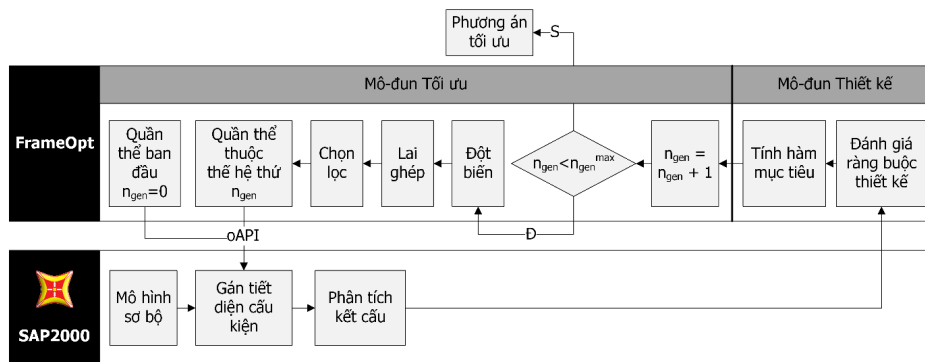
$$x_{ij} = x_j^L + \text{round} \left[\text{rnd}(0, 1) \times (x_j^U - x_j^L) \right] \quad (9)$$

trong đó hàm $\text{round}[]$ trả về số nguyên gần nhất của với giá trị trong ngoặc. Và công thức (3) được viết lại như sau:

$$\mathbf{v}_i = \mathbf{x}_{r1} + \text{round} [F \times (\mathbf{x}_{r2} - \mathbf{x}_{r3})] \quad (10)$$

3.5. Chương trình tối ưu FrameOpt

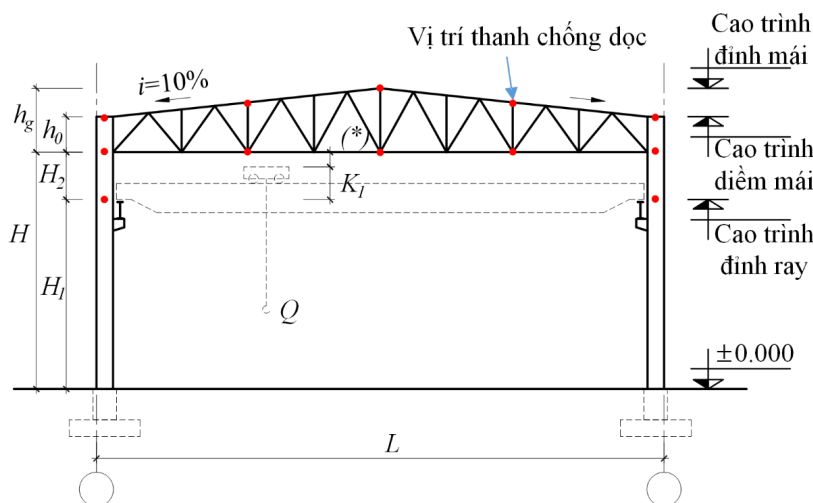
Nhằm triển khai thuật toán trên, chương trình FrameOpt được phát triển bằng ngôn ngữ lập trình VBA trên nền phần mềm Microsoft Excel, bao gồm hai mô-đun chính: mô-đun tối ưu và mô-đun thiết kế. Mô-đun tối ưu thực hiện nhiệm vụ khởi tạo các cá thể của quần thể ban đầu, đột biến, lai ghép và chọn lọc; trong khi đó mô-đun thiết kế thực hiện nhiệm vụ đánh giá các điều kiện ràng buộc và tính toán hàm mục tiêu. Để đánh giá điều kiện ràng buộc, mỗi cá thể tương ứng với một phương án khung thép được phân tích bằng phần mềm phân tích kết cấu CSI SAP2000. Phần mềm sử dụng phương pháp phần tử hữu hạn để xác định nội lực cũng như chuyển vị của kết cấu. Chương trình FrameOpt kết nối với phần mềm CSI SAP2000 thông qua các hàm oAPI. Đây là một tính năng mạnh mẽ, cho phép người dùng có thể tự động hóa nhiều thao tác như xây dựng mô hình, gán tải trọng, phân tích kết cấu, thiết kế cấu kiện, xuất kết quả. SAP2000 oAPI hỗ trợ các ngôn ngữ lập trình thông dụng như VBA, VB.NET, C#, Matlab, Python. Sơ đồ khối chương trình FrameOpt được thể hiện trong Hình 2.



Hình 2. Sơ đồ khối chương trình FrameOpt

4. Ví dụ bằng số

Áp dụng thuật toán DE để tối ưu trọng lượng một khung thép có các thông số cụ thể như sau: khung ngang có nhịp $L = 24$ m, bước $B = 7$ m, cao trình đỉnh ray $H_1 = 8$ m, khoảng cách từ đỉnh ray đến đáy dàn vì kèo $H_2 = 2$ m, chiều cao đầu dàn $h_0 = 1,5$ m (Hình 3). Công trình đặt tại vùng gió III.B, dạng địa hình B. Cầu trục sức trục $Q = 25$ T, chế độ làm việc trung bình. Vật liệu thép mác CCT34 có cường độ tiêu chuẩn (giới hạn chảy) $f_y = 220$ N/mm² và cường độ kéo đứt tiêu chuẩn (giới hạn bền) $f_u = 340$ N/mm². Thanh chống dọc được bố trí tại những vị trí như thể hiện trong Hình 3. Khoảng cách theo phương ngang giữa các thanh chống dọc trên mặt phẳng mái là 6 m.



Hình 3. Kích thước hình học của khung ngang

Do sự hạn chế chủng loại của tiết diện thép cán nóng theo tiêu chuẩn Việt Nam, nghiên cứu sử dụng danh mục thép hình cán nóng theo tiêu chuẩn Châu Âu. Cụ thể đối với cột sử dụng thép hình HE là loại tiết diện chữ I cán nóng cánh rộng, phù hợp để làm cấu kiện chịu nén uốn. Thanh dàn sử dụng tiết diện hộp vuông. Để hạn chế số loại tiết diện trong dàn vì kèo, các thanh dàn được nhóm thành 5 nhóm bao gồm: thanh cánh trên, thanh cánh dưới, thanh xiên đầu dàn, các thanh xiên còn lại và thanh đứng. Các thanh trong cùng một nhóm có tiết diện giống nhau. Tổng số biến thiết kế trong bài toán là

$D = 6$ biến với giới hạn biên được trình bày cụ thể trong Bảng 2. Các thông số khác như kích thước của khung, tải trọng tác dụng không thay đổi trong suốt quá trình tối ưu được coi là hằng số thiết kế.

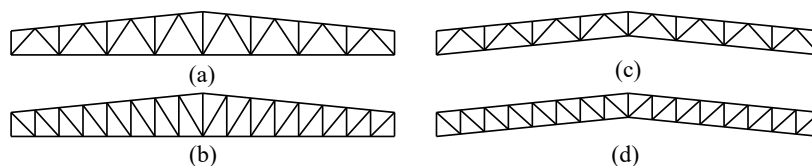
Bảng 2. Giới hạn của biến số thiết kế

STT	Biến số	Giới hạn biên
1	Cột	HE100A ÷ HE1000M (số lượng tiết diện $n_{td} = 70$)
2	Thanh cánh trên	
3	Thanh cánh dưới	
4	Thanh xiên đầu dàn	$\square 50 \times 3 \div \square 300 \times 12,5$ (số lượng tiết diện $n_{td} = 56$)
5	Thanh xiên khác	
6	Thanh đứng	

Kết cấu được coi làm việc hoàn toàn trong giai đoạn đàn hồi tuyến tính. Phân tích kết cấu không kể đến ảnh hưởng của hiệu ứng bậc hai. Kết quả phân tích được sử dụng để kiểm tra khả năng chịu lực của cấu kiện và điều kiện chuyển vị theo tiêu chuẩn Việt Nam.

Sử dụng chương trình FrameOpt đã phát triển ở Mục 3.5 để tối ưu trọng lượng cho 04 trường hợp khung thép có cùng số liệu thiết kế như trên nhưng hình dạng và cấu tạo dàn khác nhau, cụ thể:

- Khung A: dàn hình thang, hệ thanh bụng tam giác (Hình 4(a)).
- Khung B: dàn hình thang, hệ thanh bụng xiên (Hình 4(b)).
- Khung C: dàn hai cánh song song, hệ thanh bụng tam giác (Hình 4(c)).
- Khung D: dàn hai cánh song song, hệ thanh bụng xiên (Hình 4(d)).



Hình 4. Các trường hợp dàn vì kèo trong nghiên cứu

Một số nghiên cứu khuyến cáo tiến hành chạy tối thiểu 20 lần độc lập cho mỗi bài toán và sử dụng thống kê để đánh giá độ tin cậy của thuật toán [21, 22]. Tuy nhiên, do hạn chế về tài nguyên tính toán, nghiên cứu trong bài báo sử dụng kết quả với thông số được thiết lập như sau: số lượng thế hệ $n_{gen}^{max} = 100$; số lượng cá thể trong một quần thể $N_p = 10D = 10 \times 6 = 60$; hệ số khuếch đại $F = 0,7$ và hệ số lai ghép $CR = 0,8$. Chương trình chạy trên máy tính có cấu hình: bộ vi xử lý Intel Core i5-5257 2,7 Ghz, bộ nhớ trong (RAM) 8 Gb. Quá trình tối ưu một trường hợp khung yêu cầu thực hiện 6060 lần phân tích kết cấu với tổng thời gian tính toán là 6 giờ 44 phút.

Triển khai tối ưu cho 04 trường hợp khung. Kết quả tối ưu được trình bày trong Bảng 3. Giá trị hàm mục tiêu (tổng trọng lượng khung thép) nhỏ nhất ứng với mỗi thế hệ trong suốt quá trình tối ưu được thể hiện ở Hình 5.

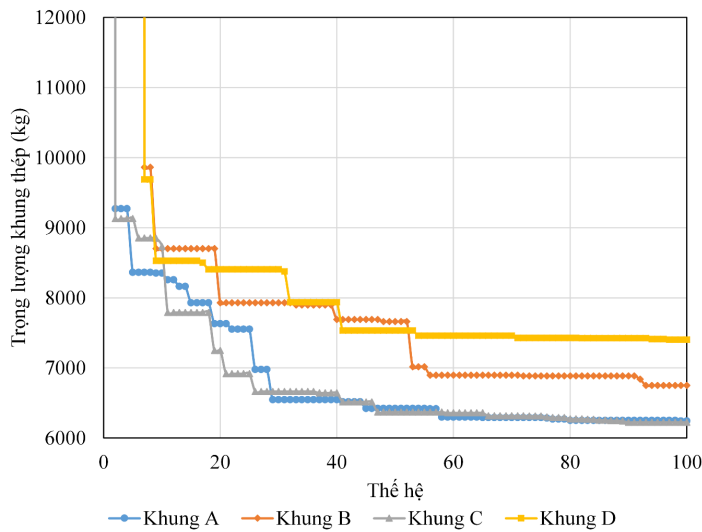
Căn cứ trên kết quả tối ưu, một số nhận xét có thể rút ra như sau:

- Khung D có trọng lượng lớn nhất, tiếp theo lần lượt là khung B, khung A và khung C. Hai khung A và C có trọng lượng chênh lệch nhau tương đối nhỏ.

- Đối với cả dạng dàn hình thang và dàn hai cánh song song, cấu tạo hệ thanh bụng tam giác cho trọng lượng nhỏ hơn hệ thanh bụng xiên.

Bảng 3. Kết quả tối ưu

	Khung A	Khung B	Khung C	Khung D
Cột	HE500A	HE500A	HE500A	HE500A
Thanh cánh trên	□250 × 8	□250 × 6,3	□250 × 8	□300 × 8
Thanh cánh dưới	□180 × 5	□250 × 8	□180 × 5	□250 × 8
Thanh xiên đầu dàn	□70 × 3,6	□60 × 4	□60 × 3	□60 × 3
Thanh xiên	□60 × 3	□50 × 3	□60 × 3	□50 × 3
Thanh đứng	□50 × 3	□50 × 4	□50 × 3	□50 × 3
Điều kiện ràng buộc chuyển vị $g_{\Delta}(\mathbf{x})$	0,85	0,87	0,86	0,83
Điều kiện ràng buộc chịu lực $g_{\sigma}(\mathbf{x})$	0,87	0,99	0,93	0,95
Tổng trọng lượng khung (kg)	6242	6748	6224	7402



Hình 5. Trọng lượng nhỏ nhất của khung thép ứng với mỗi thể hệ

5. Kết luận và kiến nghị

Bài báo trình bày một nghiên cứu trong đó thuật toán tiến hóa vi phân được sử dụng để tối ưu trọng lượng khung thép. Bài toán tối ưu có sáu biến số là tiết diện của các cấu kiện: cột, thanh dàn cánh trên, thanh dàn cánh dưới, thanh xiên đầu dàn, các thanh xiên còn lại và thanh đứng. Chương trình máy tính FrameOpt được phát triển nhằm triển khai thuật toán vào bài toán tối ưu kết cấu. Một ví dụ bằng số cho thấy tính khả thi của chương trình FrameOpt trong công tác thiết kế thực tế. Bên cạnh đó, kết quả khảo sát trên bốn trường hợp dàn có hình dáng và cấu tạo dàn khác nhau giúp đưa ra kết luận sơ bộ về sơ đồ hợp lý cho kết cấu dàn mái. Nghiên cứu có thể tiếp tục phát triển để áp dụng cho các loại kết cấu khác như kết cấu khung thép tiền chế, kết cấu dàn không gian vượt nhịp lớn, ... Bên cạnh đó, việc rút ngắn thời gian tối ưu cũng là một vấn đề cần quan tâm nghiên cứu trong tương lai.

Lời cảm ơn

Nghiên cứu sinh được hỗ trợ bởi chương trình học bổng đào tạo thạc sĩ, tiến sĩ trong nước của Quỹ Đổi mới sáng tạo Vingroup.

Tài liệu tham khảo

- [1] Spunt, L. (1971). *Optimum structural design*. Prentice-Hall. Englewood Cliffs, New Jersey, USA.
- [2] Goldberg, D. E. (1989). *Genetic algorithms in search, optimization and machine learning*. Addison-Wesley Longman Publishing Co., Inc. Boston, MA, USA.
- [3] Beyer, H. G. (2001). *The theory of evolution strategies*. Springer, Germany.
- [4] Price, K. V., Storn, R. M., Lampien, J. A. (2005). *Differential evolution: A practical approach to global optimization*. Springer, Germany.
- [5] Eberhart, R., Kennedy, J. (1995). **Particle swarm optimization**. *Proceedings of IEEE International Conference on Neural Networks IV, 1942–1948*.
- [6] Dorigo, M., Stützle, T. (2004). *Ant colony optimization*. MIT Press, USA.
- [7] Karaboga, D. (2005). *An idea based on honey bee swarm for numerical optimization*. Technical Report - TR06, Vol 200.
- [8] Kaveh, A., Kalatjari, V. (2002). **Genetic algorithm for discrete-sizing optimal design of trusses using the force method**. *International Journal for Numerical Methods in Engineering*, 55(1):55–72.
- [9] Kaveh, A., Talatahari, S. (2008). **A discrete particle swarm ant colony optimization for design of steel frames**. *Asian Journal of Civil Engineering*, 9(6):563–575.
- [10] Saka, M. P. (2003). **Optimum design of pitched roof steel frames with haunched rafters by genetic algorithm**. *Computers & Structures*, 81(18-19):1967–1978.
- [11] Saka, M. P. (2009). **Optimum design of steel sway frames to BS5950 using harmony search algorithm**. *Journal of Constructional Steel Research*, 65(1):36–43.
- [12] Hasançebi, O., Bahçecioglu, T., Kurç, O., Saka, M. P. (2011). **Optimum design of high-rise steel buildings using an evolution strategy integrated parallel algorithm**. *Computers & Structures*, 89(21-22):2037–2051.
- [13] Phan, D. T., Lim, J. B. P., Ming, C. S. Y., Tanyimboh, T., Issa, H., Sha, W. (2011). **Optimization of cold-formed steel portal frame topography using real-coded genetic algorithm**. *Procedia Engineering*, 14:724–733.
- [14] Phan, D. T., Lim, J. B. P., Sha, W., Siew, C. Y. M., Tanyimboh, T. T., Issa, H. K., Mohammad, F. A. (2013). **Design optimization of cold-formed steel portal frames taking into account the effect of building topology**. *Engineering Optimization*, 45(4):415–433.
- [15] Phan, D. T., Lim, J. B. P., Tanyimboh, T. T., Sha, W. (2017). **Optimum design of cold-formed steel portal frame buildings including joint effects and secondary members**. *International Journal of Steel Structures*, 17(2):427–442.
- [16] Vu, A. T., Werner, F. (2009). **Optimization of steel frame structures based on differential evolution algorithm**. *Proceeding of 18th International conference on the applications of computer science and mathematics in Architecture and Civil engineering (IKM)*. Weimar, Germany.
- [17] Pham, H. A., Dang, V. H. (2016). Automated optimal design of truss structures using modified DE and SAP2000 open application programming interface (OAPI). *The 4th International Conference on Engineering Mechanics And Automation (ICEMA 4)*. Hà Nội, Việt Nam.
- [18] TCVN 2737:1995. *Tải trọng và tác động*. Bộ Khoa học và Công nghệ, Việt Nam.
- [19] TCVN 5575:2012. *Kết cấu thép Tiêu chuẩn thiết kế*. Bộ Khoa học và Công nghệ, Việt Nam.
- [20] Lampien, J. (2002). A constraint handling approach for the differential evolution algorithm. *Proceedings of the 2002 Congress on Evolutionary Computation. CEC'02 (Cat. No.02TH8600)*, Honolulu, USA.
- [21] Pham, H. A. (2016). **Truss optimization with frequency constraints using enhanced differential evolution based on adaptive directional mutation and nearest neighbor comparison**. *Advances in Engineering Software*, 102:142–154.
- [22] Anh, P. H., Duong, T. T. (2019). **Weight optimisation of functionally graded beams using modified differential evolution**. *Journal of Science and Technology in Civil Engineering (STCE)-NUCE*, 13(2): 48–63.