

# Tối ưu khối lượng khung thép sử dụng phân tích trực tiếp và thuật toán tiến hóa vi phân tự thích ứng

Optimization of steel frame mass using direct analysis and self-adaptive pbest differential evolution algorithm

> NCS NGUYỄN THỊ THANH THÚY, NCS CAO TRƯỜNG SƠN, PGS.TS TRƯƠNG VIỆT HÙNG\*

Khoa Công trình, Trường Đại học Thủy lợi,

\* Corresponding author

## TÓM TẮT

Tối ưu trong thiết kế công trình nói chung và hệ khung thép nói riêng thu hút sự quan tâm ngày càng lớn nhờ sự phát triển mạnh mẽ của khoa học máy tính cũng như sự tăng cao của chi phí vật liệu xây dựng. Đối với kết cấu khung thép, tổng giá thành/khối lượng của công trình thường được tối thiểu hóa dựa trên biến thiết kế là các tiết diện điển hình của các cấu kiện. Cùng với đó, ứng xử của khung thép là phi tuyến tính và phi đàn hồi nên bài toán tối ưu khung thép có tính phi tuyến rất cao. Trên cơ sở đó, trong bài báo này, một thuật toán tối ưu meta-heuristic trên cơ sở thuật toán tiến hóa vi phân DE nhằm tối ưu hóa khối lượng khung thép sử dụng phân tích trực tiếp. Hai thuật toán tối ưu Jaya và tối ưu bầy đàn (PSO) được sử dụng để đánh giá hiệu quả của thuật toán được xây dựng thông qua ví dụ khung thép 3 nhịp x 10 tầng. Kết quả tính toán cho thấy phương pháp được đề xuất tìm kiếm được nghiệm tối ưu tốt hơn và sử dụng số lần phân tích kết cấu ít hơn khá nhiều so với Jaya và PSO.

**Từ khóa:** Khung thép; phân tích trực tiếp; Meta-heuristic; tiến hóa vi phân

## ABSTRACT

Optimum design of construction works in general and steel frame structures in particular has attracted increasing attention, thanks to the strong development of computer science as well as the increasing cost of construction materials. For steel frame structures, the total cost/volume of the building is usually minimized based on the design variables that are the typical cross-sections of the components. Besides, the behavior of the steel frame is non-linear and inelastic, so the optimization problem of the steel frame is highly non-linear. On that basis, in this paper, a meta-heuristic optimization algorithm based on DE differential evolution algorithm aims to optimize the steel frame mass using direct analysis technique. Two algorithms Jaya and swarm optimization (PSO) are used to evaluate the efficiency of the algorithm built through an example of a 3-span x 10-storey steel frame. The calculation results show that the proposed method has found the optimal solution better and uses rather much less number of structural analysis times than Jaya and PSO algorithms.

**Key words:** Steel frame; direct analysis; Meta-heuristic; differential evolution.

## 1. ĐẶT VẤN ĐỀ

Nhờ ưu điểm nổi trội của vật liệu thép về khả năng chịu lực (kéo và nén), khả năng biến dạng, cùng độ bền cao, các kết cấu thép nói chung và kết cấu khung nói riêng ngày càng được sử dụng phổ biến cho các công trình. Tiết diện cấu kiện dầm và cột của kết cấu khung thép thường thanh mảnh, khả năng chịu biến dạng cao nên trong quá trình tính toán thiết kế các đặc tính phi tuyến hình học và phi tuyến vật liệu cần được kể đến. Các phương pháp phân tích kết cấu thép truyền thống hai bước gồm: (1) xác định nội lực của hệ kết cấu sử dụng phân tích đàn hồi tuyến tính và

sau đó (2) thiết kế, kiểm tra khả năng chịu lực cho từng cấu kiện riêng lẻ có xét đến tính chất phi tuyến hình học được kể đến bằng hệ số uốn dọc xác định một cách gần đúng. Các công thức thiết kế và kiểm tra được cho sẵn trong các tiêu chuẩn thiết kế hiện hành như AISC LRFD [1], Eurocode [2]... Phương pháp tiếp cận này không mô tả được các ứng xử phi tuyến của công trình trong quá trình chịu tải trọng. Ứng xử và khả năng chịu lực cực hạn của hệ kết cấu cũng chưa được cung cấp rõ ràng. Đồng thời, việc thiết kế riêng lẻ cho từng cấu kiện vừa không đem lại hiệu quả kinh tế cao nhất vừa không đảm bảo sự tương tác trong quá trình làm việc chung. Để

khắc phục nhược điểm này, các phương pháp phân tích trực tiếp với đặc điểm là cho phép xét đến cả phi tuyến hình học và phi tuyến vật liệu của công trình đang được ứng dụng ngày càng nhiều [3-6]. Đặc điểm chính của phân tích trực tiếp là ứng xử của kết cấu được tính toán theo từng bước tải trọng nhỏ tăng dần một cách liên tục và qua đó ứng xử phi tuyến tính phi đàn hồi của toàn bộ kết cấu sẽ được ghi nhận theo từng cấp tải trọng. Từ đó, khả năng chịu tải của toàn bộ công trình được xác định. Sự an toàn của công trình sẽ được đánh giá thông qua so sánh giữa khả năng chịu tải của cả công trình và hiệu ứng do tải trọng gây ra. Ở phương pháp này không còn phải thực hiện bước phải kiểm tra từng cấu kiện riêng lẻ như trong các phương pháp thiết kế truyền thống. Như vậy, phương pháp thiết kế trực tiếp có các ưu điểm rõ ràng như: không cần dùng hệ số chiều dài tính toán, có kể đến sự tương tác do chầy dềo và mất ổn định khi tăng tải trọng, cung cấp tất cả các kết quả nội lực của toàn hệ kết cấu và có kể đến sự phân phối lại nội lực, dự đoán độ cứng hệ chính xác hơn, mô tả được ứng xử phi tuyến, trình tự và dạng phá hoại của các cấu kiện và hệ kết cấu, xác định được khả năng chịu lực cục bộ của hệ kết cấu. Một số nghiên cứu về ứng dụng các phương pháp phân tích trực tiếp trong thiết kế công trình thép có thể tham khảo các tài liệu [7] – [11].

Trong những năm gần đây, nhu cầu xây dựng kết cấu thép cũng như áp lực về giá thành vật liệu tăng cao đặt ra yêu cầu kỹ sư thiết kế phải tìm được phương án kết cấu kinh tế nhất trong khi vẫn phải đảm bảo các điều kiện về chịu lực, biến dạng và sử dụng. Từ lâu, bài toán tối ưu đã được nhiều nhà khoa học quan tâm nghiên cứu. Phương pháp truyền thống trong bài toán tối ưu thường được sử dụng là phương pháp "thử - sai". Tuy nhiên khi bài toán có nhiều biến số, phương pháp này hiệu quả không cao do phải thử đi thử lại nhiều trường hợp, kết quả thu được cũng chưa thể khẳng định là kết quả tối ưu nhất và còn bị 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. Trong ngành Xây dựng, bài toán tối ưu kết cấu công trình đã được nghiên cứu đưa vào thiết kế từ những năm 1971 [12]. Do bài toán tối ưu kết cấu công trình có những đặc điểm rất phức tạp như ứng xử phi tuyến có độ phức tạp cao, nhiều biến và các biến rời rạc, trong khi đó lại đa mục tiêu nên cho đến hiện nay sau quá trình phát triển thì các thuật toán meta-heuristic được sử dụng và thể hiện rõ hiệu quả với bài toán này. Các thuật toán meta-heuristic có thể kể đến các nhóm bao gồm: giải thuật di truyền (Genetic Algorithm - GA) - thông dụng nhất và được Rechenberg giới thiệu từ những năm 1960, quy hoạch di truyền (Genetic Programming - GP), quy hoạch tiến hóa (Evolution Programming - EP), chiến lược tiến hóa (Evolution Strategy - ES), tiến hóa vi phân (Differential Evolution - DE), Các hệ thống phân loại (Classifier Systems - CS), thuật toán tối ưu bầy đàn (Particle Swarm Optimization - PSO), thuật toán tối ưu thuộc địa đàn kiến (Ant Colony Optimization - ACO), đàn ong Artificial Bee Colony - ABC, ... Đối với kết cấu khung thép nói riêng, một số nghiên cứu về tối ưu dựa trên các thuật toán kể trên đã được triển khai [13–20].

Tiến hóa vi phân (Differential Evolution) (DE) là một thuật toán Meta-heuristic cho biến thiết kế liên tục khá đơn giản, dễ hiểu và dễ sử dụng nhưng lại là thuật toán mạnh và hiệu quả được Storn và Price đề xuất năm 1997 [21]. Đây là một loại thuật toán tiến hóa (Evolutionary Algorithms) (EA) với nền tảng cơ bản là kỹ thuật tìm kiếm ngẫu nhiên dựa trên một tập hợp các ứng viên tiềm năng (quần thể) và sử dụng các toán tử đột biến, lai tạo và lựa chọn ở mỗi thế hệ để tìm kiếm kết quả tối ưu [22]. Bằng cách sử dụng sơ

đồ đột biến đơn giản và hiệu quả này, DE đã cho thấy hiệu suất khá tốt trong các bài toán tối ưu hóa. Hiện nay ngày càng nhiều nghiên cứu nhằm mở rộng DE sang bài toán có biến thiết kế rời rạc. Các xử lý có thể là: (1) các cá thể mới đầu tiên vẫn được xác định dựa theo các kỹ thuật trong DE thông thường như là một biến liên tục nhưng sẽ được chuyển đổi sang miền rời rạc trước khi hàm mục tiêu được tính toán. Ở đây thực hiện chuyển đổi không gian dựa trên cách tiếp cận ngắt quãng miền liên tục; (2) Cách tiếp cận thứ hai là định nghĩa mỗi cá thể như là một hoán vị của các số. Các cá thể mới được tạo ra trên cơ sở tạo ra một véc-tơ hoán vị dựa trên các toán tử hoán đổi từ các cá thể hiện tại trong quần thể; (3) Cách tiếp cận thứ ba là kết hợp thêm một số thuật toán tìm kiếm địa phương vào thuật toán DE truyền thống. Nhìn chung, các cách tiếp cận trên đều dựa trên nguyên tắc cơ bản của DE về đột biến, lai tạo và lựa chọn.

Trong bài báo này, tác giả áp dụng phương pháp phân tích nâng cao để phân tích ứng xử kết cấu khung thép là kỹ thuật phân tích trực tiếp để kể đến ứng xử phi tuyến tính phi đàn hồi của kết cấu khung thép, khắc phục các nhược điểm của phương pháp thiết kế dựa trên phân tích đàn hồi. Một thuật toán nâng cấp từ DE bằng cách kết hợp hài hòa ưu điểm của hai kỹ thuật đột biến cơ bản của DE là 'DE/rand/1' và 'DE/best/1' bằng cách sử dụng phương pháp 'DE/pbest/1'. Thuật toán đề xuất được đặt tên là AEpDE (self-Adaptive pbest DE algorithm). Trong thuật toán này số lượng cá thể tốt nhất được xem xét theo phương pháp pbest dựa trên sự đa dạng của dân số hiện tại. Biến thiết kế là tiết diện của các thanh cột và dầm thép. Điều kiện ràng buộc bao gồm các điều kiện theo các tổ hợp tải trọng cường độ và sử dụng trong tiêu chuẩn AISC-LRFD[23]. Hiệu quả của phương pháp được so sánh với các kỹ thuật thông dụng như thuật toán Jaya, PSO và thuật toán đề xuất AEpDE.

## 2. THIẾT LẬP BÀI TOÁN TỐI ƯU KHỐI LƯỢNG KHUNG THÉP SỬ DỤNG KỸ THUẬT PHÂN TÍCH TRỰC TIẾP

Trong tối ưu hóa kích thước của khung thép, các mặt cắt ngang phù hợp cho dầm và cột được tìm kiếm từ các danh mục nhất định (ví dụ: danh sách tiết diện chữ W của AISC [23]). Việc tối ưu hóa rời rạc này được giải quyết bằng cách coi các biến thiết kế là các giá trị nguyên đại diện cho vị trí của các tiết diện được chọn từ một tập hợp các giá trị rời rạc cho trước. Một số nghiên cứu gần đây [24, 25, 26] về tối ưu hóa kết cấu thép sử dụng phân tích phi tuyến để mô tả chính xác hơn về ứng xử của khung thép đã được công bố. Công thức tối ưu hóa các kết cấu này đã được trình bày trong tài liệu [24] thông qua tối thiểu hóa hàm số  $W(X)$  là tổng khối lượng của kết cấu được chọn là hàm mục tiêu của bài toán và được tối thiểu hóa theo phương trình (1)

$$W(X) = \sum_{i=1}^{ng} \rho_i A(x_i) \sum_{j=1}^{n_l} L_j \quad (1)$$

Trong đó  $\rho$  là khối lượng riêng của vật liệu thép,  $A(x_i)$  là diện tích mặt cắt ngang và  $X = \{x_i\}$  vec tơ biến thiết kế là diện tích tiết diện của các thanh dầm và cột;  $i = 1, \dots, ng$  là số thứ tự của các phần tử kết cấu;  $L_j$  là chiều dài của thanh thứ  $j$  trong nhóm phần tử thứ  $i$ . Trong bài toán thiết kế có biến  $x_i$  với  $i = 1, \dots, ng$  là biến liên tục thì biến thiết kế được chọn trong khoảng giá trị cho trước. Trong bài toán thiết kế có biến là biến rời rạc thì  $x_i$  được chọn từ một tập hợp các giá trị rời rạc cho trước.

Ngoài các điều kiện ràng buộc về mặt hình học là kích thước của tiết diện dầm và cột, các điều kiện ràng buộc về chịu lực và về chuyển vị lệch tầng của khung thép theo công thức (2) như sau:

$$1 - constr_m \leq 0 \quad 1 - LF_j \leq 0 \text{ với } j=1, \dots, nst$$

$$A_i \subset [A_i] \left| \frac{d_{k,l}}{d_{k,l}^u} \right| - 1 \leq 0 \quad (2)$$

$constr_m$  là giá trị tương ứng với điều kiện ràng buộc cấu tạo thứ  $m$ ;  $LF = R/S$  là hệ số chịu tải tối đa của kết cấu  $nst$  là số lượng tổ hợp tải trọng được xem xét;  $[A]$  là tập hợp các tiết diện của phần tử kết cấu;  $d_{k,l}$  là chuyển vị lệch tầng,  $d_{k,l}^u$  là chuyển vị lệch tầng giới hạn; là tỷ lệ giữa khả năng chịu tải của công trình (R) và hệ quả do tải trọng gây ra (S).

Trên thực tế, các thuật toán Meta - heuristic không thể trực tiếp giải quyết các bài toán tối ưu có điều kiện ràng buộc. Để có thể áp dụng hiệu quả các thuật toán này, bài toán tối ưu có điều kiện ràng buộc cần phải được chuyển đổi về bài toán tối ưu không có điều kiện ràng buộc. Trong nghiên cứu này, phương pháp hàm phạt sẽ được sử dụng để thực hiện nhiệm vụ trên. Nguyên lý chính của phương pháp này là sẽ cộng thêm vào hàm mục tiêu một khối lượng gọi là giá trị phạt. Giá trị phạt này sẽ bằng 0 nếu như phương án thiết kế không vi phạm bất kỳ điều kiện ràng buộc nào. Giá trị phạt sẽ lớn hơn 0 khi phương án thiết kế đó vi phạm một điều kiện ràng buộc nào đó. Vi phạm càng nhiều thì giá trị phạt càng lớn. Nếu sử dụng một cách hợp lý giá trị phạt, giá trị hàm mục tiêu của phương án thiết kế vi phạm sẽ lớn hơn rất nhiều so với phương án thiết kế không vi phạm. Qua đó, phương án thiết kế vi phạm sẽ bị loại bỏ trong quá trình tối ưu do quá trình tối ưu là nhằm tìm ra phương án có giá trị hàm mục tiêu (khối lượng hệ khung) là bé nhất. Hàm phạt cho bài toán tối ưu khung trong công thức (3) như sau:

$$W_{im}(X) = \left( 1 + \sum_{m=1}^{N_{constr}} \alpha_{constr,m} \beta_{1,m} + \sum_{j=1}^{N_{strength}} \alpha_{str,j} \beta_{2,j} + \sum_{k=1}^{N_{service}} \alpha_{disp,k} \beta_{3,k} \right) xW(X)$$

$$\beta_{1,m} = \max(1 - constr_m, 0);$$

$$\beta_{2,j} = \max(1 - LF_j, 0);$$

$$\beta_{3,k} = \sum_{l=1}^m \max \left( \left| \frac{d_{k,l}}{d_{k,l}^u} \right| - 1 \right) \quad (3)$$

Trong đó  $l=1, \dots, N_{story}$  là số thứ tự tầng,  $j=1, \dots, N_{strength}$  và  $k=1, \dots, N_{service}$  tương ứng với tổ hợp tải trọng theo giới hạn cường độ và theo điều kiện chuyển vị; Nhằm đạt được mục đích để các cá thể vi phạm điều kiện ràng buộc sẽ tự động bị loại bỏ dần trong quá trình tối ưu, các hệ số phạt  $\alpha_{constr,m}$ ,  $\alpha_{str,j}$  và  $\alpha_{disp,k}$ . Để đảm bảo các thiết kế vi phạm điều kiện ràng buộc sẽ bị loại bỏ trong quá trình tối ưu, các giá trị hàm phạt cần chọn đủ lớn. Trong nghiên cứu này các giá trị hàm phạt được lấy bằng 10000.

### 3. MỘT SỐ THUẬT TOÁN META-HEURISTIC SỬ DỤNG TRONG BÀI TOÁN TỐI ƯU KHUNG THÉP VỚI BIẾN RỜI RẠC

#### 3.1. Thuật toán tiến hóa vi phân cơ bản (Differential Evolution - DE)

Thuật toán tối ưu DE được được Storn và Price giới thiệu lần đầu tiên vào năm 1997 cho các bài toán tối ưu với biến thiết kế là liên tục [21]. nhiều nghiên cứu đã được thực hiện để cải thiện và áp dụng DE trong việc giải quyết các vấn đề tối ưu hóa kết cấu. Các bước chính của thuật toán DE như sau:

**a) Khởi tạo:** Quần thể được hình thành từ các cá thể

$X_i = (x_j) (j=1, \dots, D)$  được lựa chọn một cách ngẫu nhiên từ miền giá trị cho trước của các biến thiết kế.  $D$  là số biến thiết kế

**b) Đột biến:** Tương ứng với từng cá thể  $X_i$ , một cá thể đột biến  $V = (v_1, v_2, \dots, v_D)$  được tạo ra dựa trên kỹ thuật đột biến DE.

Một số kỹ thuật đột biến DE thường được sử dụng là:

$$DE/rand/1: V = X_{r_1} + F \times (X_{r_2} - X_{r_3}) \quad (4)$$

$$DE/best/1: V = X_{best} + F \times (X_{r_1} - X_{r_2}) \quad (5)$$

Trong đó:  $F$  là biên độ đột biến;  $X_{best}$  là cá thể tốt nhất trong quần thể hiện tại;  $r_1, r_2$  và  $r_3$  là 3 số nguyên ngẫu nhiên được lựa chọn trong khoảng  $[1, D]$  và thỏa mãn điều kiện  $i \neq r_1 \neq r_2 \neq r_3$ . Vec tơ cơ sở  $X_{r_1}$  được chọn ngẫu nhiên trong tất cả các cá thể hiện tại nên thường hội tụ chậm và yếu trong khai thác tối ưu cục bộ. Ngược lại  $X_{best}$  chọn cá thể tốt nhất làm vec tơ cơ sở nên DE/best/1 thường nhanh hội tụ và tốt trong tìm kiếm cục bộ nhưng lại dễ bị rơi vào cục bộ địa phương hơn.

**c) Lai ghép:** Một cá thể mới  $U = (u_1, u_2, \dots, u_D)$  được tạo ra thông qua việc lai tạo giữa  $X_i$  và  $V$  thông qua cơ chế như sau:

$$u_j = \begin{cases} v_j & \text{if } (rand(0,1) < CR) \text{ or } (j = I) \\ x_{ij} & \text{otherwise} \end{cases} \quad (6)$$

Trong đó CR là tỷ lệ lai chéo không đổi được xác định trước trong đoạn  $[0,1]$  và  $i$  là số nguyên ngẫu nhiên trong khoảng  $[1, D]$

**d) Lựa chọn:** Cuối cùng, hàm mục tiêu của  $U$  được đánh giá.  $U$  được chọn để thay thế cho vị trí của  $X_i$  trong quần thể mới nếu giá trị hàm mục tiêu của nó tốt hơn của  $X_i$ .

#### 3.2. Thuật toán tiến hóa vi phân tự thích ứng AEpDE

AEpDE được xây dựng dựa trên cơ sở kết hợp hài hòa ưu điểm của hai kỹ thuật đột biến cơ bản của DE bằng cách sử dụng phương pháp 'DE/pbest/1'. Trong quá trình tối ưu hóa, bằng cách kiểm soát giá trị  $p$ , lợi ích của 'DE/rand/1' và 'DE/best/1' đều được sử dụng một cách hiệu quả. Cụ thể, một cá thể thử nghiệm  $U$  sẽ được tạo bằng cách sử dụng một cá thể ngẫu nhiên  $X_{pbest}$  ở đầu 100% quần thể như sau:

$$U = X_{pbest} + F \times (X_1 - X_2), \quad (7)$$

Trong đó  $X_1$  và  $X_2$  là hai cá thể ngẫu nhiên khác và  $F$  là một hệ số tỷ lệ. Rõ ràng, 'DE/rand/1' và 'DE/best/1' là 2 trường hợp đặc biệt của 'DE/pbest/1' khi  $p$  tương ứng bằng 1.0 và  $1/NP$ , với  $NP$  là quy mô dân số.

Cụ thể là, 'DE/rand/1', tốt trong việc khám phá tối ưu tổng thể nhưng hội tụ chậm, nên được sử dụng ở giai đoạn đầu của quá trình tối ưu hóa. Ngược lại, 'DE/best/1' với khả năng hội tụ nhanh và tìm kiếm cục bộ tốt được áp dụng trong giai đoạn sau của quá trình tối ưu hóa. Trước đây Truong và Kim [27] đã đề xuất phương trình sau để kiểm soát giá trị  $p$  trong quá trình tối ưu hóa

$$p(k) = A \times NP^{\left( \frac{-B \times (k-1)}{total\_generation-1} \right)} \quad (8)$$

Trong đó  $NP$ ;  $A$  và  $B$  là các tham số có thể kiểm soát giá trị  $p$  trong quá trình tối ưu hóa. Như có thể được quan sát trong biểu thức (8), ở thế hệ đầu tiên,  $p=A$ , do đó  $A$  kiểm soát số cá thể tốt nhất được sử dụng lúc đầu. Trong khi đó,  $B$  ảnh hưởng đến tốc độ giảm của giá trị  $p$ . Nếu  $B$  tăng, giá trị  $p$  sẽ giảm nhanh hơn và khi đó số lượng cá thể tốt nhất được sử dụng cho kỹ thuật đột biến sẽ giảm nhanh hơn. Tốc độ hội tụ của quá trình tối ưu hóa được tăng lên. Tuy nhiên, như có thể thấy trong biểu thức (8), giá trị  $p$  phụ

thuộc vào số thế hệ tối đa được xác định trước. Trong nhiều trường hợp, vấn đề này gây ra sự thay đổi của giá trị p để không phản ánh chính xác tốc độ hội tụ dân số.

Trong nghiên cứu này, kỹ thuật tự thích ứng được sử dụng để giải quyết vấn đề trên, phương pháp này được đặt tên là AEpDE. Giá trị p được tính như sau:

$$p = A \times NP^{\left(-B \times \left(\frac{TD_{(t)}}{TD_{(0)}} - 1\right)\right)} \tag{9}$$

Trong đó:

$$TD_{(t)} = \frac{1}{D} \sqrt{\frac{\sum_{i=1}^D \frac{1}{NP} \sum_{j=1}^{NP} \left(\frac{x_{j,i}}{x_i^{UB}}\right)^2}{\sum_{i=1}^D \left(\frac{1}{NP} \sum_{j=1}^{NP} \frac{x_{j,i}}{x_i^{UB}}\right)^2}} \tag{10}$$

Ở đây D là số lượng biến thiết kế;  $x_{j,i}$  là  $i^{th}$  biến thiết kế của  $j^{th}$  cá thể; và  $TD_{(t)}$  là chỉ số đa dạng quần thể ở  $t^{th}$  thế hệ;  $x_i^{UB}$  là cận

giá trị trên cho trước của biến thiết kế  $x_i$ . Chỉ số đa dạng TD như được trình bày trong biểu thức trên được đề xuất bởi Wineberg và Oppacher vào năm 2003 [28]. Phương trình này biểu thị độ lệch chuẩn trung bình của mỗi cá thể trong quần thể. Như đã trình bày trong biểu thức (9) và (10), p phụ thuộc vào  $TD_{(t)}$ .

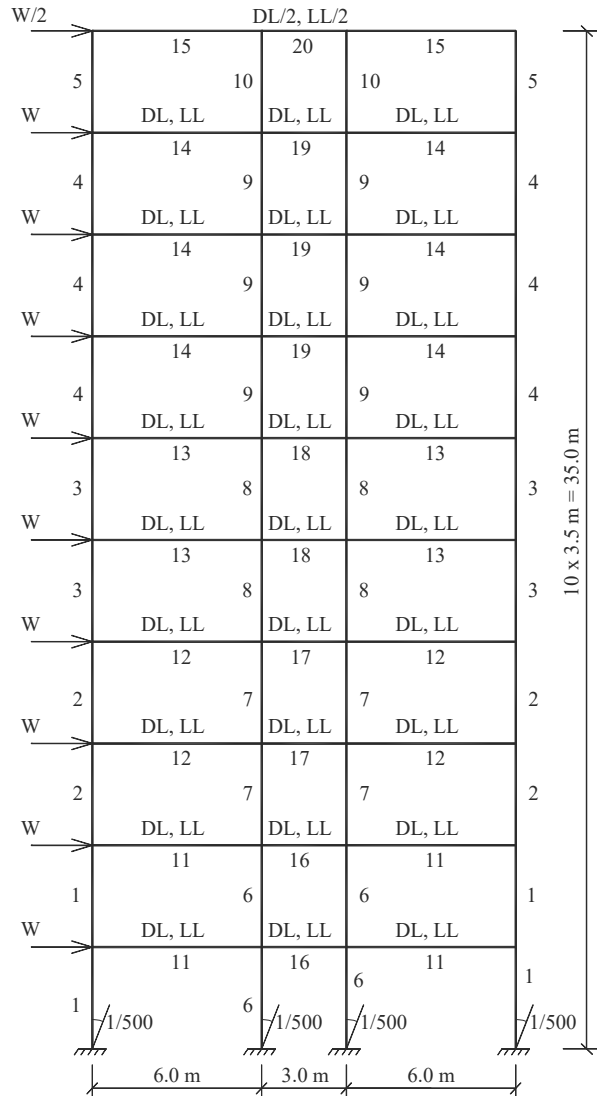
#### 4. TRƯỜNG HỢP NGHIÊN CỨU

Hiệu quả của phương pháp đề xuất được thể hiện qua khung thép ba nhịp 10 tầng 3x10 như trong Hình 1. Vật liệu thép là thép A992 với cường độ chảy 345 MPa và mô đun đàn hồi 200.000 MPa. Khung có 40 cột và 30 dầm mà các mặt cắt được chia thành 10 nhóm tiết diện dầm và 10 nhóm tiết diện cột. Tiết diện cột sử dụng từ 158 tiết diện điển hình loại W12, W14, W18, W21, W24 và W27, trong khi đối với tiết diện dầm là 267 tiết diện từ W10-W44. Tình tải và hoạt tải phân bố đồng đều tác dụng lên tất cả các dầm, trong khi tải trọng gió được quy đổi thành tải trọng tập trung ở mức sàn. Các giá trị của DL, LL và W lần lượt là 50 (kN/m), 30 (kN/m) và 35 (kN) trong đó DL, LL và W lần lượt là tải trọng tĩnh tải, hoạt tải và gió. Ba tổ hợp tải được xem xét là: 2 tổ hợp trạng thái giới hạn cường độ là (1.2DL+1.6LL), (1.2DL+1.6WL+0.5LL) và 01 tổ hợp trạng thái giới hạn sử dụng là (1.0DL+0.7WL+0.5LL) [23]. Tóm lại, bài toán tối ưu hóa này có 20 biến thiết kế với hơn 1,78E+46 hoán vị. Chỉ xem xét các ràng buộc về khả năng thi công của liên kết dầm-cột. Số ràng buộc về khả năng xây dựng, độ bền và khả năng phục vụ lần lượt là 13, 2 và 1.

Đối với tổ hợp trạng thái giới hạn sử dụng, điều kiện ràng buộc về chuyển vị là độ chuyển vị ngang cho phép và bằng H/400 với H chiều cao khung. Phân tích phi đàn hồi phi tuyến tính của khung thép được thực hiện bằng chương trình PAAP [45]. Chương trình tối ưu hóa được viết bằng ngôn ngữ Python và các thư viện phần mềm nguồn mở bao gồm Sklearn, Tensorflow và Keras. Ba thuật toán tối ưu hóa tương đối mới được sử dụng để so sánh như AEpDE, Jaya và PSO. Tổng số của tất cả các thuật toán được xem xét là bằng 25. Các tham số A và B trong AEpDE được chọn tương ứng là 1.0 và 1.0. Hệ số F = 0.7 và CR ngẫu nhiên trong khoảng giá trị [0, 1]. Jaya không yêu cầu các tham số hệ thống. Các tham số của PSO được chọn như sau bằng cách sử dụng thử và sai:  $c_1=1.5$ ,  $c_2=2.0$ ,  $w=1.0$  và  $w_{damp} = 0.995$ .

Bảng 2 trình bày kết quả tối ưu hóa của các thuật toán được xem xét, bao gồm Jaya, PSO và AEpDE. Trong số ba thuật toán siêu dữ liệu được sử dụng là Jaya, PSO và AEpDE, thuật toán tối ưu hóa được đề xuất AEpDE có hiệu suất tốt nhất vì nó tìm thấy các trọng số tối ưu nhỏ nhất, kém nhất và trung bình là 30.320,6 (kg), 30.872,7 (kg) và 30.606,9 (kg), tương ứng. Trọng lượng tốt nhất, tệ nhất và trung bình tìm thấy là 30.801,1 (kg). Jaya và PSO rất hiệu quả trong việc tối ưu hóa khung thép vì họ có thể tìm ra các thiết kế khả thi. Tuy nhiên, so với AEpDE, kết quả của Jaya và PSO tương đối kém hơn trong nghiên cứu trường hợp này. Các cuộc thảo luận ở trên chứng minh rằng thuật toán đề xuất AEpDE cạnh tranh với các phương pháp khác được xem xét.

Chất lượng của các thiết kế tối ưu được tìm thấy qua AEpDE là tốt. Điều này có nghĩa là kỹ thuật sử dụng hệ số an toàn đã thành công trong kiểm soát độ chính xác của AEpDE. Tuy nhiên, xét về nỗ lực tính toán, AEpDE tốt hơn so với các thuật toán khác nhưng vẫn còn tồn tương đối nhiều thời gian. Như có thể thấy trong Bảng 3, số lần phân tích kết cấu là 94.847 đối với AEpDE và 150.000 đối với Jaya và PSO. Kết quả là, thời gian tính toán của AEpDE, Jaya, PSO tăng dần tương ứng. Hơn nữa, Bảng 3 trình bày các thiết kế tối ưu nhất của các phương pháp tối ưu hóa được xem xét.



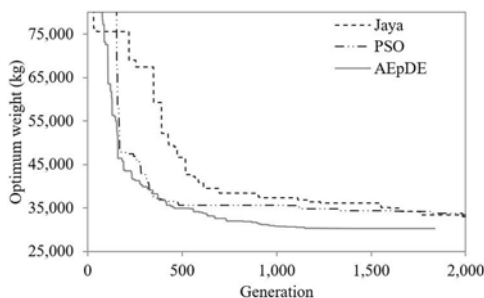
Hình 1. Sơ đồ khung thép 3 nhịp x 10 tầng

Bảng 1: Kết quả tối ưu cho khung thép 3 nhịp x 10 tầng

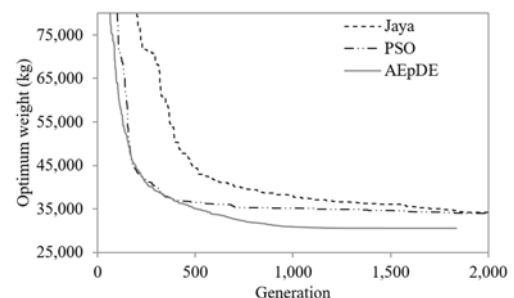
Số liệu	Jaya	PSO	AEpDE
Trọng lượng tốt nhất (kg)	33.419,8	33.028,4	30.320,6
Trọng lượng tồi tệ nhất (kg)	35.351,3	34.282,7	30.872,7
Trọng lượng trung bình (kg)	34.230,9	33.689,3	30.606,9
Tiêu chuẩn trọng lượng (kg)	707,7	514,3	167,0
Phân tích kết cấu trung bình	150.000	150.000	94,847
Thời gian tính toán trung bình (giây)	165.000	165.000	104,332
Tỷ lệ thời gian tính toán	158,1%	158,1%	100%

Bảng 2: Thuật toán thiết kế tối ưu nhất cho khung thép 3 nhịp x 10 tầng

Số liệu	Jaya	PSO	AEpDE
Tổng trọng lượng (kg)	33.419,8	33.028,4	30.320,6
Hệ số tải kết cấu cho (1.2DL+0.5LL+1.6W)	1.384	1.265	1.214
hệ số tải kết cấu cho (1.2DL+1.6LL)	1.024	1.045	1.034
Chuyển vị ngang chuẩn hóa (1.0DL+0.5LL+0.7W)	0,994	0,995	0,9996
Nhóm 1	W16x40	W16x36	W16x40
Nhóm 2	W14x38	W12x35	W16x31
Nhóm 3	W16x26	W14x38	W16x26
Nhóm 4	W12x26	W16x26	W16x26
Nhóm 5	W12x30	W14x22	W12x16
Nhóm 6	W16x89	W16x77	W16x67
Nhóm 7	W14x68	W14x61	W16x57
Nhóm 8	W12x45	W16x40	W16x40
Nhóm 9	W16x31	W16x36	W16x26
Nhóm 10	W14x34	W12x26	W12x16
Nhóm 11	W16x26	W12x30	W16x26
Nhóm 12	W14x30	W14x26	W16x26
Nhóm 13	W16x26	W16x26	W16x26
Nhóm 14	W16x26	W16x26	W16x26
Nhóm 15	W10x17	W10x17	W12x16
Nhóm 16	W24x68	W21x57	W24x55
Nhóm 17	W21x57	W21x68	W24x55
Nhóm 18	W16x31	W10x33	W21x44
Nhóm 19	W12x14	W16x26	W14x22
Nhóm 20	W10x19	W10x30	W10x12



Hình 2. Đường cong hội tụ trọng lượng tốt nhất cho khung thép 3 nhịp x 10 tầng



Hình 3. Đường cong hội tụ trọng lượng trung bình cho khung thép 3 nhịp x 10 tầng



Lịch sử hội tụ của các thuật toán được trình bày trong Hình 2 và 3, trong đó Hình 2 là tốc độ hội tụ tối ưu hóa cho thiết kế có khối lượng tốt nhất và Hình 3 là mức trung bình của tất cả các trường hợp. Tốc độ hội tụ của AEpDE tốt hơn nhiều so với Jaya và PSO. Thực tế, trong giai đoạn đầu, tốc độ hội tụ của PSO khá tốt vì nó gần giống với AEpDE. Nhưng sau đó có vẻ như dễ bị rơi vào tối ưu cục bộ khi những trường hợp tốt nhất được cải thiện rất chậm.

## 5. KẾT LUẬN

Bài báo đề xuất một thuật toán tiến hóa vi phân AEpDE cải tiến từ thuật toán DE truyền thống còn gọi là thuật toán tiến hóa vi phân tự thích ứng sử dụng cho bài toán tối ưu kết cấu khung thép nhằm tối ưu hóa khối lượng khung thép có kể đến ứng xử phi tuyến với nhiều biến thiết kế rời rạc. Trong thuật toán này số lượng cá thể tốt nhất được xem xét theo phương pháp pbest dựa trên sự đa dạng của dân số hiện tại. Ứng xử phi tuyến hình học và vật liệu của khung thép được xét đến thông qua sử dụng phân tích trực tiếp phi tuyến tính phi đàn hồi. Kết quả nghiên cứu khung phẳng 3 nhịp 10 tầng với 20 biến thiết kế rời rạc cho thấy rằng AEpDE có hiệu quả tối ưu tốt hơn so với Jaya và PSO về khối lượng khung thép. Tốc độ hội tụ của AEpDE cũng tốt hơn so với Jaya và Rao. Hướng nghiên cứu tiếp theo là xây dựng mô hình dự báo nhằm nâng cao hơn nữa hiệu suất của thuật toán AEpDE.

### Lời cảm ơn

Nghiên cứu này được tài trợ bởi Bộ Giáo dục và đào tạo trong đề tài mã số B2022-XDA-07.

### TÀI LIỆU THAM KHẢO

- [1] AISC-LRFD. Manual of steel construction – load and resistance factor design. Chicago (IL): American Institute of Steel Construction, 1999.
- [2] EN 1993-1-1, Eurocode 3. Design of steel structures – part 1-1: general rules and rules for building. Brussels: European Committee for Standardization, 2005.
- [3] V.H. Truong, S.E. Kim. A robust method for optimization of semi-rigid steel frames subject to seismic loading. *Journal of Constructional Steel Research* 2018; 145: 184-195.
- [4] V.H. Truong, H.M. Hung, P.H. Anh, T.D. Hoc. Optimization of steel moment frames with panel-zone design using an adaptive differential evolution. *Journal of Science and Technology in Civil Engineering (STCE)-HUCE* 2020; 14(2): 65-75.
- [5] H.A. Pham, V.H. Truong, T.C. Vu. Fuzzy finite element analysis for free vibration response of functionally graded semi-rigid frame structures. *Applied Mathematical Modelling* 2020; 88: 852-869.
- [6] M.H. Ha, Q.V. Vu, V.H. Truong. Optimization of nonlinear inelastic steel frames considering panel zones. *Advances in Engineering Software* 2020; 142: 102771.
- [7] V.H. Truong, Q.V. Vu, H.T. Thai, M.H. Ha. A robust method for safety evaluation of steel trusses using Gradient Tree Boosting algorithm. *Advances in Engineering Software* 147, 102825 (2020).
- [8] H.M. Hùng, T.V. Hùng, Đ.V. Thuật, V.Q. Việt. Phương pháp xử lý hiện tượng mô hình quá khớp trong xây dựng mô hình học sâu để ước lượng khả năng chịu tải của giàn phi tuyến. *Tạp chí Khoa học Công nghệ Xây dựng (KHCNXD)-ĐHXD* 14 (1V), 12-20 (2020).
- [9] V.H. Truong, S.E. Kim. Reliability-based design optimization of nonlinear inelastic trusses using improved differential evolution algorithm. *Advances in Engineering Software* 2018; 121: 59-74.
- [10] V.H. Truong, Q.V. Vu, V.T. Dinh. A deep learning-based procedure for estimation of ultimate load carrying of steel trusses using advanced analysis. *Journal of Science and Technology in Civil Engineering (STCE)-NUCE* 2019; 13(3): 113-123.
- [11] S.E. Kim, V.H. Truong. Reliability Evaluation of Semirigid Steel Frames Using Advanced Analysis. *Journal of Structural Engineering* 2020; 146(5): 04020064.
- [12] Spunt, L. (1971). *Optimum structural design*. Prentice-Hall. Englewood Cliffs, New Jersey, USA.
- [13] 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.
- [14] Saka, M. P. (2003). Optimum design of pitched roof steel frames with haunched rafters by genetic algorithm. *Computers & Structures*, 81(18-19):1967–1978.
- [15] [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.
- [16] Hasanc\_ebi, O., Bahc\_ecioglu, T., Kurc, 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.
- [17] 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.
- [18] [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.
- [19] 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.
- [20] 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
- [21] R. Storn, K. Price (1997), "Differential evolution - a simple and efficient heuristic for global optimization over continuous spaces", *Journal of Global Optimization*, 11(4), pp.341-359.
- [22] Price, K. V., Storn, R. M., Lampien, J. A. (2005). *Differential evolution: A practical approach to global optimization*. Springer, Germany.
- [23] AISC, "ANSI/AISC 360-05 Specification for Structural Steel Buildings, American Institute of Steel Construction." American Institute of Steel Construction, 2005.
- [24] V.H. Truong, S.E. Kim (2018a), "A robust method for optimization of semi-rigid steel frames subject to seismic loading", *Journal of Constructional Steel Research*, 145, pp.184-195.
- [25] H.M. Ha, Q.V. Vu, V.H. Truong (2020), "Optimization of nonlinear inelastic steel frames considering panel zones", *Advances in Engineering Software*, 142, DOI: 10.1016/j.advengsoft.2020.102771.
- [26] V.H. Truong, S.E. Kim (2017b), "An efficient method for reliability-based design optimization of nonlinear inelastic steel space frames", *Struct. Multidisc. Optim.*, 56(2), pp.331-351
- [27] Truong V.H, Kim S.E. Reliability-based design optimization of nonlinear inelastic trusses using improved differential evolution algorithm. *Advances in engineering software* 2018; 121: 59-74
- [28] Wineberg M, Oppacher F. The underlying similarity of diversity measures used in evolutionary computation. In *Genetic and Evolutionary Computation Conference*; Springer: Berlin/Heidelberg, Germany, 2003; 2724: 1493–1504.
- [29] Rao R. Jaya: a simple and new optimization algorithm for solving constrained and unconstrained optimization problems. *International Journal of Industrial Engineering Computations* 2016; 7(1):19–34.
- [30] Eberhart, R., Kennedy, J. (1995). Particle swarm optimization. *Proceedings of IEEE International Conference on Neural Networks IV*, 1942–1948.