

Sử dụng thuật toán BCMO giải bài toán tối ưu cân bằng thời gian và chi phí trong dự án xây dựng

Applications of BCMO algorithm to solve the time-cost trade-off optimization problem in construction projects

> **TRẦN VĂN NAM¹, BÙI NGUYỄN DŨNG NHÂN²**

¹Viện Kỹ thuật Công trình đặc biệt, Học viện Kỹ thuật Quân sự

²Khoa Kinh tế vận tải, Trường Đại học Công nghệ Giao thông vận tải

Email: vannamhvtqs@lqdtu.edu.vn; nhanbnd@utt.edu.vn.

TÓM TẮT

Phân tích cân bằng thời gian - chi phí là một trong những bài toán quan trọng nhất của công tác lập kế hoạch và kiểm soát dự án xây dựng. Trong số các kỹ thuật được áp dụng, nhóm phương pháp metaheuristic được đánh giá là có khả năng mạnh mẽ và giải quyết có hiệu quả cao bài toán cân bằng thời gian - chi phí. Bài báo này trình bày việc xây dựng mô hình tính cho bài toán tối ưu cân bằng thời gian - chi phí trên cơ sở áp dụng thuật toán Balancing Composite Motion Optimization (BCMO) - một thuật toán thuộc nhóm các phương pháp metaheuristic mới được giới thiệu gần đây - kết hợp với phương pháp trọng số thích ứng cải tiến. Một ví dụ được phân tích để minh họa khả năng của mô hình trong việc tạo ra các giải pháp tối ưu/gần tối ưu. Kết quả thu được từ thử nghiệm số cũng được so sánh với kết quả khi ứng dụng các thuật toán khác nhau đã được công bố trước đây, cho thấy ưu điểm của thuật toán trong việc giải quyết các bài toán tối ưu phức tạp.

Từ khóa: Cân bằng thời gian - chi phí; tối ưu; metaheuristic; phương pháp trọng số; balancing Composite Motion Optimization.

ABSTRACT

Time-cost trade-off analysis is one of the most important problems of planning and controlling construction projects. Among the applied techniques, the group of metaheuristic methods is evaluated as having a strong capability and being able to solve in high efficiency the time-cost trade-off problem. This paper presents the creation of a computing model for the time-cost trade-off optimization problem based on applying the Balancing Composite Motion Optimization (BCMO) - an algorithm belonging to a group of recently introduced metaheuristic methods - in combination with the Modified Adaptive Weight Approach. An example is analyzed to demonstrate the model's capability in generating optimal/near-optimal solutions. The results obtained from the numerical experiment are also compared with the results when applying different algorithms which were published previously, showing the algorithm's advantage in solving complex optimization problems.

Keywords: Time-cost trade-off, Optimization, Metaheuristic; Weight approach, Balancing Composite Motion Optimization.

1. GIỚI THIỆU

Bài toán cân bằng thời gian - chi phí (tên tiếng Anh: time-cost trade-off problem) truyền thống đã trở thành chủ đề nghiên cứu chuyên sâu kể từ khi phát triển phương pháp đường găng (Critical Path Method - CPM) vào cuối những năm 1950 [1]. Bản chất của bài toán là tìm cách giải quyết mâu thuẫn trong việc đạt được cả hai mục tiêu trong một dự án: thời gian ngắn và chi phí thấp. Hai mục tiêu này bị xung đột vì việc hoàn thành một nhiệm vụ trong khoảng thời gian bắt buộc (rút ngắn) dẫn đến phải sử dụng nhiều chi phí và nguồn lực trực tiếp hơn. Nhưng mặt khác, nó dẫn đến giảm tổng thời gian của dự án và các chi phí gián tiếp. Sự cân bằng giữa khung thời gian và chi phí liên quan của các

nhiệm vụ dự án được xây dựng về mặt toán học như một bài toán tối ưu hóa đa mục tiêu, cụ thể là hai mục tiêu: thời gian và chi phí (Time and Cost Optimization - TCO).

Để giải quyết bài toán TCO, các nhà nghiên cứu đã áp dụng nhiều kỹ thuật khác nhau. Theo đánh giá từ các tài liệu [2-3], cho đến nay, các phương pháp giải quyết bài toán TCO hiện tại có thể được chia thành ba nhóm: phương pháp tìm kiếm (heuristic methods), phương pháp quy hoạch toán học (mathematical programming models) và các thuật toán tối ưu dựa trên nền tảng của sự tiến hóa (evolutionary-based optimization algorithms_EOAs), cũng chính là các thuật toán metaheuristic.

Nhóm các thuật toán metaheuristic như là một giải pháp để khắc phục các thiếu sót, hạn chế của hai nhóm phương pháp đầu tiên đã nói ở trên [4]. Trong số đó, có thể kể đến thuật giải di truyền (Genetic Algorithm - GA) [5-7], thuật toán tối ưu đàn kiến (Ant Colony Optimization - ACO) [2,8-9], thuật toán tiến hóa vi phân (Differential Evolution - DE) [3], thuật toán tối ưu bầy đàn (Particle Swarm Optimization - PSO) [10-12], v.v...

Năm 2020, thuật toán Balancing Composite Motion Optimization (BCMO) được nhóm tác giả Thanh Le-Duc, Quoc-Hung Nguyen, H. Nguyen Xuan công bố trên tạp chí Information Sciences [13]. BCMO là một thuật toán tối ưu hóa metaheuristic mới được xây dựng dựa trên sự cân bằng giữa hai chuyển động tìm kiếm tổng thể và cục bộ, đã được chứng minh là mang lại kết quả có độ chính xác cao trong việc xác định giải pháp tối ưu trong các bài toán thử truyền thống và 3 bài toán thiết kế kỹ thuật thực tế. Trong bài báo này, thuật toán BCMO được phát triển cho các biến rời rạc kết hợp với phương pháp trọng số thích ứng cải tiến (Modified Adaptive Weight Approach - MAWA) để giải quyết bài toán TCO cho những dự án không lặp lại trong xây dựng.

2. TỐI ƯU ĐA MỤC TIÊU, BÀI TOÁN TCO VÀ PHƯƠNG PHÁP TRỌNG SỐ THÍCH ỨNG CẢI TIẾN

2.1. Tối ưu đa mục tiêu (Multi-Objective Optimization)

Dạng toán học của bài toán tối ưu đa mục tiêu được phát biểu như sau:

Tìm tập $\mathbf{X} = (x_1, x_2, \dots, x_D)$; với $\mathbf{x} \in R^D$;

Sao cho các hàm $f_k(\mathbf{x}) \rightarrow$ tối ưu;

Với $k = 1, 2, \dots, K$; ($K \geq 2$) (1)

Trong thực tế các hàm mục tiêu thường xung đột với nhau, có nghĩa là tất cả K hàm mục tiêu không thể đạt cực trị đồng thời. Như thế, một nghiệm tốt nhất cho bài toán là không dễ xác định. Một lời giải được cho là tốt nhất thường là sự thỏa hiệp giữa các mục tiêu và phụ thuộc vào hàm mục tiêu được đánh giá là quan trọng nhất. Vì không thể so sánh đơn giản các giải pháp với nhau, nên giải pháp "tốt nhất" được tạo ra từ việc tối ưu hóa sẽ tương ứng với lựa chọn chủ quan của người ra quyết định từ một tập giải pháp tiềm năng, xét về đặc điểm cụ thể của chúng. Tập giải pháp tối ưu tiềm năng như vậy được gọi là biên Pareto (Pareto front) và mục tiêu của tối ưu hóa đa mục tiêu là thiết lập toàn bộ biên Pareto cho bài toán thay vì một giải pháp tốt nhất [5,14].

2.2. Bài toán TCO

TCO là bài toán tối ưu hai mục tiêu, là mối quan hệ cân bằng giữa thời gian và chi phí. Trong quá trình lập kế hoạch hoặc trong trường hợp bị chậm trễ, người quản lý dự án cần phải cân bằng giữa thời gian và chi phí của dự án để nâng cao hiệu quả tổng thể. Nói cách khác, trong quá trình tối ưu hóa tương ứng, người ta cố gắng xác định phương án thực hiện từng công tác của dự án xây dựng bao gồm M công tác để đạt được lịch trình tối ưu, cùng nhau dẫn đến thời gian dự án và tổng chi phí của dự án là tối thiểu. Do đó, bài toán cân bằng thời gian - chi phí được điều chỉnh để xác định tập hợp các phương án thời gian - chi phí sẽ cung cấp lịch trình tối ưu. Thời gian và tổng chi phí của các mục tiêu dự án được tính lần lượt theo các công thức (2) và (3) [2] như sau:

$$T = \max \left[\sum_{i \in L_i} t_i^{(k)} x_i^{(k)} \right] \quad (2)$$

trong đó $t_i^{(k)}$ là thời lượng của công tác i khi thực hiện phương án thứ k , $x_i^{(k)}$ là biến chỉ số của công tác i khi thực hiện phương án thứ k : $x_i^{(k)} = 1$ khi i thực hiện phương án k và $x_i^{(k)} = 0$ trong

các trường hợp còn lại, với $\sum_{i=1}^k x_i^{(k)} = 1$. L_k là chuỗi công tác trên

đường xuyên mạng (trong sơ đồ mạng, đó đường nối các công việc, từ khởi đầu đến kết thúc dự án) thứ k và $L_k = \{i_{1k}, i_{2k}, \dots, i_{nk}\}$,

trong đó i_{jk} biểu diễn thứ tự của công tác thứ j trên đường xuyên mạng thứ k . L là tập tất cả các đường xuyên mạng và $L = \{L_k | k = 1, 2, \dots, m\}$ trong đó m là tổng số đường xuyên mạng.

$$C = \sum_{i \in M} dc_i^{(k)} x_i^{(k)} + T \times ic_i^{(k)} \quad (3)$$

trong đó $dc_i^{(k)}$ là chi phí trực tiếp của công tác i khi thực

hiện phương án thứ k , $ic_i^{(k)}$ là chi phí gián tiếp của công tác thứ i khi thực hiện theo phương án thứ k , có thể tính toán bởi các chuyên gia bằng cách ước lượng hoặc thu được từ việc chia chi phí gián tiếp của ngân sách theo tổng thời gian của hợp đồng; M là tập hợp các công tác trong mạng.

2.3. Phương pháp trọng số thích ứng cải tiến

Phương pháp trọng số là việc gán trọng số cho từng hàm mục tiêu và kết hợp chúng thành một hàm đơn đối tượng, được Zadeh [15] đề xuất lần đầu tiên vào năm 1965. Phương pháp trọng số thích ứng (Adaptive Weight Approach - AWA) được Gen và Cheng [16] áp dụng vào bài toán TCO trong xây dựng năm 2000. Tuy khắc phục được điểm yếu của các cách tiếp cận thông thường, nhưng AWA của Gen và Cheng vẫn còn một số tồn tại. Phương pháp trọng số thích ứng cải tiến (Modified Adaptive Weight Approach - MAWA) do Zheng và cộng sự đề xuất năm 2004 [5] để giải bài toán TCO trên cơ sở thuật toán GA. Một số tác giả sau này khi giải bài toán TCO bằng các thuật toán khác cũng sử dụng MAWA của Zheng [2-3,6].

Nội dung của MAWA có thể tóm tắt như sau:

Với ký hiệu: T_{obj} là giá trị hàm mục tiêu tổng thời gian; C_{obj}

là giá trị hàm mục tiêu tổng chi phí;

+ Nếu $T_{obj}^{\max} \neq T_{obj}^{\min}$ & $C_{obj}^{\max} \neq C_{obj}^{\min}$

$$v_t = \frac{T_{obj}^{\min}}{(T_{obj}^{\max} - T_{obj}^{\min})}; \quad v_c = \frac{C_{obj}^{\min}}{(C_{obj}^{\max} - C_{obj}^{\min})};$$

$$v = v_t + v_c;$$

$$w_t = \frac{v_t}{v}; \quad w_c = \frac{v_c}{v};$$

+ Nếu $T_{obj}^{\max} = T_{obj}^{\min}$ & $C_{obj}^{\max} = C_{obj}^{\min}$

$$w_t = w_c = 0,5;$$

+ Nếu $T_{obj}^{max} \neq T_{obj}^{min}$ & $C_{obj}^{max} = C_{obj}^{min}$

$w_t = 0,1; w_c = 0,9;$

+ Nếu $T_{obj}^{max} = T_{obj}^{min}$ & $C_{obj}^{max} \neq C_{obj}^{min}$

$w_t = 0,9; w_c = 0,1.$

Khi đó bài toán TCO được viết dưới dạng:

$$f(x) = w_t \frac{T_{obj}^{max} - T_{obj} + \gamma}{T_{obj}^{max} - T_{obj}^{min} + \gamma} + w_c \frac{C_{obj}^{max} - C_{obj} + \gamma}{C_{obj}^{max} - C_{obj}^{min} + \gamma} \quad (4)$$

Trong công thức (4):

$f(x)$ - giá trị mới của hàm cho mỗi cá thể khi chuyển thành một mục tiêu;

$T_{obj}^{max}, C_{obj}^{max}$ - giá trị lớn nhất của hàm thời gian và giá thành trong tập các phương án chọn (quần thể đang xét);

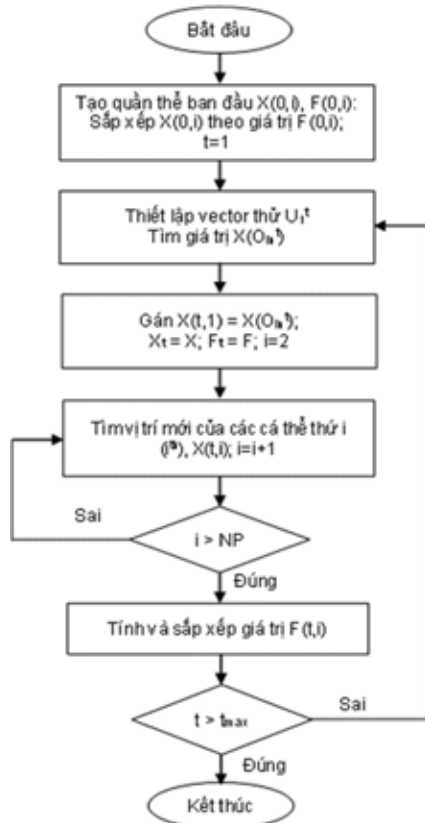
$T_{obj}^{min}, C_{obj}^{min}$ - giá trị nhỏ nhất của hàm thời gian và giá thành trong tập các phương án chọn (quần thể đang xét);

w_t, w_c - trọng số tương ứng với hàm thời gian và giá thành;

T_{obj} - tổng thời gian ứng với một nghiệm $\{x\}$ nào đó của tập nghiệm đang xét;

C_{obj} - tổng giá thành ứng với một nghiệm $\{x\}$ nào đó của tập nghiệm đang xét;

γ - số ngẫu nhiên phân bố đều trong khoảng $[0, 1]$.



Hình 1. Sơ đồ thuật toán BCMO

3. THUẬT TOÁN BCMO

BCMO được phát triển dựa trên giả thiết rằng giải pháp/nghiệm tối ưu có thể tìm được trong không gian tìm kiếm trong khi chuyển động tìm kiếm của các giải pháp ứng viên là như nhau trong tìm kiếm tổng thể cũng như tìm kiếm cục bộ. Những giả thiết này cho phép các giải pháp ứng viên tốt nhất trên mỗi thế hệ tăng cường tìm kiếm trong không gian cục bộ ban đầu hoặc chuyển sang không gian cục bộ khác để tiếp tục tìm kiếm. Một mô hình toán học xác suất được tạo ra để quản lý sự di chuyển của từng cá thể. Khi mỗi cá thể đạt được sự cân bằng giữa chuyển động tìm kiếm và khai thác của nó, thì khả năng tìm kiếm của toàn bộ quần thể cũng có thể được cân bằng, do đó có thể đạt được giải pháp tối ưu.

BCMO được tiến hành theo ba bước sau:

- Bước 1: Tạo quần thể ban đầu.

Giống như các thuật toán tối ưu Metaheuristic khác, quần thể ban đầu của BCMO được hình thành theo luật phân bố đều trong không gian nghiệm theo công thức sau:

$$x_i = x_i^L + \text{rand}(1, d) \times (x_i^U - x_i^L) \quad (5)$$

trong đó x_i^U, x_i^L là giới hạn trên và giới hạn dưới của biến thứ i ; $\text{rand}(1, d)$ là vector có độ lớn d thỏa mãn luật phân bố đều trong khoảng $[0, 1]$ và d là số tham số đầu vào.

- Bước 2: Xác định điểm tổng thể hiện tại và cá thể tốt nhất.

Trong bước này, véc tơ chuyển động của cá thể thứ i trong mỗi thế hệ đối với điểm tối ưu tổng thể O , được ký hiệu là v_i có thể được tính như sau:

$$v_i = v_{i/j} + v_j \quad (6)$$

trong đó v_j là chuyển động của cá thể thứ j trong mỗi thế hệ đối với điểm O , $v_{i/j}$ là chuyển động tương đối của cá thể i so với cá thể j .

Như vậy, v_i sẽ được tính phụ thuộc vào hai điểm trong đó có điểm O là nghiệm của bài toán nhưng chưa được xác định. Để khắc phục tình trạng này, các tác giả của thuật toán đưa ra khái niệm "điểm tối ưu tổng thể thay thế", ký hiệu là O_{in} và có thể nhận được từ công thức (7) sau đây:

$$x_{O_m}^t = \begin{cases} u_1^t & \text{if } f(u_1^t) < f(x_1^{t-1}) \\ x_1^{t-1} & \text{otherwise} \end{cases} \quad (7)$$

trong đó cá thể tốt nhất của thế hệ hiện tại u_1^t được tính toán dựa trên thông tin quần thể của thế hệ trước:

$$u_1^t = \frac{LB + UB}{2} + v_{k1/k2}^t + v_{k2/1}^t \quad (8)$$

với LB, UB lần lượt là cận dưới và cận trên của không gian tìm kiếm; $v_{k1/k2}^t$ và $v_{k2/1}^t$ tương ứng là các chuyển động giả tương đối của cá thể k_1^{th} đối với cá thể k_2^{th} và của cá thể k_2^{th} đối với cá thể tốt nhất trước đó. k_1 được chọn ngẫu nhiên trong khoảng $[2, NP]$ và $k_2 < k_1$ (NP : số cá thể trong quần thể).

- Bước 3: Tính toán chuyển động tổng hợp của cá thể trong không gian nghiệm.

Ở bước 3, chuyển động tìm kiếm tổng thể $\mathbf{v}_j$ được xác định như sau:

$$\mathbf{v}_j = \alpha_j (\mathbf{x}_{O_m} - \mathbf{x}_j) \quad (9)$$

trong đó α_j là đạo hàm bậc nhất của khoảng cách giữa $\mathbf{O}_{in}$ và cá thể thứ j , được tính như sau:

$$\alpha_j = L_{GS} \times d\mathbf{v}_j \quad (10)$$

với L_{GS} là chiều dài bước chuyển động tìm kiếm tổng thể của cá thể thứ j ; $d\mathbf{v}_j$ là vector chỉ phương, dấu của nó là âm hay dương với xác suất bằng 0,5.

Tương tự, với các công thức (9) và (10), ta tính được chuyển động tương đối của cá thể thứ i đối với cá thể thứ j :

$$\mathbf{v}_{i/j} = \alpha_{ij} (\mathbf{x}_j - \mathbf{x}_i) \quad (7)$$

Cuối cùng, vị trí của cá thể thứ i ở thế hệ tiếp theo được tính bởi công thức (11) như sau:

$$\mathbf{x}_i^{t+1} = \mathbf{x}_i^t + \mathbf{v}_{i/j} + \mathbf{v}_j \quad (11)$$

Sơ đồ nguyên lý hoạt động của thuật toán BCMO được biểu diễn trên hình 1.

4. MÔ HÌNH BÀI TOÁN TCO SỬ DỤNG THUẬT TOÁN BCMO KẾT HỢP MAWA

Trên cơ sở trình tự và sơ đồ thuật toán BCMO nêu trên, các nội dung cơ bản khi ứng dụng thuật toán BCMO kết hợp MAWA cho bài toán TCO như sau:

- **Bước 1:** Xây dựng quần thể ban đầu với NP cá thể. Trong bài toán TCO, các biến tham gia là các công tác, nhận các giá trị rời rạc trong phạm vi số lựa chọn có thể cho mỗi công tác k_i . Công thức (5) dùng cho các biến liên tục sẽ được chuyển đổi để tính với các biến rời rạc: $\mathbf{x}_i = \mathbf{randi}(k_i)$. Tương ứng sẽ có được thời gian và chi phí cho từng công tác thành phần. Sau khi được NP cá thể phù hợp, quần thể ban đầu $X[0, i]$ được xây dựng xong. Tính T_{obj} và C_{obj} của các cá thể theo các biểu thức (2) và (3). Xác định T_{obj}^{max} & C_{obj}^{max} và T_{obj}^{min} & C_{obj}^{min} của quần thể rồi tính trọng số w_t, w_c theo MAWA. Tính giá trị hàm mục tiêu $F(0, i)$ cho mỗi cá thể theo công thức (4) và sắp xếp $X[0, i]$ theo thứ tự giá trị $F(0, i)$. Lưu ý tính chất của hàm $f(x)$ theo (4) của bài toán TCO để lấy $F(0, i)$ là nghịch đảo của $f(x)$ cho phù hợp với các mô tả trong thuật toán ở mục 3.

- **Bước 2:** Thiết lập vector thử U_1^t theo công thức (8). Tính T_{obj} và C_{obj} của U_1^t rồi tính giá trị hàm mục tiêu $F(U_1^t)$ với các trọng số ở trên. Xác định điểm tối ưu tổng thể thay thế O_m^t theo công thức (7).

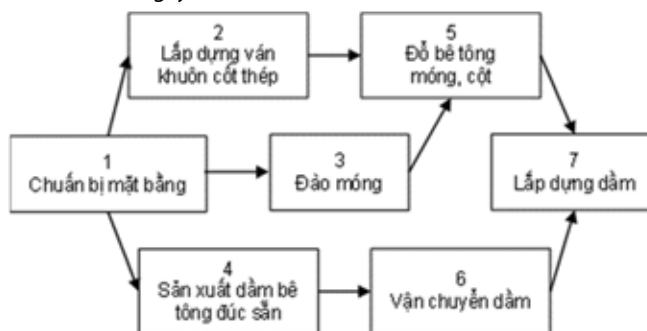
- **Bước 3:** Cho $X(t, 1)$ nhận giá trị $X(O_m^t)$. Tính toán chuyển động tổng hợp của cá thể trong không gian nghiệm để tìm vị trí mới của các cá thể từ $2 \leq NP$. Tính trọng số của quần thể mới theo MAWA rồi tính giá trị hàm mục tiêu $F(t, i)$ cho mỗi cá thể và sắp xếp theo thứ tự giá trị $F(t, i)$.

Bước 2 và bước 3 nằm trong vòng lặp với số lần lặp t_{max} được dự kiến trước. Nếu điều kiện dừng thỏa mãn thì kết thúc tính toán và in kết quả cuối cùng.

5. VÍ DỤ SỐ

5.1. Số liệu của bài toán

Trên cơ sở mô hình tính như trên, các tác giả lập trình để có chương trình tính (đặt tên là TCO_BCMO) và tính toán cụ thể cho bài toán 7 công tác đã được giải bằng các thuật toán GA [5], ACO [2] và DE [3]. Sơ đồ mạng của bài toán như trên hình 2, số liệu về thời gian và chi phí liên quan tới các lựa chọn của từng công tác của dự án được cho trong bảng 1. Bên cạnh đó, chi phí gián tiếp được giả định là 1.500\$ mỗi ngày.



Hình 2. Sơ đồ mạng của bài toán ví dụ số

Bảng 1. Các lựa chọn công việc của bài toán ví dụ số

ID	Lựa chọn	Thời gian (ngày)	Chi phí trực tiếp (\$)
1	1	14	23.000
	2	20	18.000
	3	24	12.000
2	1	15	3.000
	2	18	2.400
	3	20	1.800
	4	23	1.500
	5	25	1.000
3	1	15	4.500
	2	22	4.000
	3	33	3.200
4	1	12	45.000
	2	16	35.000
	3	20	30.000
5	1	22	20.000
	2	24	17.500
	3	28	15.000
	4	30	10.000
6	1	14	40.000
	2	18	32.000
	3	24	18.000
7	1	9	30.000
	2	15	24.000
	3	18	22.000

Ở bài toán này, trong bài báo [2] đã chỉ ra điểm lý tưởng z^- (điểm đồng thời có thời gian ngắn nhất và chi phí thấp nhất) và điểm không mong muốn nhất z^+ (điểm đồng thời có thời gian dài nhất và chi phí lớn nhất): $z^-(\text{time, cost}) = (60, 220.500)$; $z^+(\text{time, cost}) = (105, 323.000)$.

5.2. Kết quả và bàn luận

Chạy chương trình TCO_BCMO với các tham số được thiết lập như sau: số cá thể trong quần thể NP=30, số lần lặp 50. Kết quả giải

Bảng 2. Kết quả giải bài toán theo GA, ACO, DE và BCMO

STT	TCO_GA [5]		TCO_ACO [2]		TCO_DE [3]		TCO_BCMO	
	Thời gian	Chi phí	Thời gian	Chi phí	Thời gian	Chi phí	Thời gian	Chi phí
1	66	236.500	60	233.500	60	223.500	62	233.000
2	73	251.500	62	233.000	61	225.000	63	225.500
3	84	251.000	63	225.500	63	220.400	67	224.000
4			67	224.000	63	218.500	68	220.500

Đối với kết quả của nghiên cứu với thuật toán DE [3], các lời giải nhận được với sự bổ sung ràng buộc: đó là thời gian dự án không vượt quá 70 ngày (thời hạn hợp đồng) và thưởng-phạt cho hoàn thành dự án sớm hoặc muộn là 1.000\$/ngày. Do đó, thông tin về các kết quả của TCO_DE khi đưa vào trong bảng 2 phải thêm cột tiền thưởng để cộng thêm vào chi phí khi so sánh với kết quả từ các thuật toán còn lại. Tuy nhiên, cũng dễ nhận ra rằng, do tác động của điều kiện thời hạn hợp đồng, các phương án nhận được từ TCO_DE đều thiên về thời gian ngắn nhất (60, 61 và 63 ngày) và nó tương đồng với TCO_ACO ở hai phương án 60 và 63 ngày.

Từ bảng 2 cho thấy TCO_BCMO cho các kết quả tương tự như TCO_ACO và đều là các kết quả thuộc biên Pareto. Tuy nhiên, TCO_BCMO chưa thấy cho phương án 60 ngày (thời gian ngắn nhất có thể). Ngược lại, nó cho phương án 68 ngày với chi phí 220.500\$, có tập các lựa chọn công việc [1,1,1,3,4,3,1] tương ứng với thời gian các công việc [14,15,15,20,30,24,9](ngày) và chi phí các công việc [23.000,3.000,4.500,30.000,10.000,18.000,30.000](\$). Trong điều kiện không có ràng buộc về thời gian hợp đồng và chi phí thưởng-phạt, đây là phương án với thời gian tương đối ngắn và có chi phí thấp nhất.

Bảng 3. Kết quả giải bài toán khi có ràng buộc thời hạn hợp đồng và thưởng-phạt

STT	TCO_DE		TCO_BCMO	
	Thời gian	Chi phí	Thời gian	Chi phí
1	60	223.500	60	223.500
2	61	225.000	61	225.000
3	63	220.400	63	220.400
4	63	218.500	63	218.500
5			65	222.900
6			67	221.000
7			68	218.500

Thử nghiệm chạy chương trình với các điều kiện thêm vào về thời hạn hợp đồng kèm theo thưởng-phạt, chúng tôi cũng nhận được kết quả tương tự như của TCO_DE (xem bảng 3), nghĩa là khi đó xuất hiện phương án 60 ngày và thiên về các phương án có thời gian ngắn nhất (60, 61 và 63 ngày). Các phương án dài ngày hơn (65, 67, 68 ngày) xuất hiện nhưng với tần suất thấp. Trong số các phương án mà TCO_BCMO đưa ra, có thể đánh giá phương án 68 ngày với chi phí 218.500\$ là phương án có ý nghĩa thực tế cao, bởi vì nó vẫn nằm trong thời hạn hợp đồng, có chi phí thấp (bằng với phương án 63 ngày) nhưng nhà thầu lại không bị căng thẳng về thời gian thực hiện.

Vì không có điều kiện để lập trình và chạy các chương trình khác trên cùng một nền tảng máy tính nên ở đây không so sánh về tốc độ tính toán. Tuy nhiên, từ các tham số được thiết lập nêu trên cũng có thể đánh giá rằng BCMO trong bài toán TCO tương đương với các thuật toán khác về tốc độ tính toán.

bài toán theo thuật toán BCMO và so sánh với kết quả từ các thuật giải khác được trình bày trên bảng 2.

6. KẾT LUẬN

Với sự cân bằng giữa tìm kiếm tổng thể và cục bộ, BCMO là một thuật toán tối ưu đầy hứa hẹn, đặc biệt trong trường hợp các hàm tối ưu với các biến thiết kế lớn. Trong thuật toán không dùng tham số điều khiển quá trình tìm nghiệm, do vậy tạo điều kiện thuận lợi cho người dùng. Việc áp dụng thuật toán BCMO vào giải bài toán tối ưu cân bằng thời gian - chi phí với các kết quả trên đã củng cố thêm nhận định về khả năng và ưu thế của thuật toán. Các phương án được đề xuất từ mô hình TCO_BCMO là phù hợp và có phần hợp lý hơn kết quả thu được từ các mô hình metaheuristic khác. Điều đó cho phép kỳ vọng vào sự phát triển của thuật toán để giải các bài toán phức tạp hơn trong các lĩnh vực khác nhau.

TÀI LIỆU THAM KHẢO

1. Afshar, A., Kaveh, A., & Shoghli, O. R. (2007). "Multi-objective optimization of time-cost-quality using multi-colony ant algorithm". *Asian Journal of Civil Engineering (Building and Housing)*, 8(2), 113-124.
2. Ya-ping, K., & Ying, X. (2006). "Construction time-cost trade-off analysis using ant colony optimization algorithm". In *2006 International Conference on Management Science and Engineering* (pp. 2039-2044). IEEE.
3. Nguyễn Quán Thắng, Bùi Đức Năng và Nguyễn Thị Tuyết Mai (2014). "Sử dụng thuật toán DE giải bài toán tối ưu cân bằng thời gian và chi phí trong quản lý dự án xây dựng". *Tạp chí Kết cấu và Công nghệ xây dựng*, số 15 (III-2014), tr.65-72).
4. Toğan, V., & Eirgash, M. A. (2018). "Time-cost trade-off optimization with a new initial population approach". *Teknik Dergi*, 30(6), 9561-9580.
5. Zheng DXM, Ng ST, Kumaraswamy MM (2004). "Applying a genetic algorithm-based multiobjective approach for time-cost optimization". *J Constr Eng Manage*; 130(2):168-76.
6. Parveen S, Saha SK (2012). "GA based multi-objective time-cost optimization in a project with resources consideration". *Int J Modern Eng Res*;2(6):4352-9.
7. Agdas D, Warne DJ, Osio-Norgaard J, Forrest JM (2018). "Utility of genetic algorithms for solving large-scale construction time-cost trade-off problems". *J Comput Civil Eng*;32(1):04017072.
8. Ng ST, Zhang YS (2008). "Optimizing construction time and cost using ant colony optimization approach". *J Constr Eng Manage*;134(9):721-8.
9. Li Y, Wang S, He Y (2020). "Multi-objective optimization of construction project based on improved ant colony algorithm". *Technical Gazette*;27(1):184-90.
10. Prascevic N, Prascevic Z (2014). "Application of particle swarms for project time-cost optimization". *Građevinar*;66(12):1097-107.
11. Elbeltagi E, Ammar M, Sanad H, Kassab M (2016). "Overall multiobjective optimization of construction projects scheduling using particle swarm". *Eng, Constr Arch Manage*;23(3):265-82.
12. Aminbakhsh S, Sonmez R (2017). "Pareto front particle swarm optimizer for discrete timecost trade-off problem". *J Comput Civil Eng*;31(1):1-10.
13. Thang Le-Duc, Quoc-Hung Nguyen, H. Nguyen Xuan (2020). "Balancing composite motion optimization". *Information Sciences*, 520, 250-270.
14. Gen, M., and Cheng, R. (2000). *Genetic algorithms & engineering optimization*, Wiley-Interscience, New York.
15. Zadeh, L. A. (1965). "Fuzzy sets". *Inf. Control*, 8, 338-353.