Tạp chí Tin học và Điều khiển học, T.25, S.3 (2009), 231–245

CÁC KỸ THUẬT LAI GHÉP TRONG GIẢI THUẬT DI TRUYỀN GIẢI BÀI TOÁN CÂY KHUNG NHỎ NHẤT VỚI ĐƯỜNG KÍNH BỊ CHẶN

HUỲNH THỊ THANH BÌNH, NGUYỄN ĐỨC NGHĨA

Viện Công nghệ thông tin và Truyền thông, Đại học Bách khoa Hà Nội

Abstract. Given a connected, weighted, undirected graph $G = (V, E)$ and a positive number D. The Bounded Diameter Minimum Spanning Tree problem ($BDMST$) is to find spanning tree on G with the smallest weight, in which no path between two vertices contains more than D edges. This problem is NP-hard for $4 \leq D < |V| - 1$. This paper proposes multi-parent recombination operator using three different methods to choose parents for crossover in genetic algorithm for solving Bounded Diameter Minimum Spanning Tree Problem. Three methods to choose the edges in parents are also applied. Results of computational experiments are reported to show the efficiency of proposed parent selections.

Tóm tắt. Cho đồ thị vô hướng, liên thông, có trọng số $G = (V, E)$ và một số nguyên dương D. Bài toán cây khung nhỏ nhất với đường kính bị chặn ($BDMST$) đòi hỏi tìm cây khung có trọng số nhỏ nhất trong số các cây khung có đường kính không vượt quá D. $BDMST$ là bài toán NP-khó với $4 \leq D < |V| - 1$. Bài báo này giới thiệu kỹ thuật lai ghép đa cha mẹ sử dụng ba phương pháp chọn cha mẹ khác nhau cùng với ba phương pháp chọn cạnh để lai ghép trong giải thuật di truyền giải bài toán $BDMST$. Kết quả thực nghiệm đã tiến hành cho thấy hiệu quả của thuật toán di truyền với kĩ thuật lựa chọn cha mẹ được đề xuất.

1. MỞ ĐẦU

Bài toán cây khung nhỏ nhất với đường kính bị chặn -Bounded Diameter Minimum Spanning Tree ($BDMST$) là bài toán tối ưu tổ hợp có nhiều ứng dụng trong thiết kế mạng [5], nén dữ liệu và trong các thuật toán liên quan tới hệ phân tán [4, 26]. Phát biểu đầy đủ và các ứng dụng của $BDMST$ được giới thiệu chi tiết trong luận văn tiến sỹ của Abdalla [2].

Trước khi phát biểu bài toán $BDMST$, ta nhắc lại một số khái niệm cơ bản liên quan tới đường kính của cây và tâm. Cho một cây khung T, độ lệch lớn nhất của một đỉnh v là độ dài (tính bằng số cạnh) của đường đi đơn dài nhất từ v tới các đỉnh khác. Đường kính của cây T, ký hiệu $diam(T)$, là độ lệch lớn nhất của các nút trên cây T (độ dài đường đi dài nhất giữa hai đỉnh bất kỳ trên cây T). Giả sử đường kính của cây được xác định bởi đường đi $v_0, v_1, v_2, , v_{\lfloor \frac{k}{2} \rfloor}, v_{\lfloor \frac{k}{2} \rfloor + 1}, , v_k$. Nếu k chẵn thì $v_{\lfloor \frac{k}{2} \rfloor}$ là tâm của cây. Nếu k lẻ thì $v_{\lfloor \frac{k}{2} \rfloor}]$ và $v_{\lfloor \frac{k}{2} \rfloor + 1}$ là tâm của cây, đồng thời trong trường hợp này, cạnh $(v_{\lfloor \frac{k}{2} \rfloor}], v_{\lfloor \frac{k}{2} \rfloor + 1})$ được gọi là cạnh tâm.

Cho $G = (V, E)$ là một đồ thị vô hướng liên thông với trọng số dương trên các cạnh $w(e)$, $e \in E$. Bài toán $BDMST$ có thể phát biểu như sau: trong số các cây khung của đồ thị G có đường kính không vượt quá D, hãy tìm cây khung có trọng số nhỏ nhất (trọng số của cây khung là tổng trọng số trên các cạnh của nó). Không giảm tổng quát, giả sử rằng G là đồ thị đầy đủ.

Mô hình toán học của bài toán có thể phát biểu như sau:

Tìm cực tiểu hàm

$$\mathrm{W}(T) = \sum_{e \in T} \mathrm{w}(e)$$

với điều kiện, $T = (V, E)$ - cây khung của đồ thị G; $diam(T) \leq D$.

Bài toán đã được chứng minh là NP-khó khi $4 \leq D < |V| - 1$ (xem [6]). Hơn thế nữa, việc giải xấp xỉ $BDMST$ cũng là bài toán khó. Trong [18] đã chứng minh là không tồn tại thuật toán đa thức tìm lời giải $log|V|$ xấp xỉ, ngoại trừ $P = NP$. Như vậy, các kỹ thuật heuristic và meta-heuristic là các kỹ thuật chấp nhận được để cải tiến chất lượng lời giải cho bài toán $BDMST$, đặc biệt là khi $|V|$ lớn.

Bài báo trình bày kỹ thuật lai ghép đa cha mẹ sử dụng ba phương pháp chọn cha mẹ khác nhau để lai ghép trong giải thuật di truyền giải bài toán $BDMST$. Kỹ thuật lai ghép đa cha mẹ đề xuất sử dụng nhiều hơn hai cha mẹ để lai ghép ra một con. Ba phương pháp lựa chọn cha mẹ được trình bày trong bài: dựa trên khoảng cách Levenshtein giữa các cá thể để chọn cha mẹ, chọn những cá thể tốt nhất trong quần thể đem đi lai ghép, chọn các cá thể ngẫu nhiên trong quần thể để lai ghép. Bài báo cũng so sánh kết quả giải thuật di truyền sử dụng ba phương pháp chọn cha mẹ khác nhau để lai ghép cùng với giải thuật di truyền sử dụng toán tử lai ghép hai cha mẹ trong [25] của Raidl và Julstrom để cho thấy sự khác nhau giữa các toán tử lai ghép. Với mỗi phương pháp chọn cha mẹ, bài báo cũng thử nghiệm với số cha mẹ khác nhau để cho thấy sự khác biệt không những về phương pháp chọn cha mẹ mà cả về số lượng cha mẹ được chọn trong giải thuật di truyền giải bài toán $BDMST$.

Bài báo được trình bày trong 5 mục. Mục 2 trình bày tổng quan về bài toán $BDMST$. Mục 3 mô tả phép toán lai ghép đề xuất để giải bài toán $BDMST$. Các kết quả thực nghiệm được trình bày trong Mục 4. Mục 5 trình bày kết luận và hướng phát triển.

2. TỔNG QUAN VỀ BÀI TOÁN *BDMST*

Tổng quan về bài toán *BDMST*

Các kỹ thuật để giải bài toán $BDMST$ được chia làm hai hướng: giải chính xác và giải gần đúng. Cách tiếp cận để giải chính xác bài toán $BDMST$ dựa trên quy hoạch tuyến tính nguyên [3, 8]. Gần đây, Gruber và Raidl trình bày giải thuật nhánh và cắt dựa trên quy hoạch tuyến tính nguyên 0-1 [9]. Tuy nhiên, các cách tiếp cận này chỉ giải được bài toán có kích thước nhỏ (chẳng hạn đồ thị có kích thước nhỏ hơn 100 đỉnh).

Abdalla [1] đã giới thiệu giải thuật heuristic tham lam, One Time Tree Construction

($OTTC$), để giải bài toán $BDMST$. $OTTC$ được dựa trên thuật toán Prim [23]. Giải thuật bắt đầu từ một tập các đỉnh được chọn ngẫu nhiên. Tập đỉnh này được lặp liên tục và mở rộng bằng cách thêm đỉnh mới gần nó nhất vào nếu đỉnh này kết nạp mà không bị vi phạm về ràng buộc đường kính. Thuật toán này có độ phức tạp $O(n^3)$. Thuật toán phụ thuộc nhiều vào đỉnh khởi tạo [28].

Raidl và Julstrom [25] đã đề xuất một giải thuật cải tiến của giải thuật $OTTC$, với tên gọi Random Greedy Heuristics (RGH). RGH bắt đầu từ tâm bằng cách chọn ngẫu nhiên một đỉnh và coi đó là tâm cố định trong suốt giải thuật. Sau đó, thuật toán lặp lại các bước thêm đỉnh mới vào cây khung. Đỉnh mới này được chọn ngẫu nhiên trong tập đỉnh còn lại và được nối vào cây khung sao cho cạnh nối có trọng số nhỏ nhất mà vẫn đảm bảo ràng buộc về đường kính của cây.

Raidl và Julstrom cũng đề xuất giải thuật di truyền sử dụng mã hóa danh sách cạnh [25] và mã hóa hoán vị [15] để giải bài toán $BDMST$. Mã hóa hoán vị được đánh giá là có kết quả tốt hơn mã hóa danh sách cạnh nhưng tốn nhiều thời gian chạy hơn. Một giải thuật di truyền khác được giới thiệu trong [16] dựa trên mã hóa khóa ngẫu nhiên (random key). Giải thuật này cũng tương tự như giải thuật di truyền mã hóa hoán vị. Trong [10], Gruber sử dụng bốn kiểu lân cận để thực hiện tìm kiếm cục bộ lân cận biến thiên giải bài toán $BDMST$. Ngay sau đó, Raidl và Julstrom [17] sử dụng các kiểu lân cận biến thiên như trong [16] và nhúng vào giải thuật tối ưu hóa đàn kiến (Ant Colony Optimization - ACO) và giải thuật di truyền để giải bài toán $BDMST$. Cả hai giải thuật để xuất (ACO và GA) đều sử dụng cấu trúc tìm kiếm cục bộ lân cận biến thiên để nâng cao chất lượng lời giải. Nghĩa và Bình [20] đề xuất kỹ thuật lai ghép đa cha mẹ trong giải thuật di truyền để giải bài toán $BDMST$.

Gần đây, Singh và Gupta [28] đề xuất hai cải tiến của giải thuật RGH heuristics (trình bày trong [25]) và giải thuật di truyền giải bài toán $BDMST$ (tên gọi trong bài báo là $PEA-I$). Giải thuật $RGH-I$ trình bày trong [28], với tên gọi RGH_1 trong bài báo này, sẽ lặp để nâng cao chất lượng từ lời giải tìm được từ giải thuật RGH bằng cách sử dụng đột biến thay thế cạnh tham lam. Trong tài liệu [28] chỉ ra rằng $RGH-I$ có kết quả tốt hơn các kết quả thu được từ các giải thuật heuristic đã biết để giải bài toán $BDMST$. $PEA-I$ sử dụng mã hóa hoán vị. $PEA-I$ cho kết quả khá tốt trong số các giải thuật giải bài toán $BDMST$. Trong [12], Bình và các tác giả đã thực hiện một biến thể khác của giải thuật RGH, với tên gọi RGH_2. RGH_2 tương tự như RGH nhưng khi kết nạp một đỉnh mới thuật toán sẽ chọn ngẫu nhiêu trong tập đỉnh còn lại và nối với một đỉnh ngẫu nhiên trong tập đỉnh đã kết nạp mà vẫn đảm bảo ràng buộc về đường kính của cây. Bình và các tác giả cũng đề xuất giải thuật di truyền lai đa quần thể, trong đó mỗi quần thể được khởi tạo bởi các giải thuật heuristic khác nhau.

Kết quả của giải thuật di truyền phụ thuộc nhiều vào các tham số: toán tử lai ghép, toán tử đột biến, xác suất lai ghép, xác suất đột biến và kích thước quần thể Đã có nhiều bài báo nghiên cứu về giải thuật di truyền giải bài toán $BDMST$ [1, 2, 8, 9, 10, 11, 12, 21, 13, 14, 15, 16, 17, 18, 28] và các bài toán NP-khó khác nhưng chưa có công trình nghiên cứu nào tìm hiểu về sự khác nhau giữa các kỹ thuật lai ghép. Vì vậy, bài báo này trình bày kỹ thuật lai ghép đa cha mẹ sử dụng ba phương pháp chọn cha mẹ giải thuật di truyền giải bài toán $BDMST$ cùng với ba cách chọn cạnh từ cha mẹ để kết nạp vào cây con. Hy vọng rằng các phương pháp đề xuất không chỉ hiệu quả đối với bài toán được xét trong bài báo này mà còn có thể áp dụng tốt với các bài toán khác.

3. GIẢI THUẬT DI TRUYỀN ĐỀ XUẤT

3.1. Biểu diễn cá thể

Các phương pháp biểu diễn cây khung đã được nghiên cứu về lý thuyết và áp dụng trong các bài toán cụ thể rất nhiều [22, 7, 27]. Đối với bài toán $BDMST$, ba phương pháp biểu diễn được quan tâm: mã hóa danh sách cạnh (edge-set-coded [24, 25]), mã hóa hoán vị (permutation coded [15, 28]), và mã hóa khóa ngẫu nhiên (random key representations [16]). Bài báo này sử dụng phương pháp mã hóa danh sách cạnh để biểu diễn các nhiễm sắc thể như trong [24] đã trình bày.

3.2. Khởi tạo

Sử dụng các giải thuật heurisitc $OTTC, RGH, RGH_1, RGH_2$ được trình bày ở trên để khởi tạo quần thể.

3.3. Toán tử lai ghép

Trong giải thuật di truyền, toán tử lai ghép sẽ tạo ra con mới kế thừa các tính chất của cha mẹ. Trong phép toán lai ghép truyền thống, con sẽ được tạo ra từ hai cha mẹ nhưng trong toán tử lai ghép đề xuất ở đây, con sẽ được tạo từ nhiều cha mẹ nhằm kế thừa nhiều tính chất tốt của cha mẹ.

Trước khi trình bày thuật toán, ta đưa ra một số ký hiệu sau. V là tập đỉnh của đồ thị. $T[i]$ là cây thứ i trong tập cha mẹ. T_c là tập cạnh của cây con, C là tập đỉnh của cây có độ sâu nhỏ hơn $\frac{D}{2}$, U là tập các đỉnh chưa kết nạp vào cây.

Sơ đồ của phép toán lai ghép đa cha mẹ.

F_i: the edges in i parents ($i \in \{1..k\}$)

$T \leftarrow \emptyset$;

\\ Determine center

If D is even **then**

  $v_0 \leftarrow$ random center in $T[i]$;

  $U \leftarrow V - \{v_0\}$;

  $C \leftarrow \{v_0\}$;

  $depth[v_0] \leftarrow 0$;

Else

  $(v_0, v_1) \leftarrow$ random center in $T[i]$;

  $T \leftarrow \{(v_0, v_1)\}$;

$U \leftarrow V - \{\mathrm{v}_0,\mathrm{v}_1\}$;
$C \leftarrow \{\mathrm{v}_0,\mathrm{v}_1\}$;
$depth[v_0] \leftarrow 0$;
$depth[v_1] \leftarrow 0$;
\\ Loop for adding edge into tree A_k: the edges in F_k which connect to the centers
While $U \neq \emptyset$ **do**
Choose the edge;
$T \leftarrow T \cup \{(\mathrm{u},\mathrm{v})\}$;
$U \leftarrow U - \{v\}$;
$depth[v] \leftarrow depth[u] + 1$;
If $depth[v] < \lfloor \frac{D}{2} \rfloor$ **then**
$A_k = A_k \cup$ the edge in F_k which connect to v;
$C \leftarrow C \cup \{v\}$;
Return T

Đỉnh bắt đầu v_0 được chọn ngẫu nhiên trong $T[k]$. Nếu D chẵn, v_0 là một tâm. Nếu D lẻ, đỉnh v_1 sẽ được chọn ngẫu nhiên trong $T[k]$ và cạnh (v_0, v_1) sẽ là cạnh tâm.

Trong toán tử lai ghép, phương pháp chọn cha mẹ và chọn cạnh kết nạp vào cây con đóng vai trò rất quan trọng. Bài báo này đề xuất ba phương pháp chọn cha mẹ như sau.

1. Phương pháp b: Sắp xếp các cá thể trong quần thể tăng dần theo trọng số. F là tập x cá thể đầu tiên. Chọn ngẫu nhiên k cá thể trong tập F đem đi lai ghép. (toán tử lai ghép k_1).

2. Phương pháp r: Chọn ngẫu nhiên k cá thể trong quần thể đem đi lai ghép (toán tử lai ghép k_2).

3. Phương pháp l: Sắp xếp các cá thể trong quần thể theo thứ tự tăng dần của trọng số. F là tập x cá thể đầu tiên. Chọn k cá thể đem đi lai ghép dựa trên khoảng cách Levenshtein giữa chúng. (toán tử lai ghép k_3).

Tuy nhiên, việc chọn cạnh trong cha hoặc mẹ để kết nạp vào cây con cũng đóng vai trò quan trọng. Vì vậy, bài báo cũng thử nghiệm với ba phương pháp chọn cạnh từ cha mẹ để kết nạp vào cây con như sau:

1. Chọn cạnh ngẫu nhiên từ cha mẹ (d)

   Chọn $(u, v) \in A_i$ ngẫu nhiên: $depth[u] < \lfloor \frac{D}{2} \rfloor$

2. Chọn cạnh nhỏ nhất (m)

   Chọn $(u, v) \in A_i : min\{c\,[i, j]\,, i \in C, j \in U, depth[i] < \lfloor \frac{D}{2} \rfloor\}$;

3. Chọn cạnh nhỏ nhất trong lớp cạnh chung lớn nhất (g)

   $k \leftarrow$ số cha mẹ đem đi lai ghép;

Giải thuật di truyền sử dụng các phương pháp chọn cha mẹ và chọn cạnh như đã trình bày ở trên để giải bài toán $BDMST$ được ký hiệu là $EA - xyk$. Trong đó:

x: phương pháp chọn cha mẹ ($x = b, r, l$);

y: phương pháp chọn cạnh trong cha mẹ ($y = d, m, g$);

k: số cha mẹ.

3.4. Toán tử đột biến

Sử dụng bốn toán tử đột biến: đột biến xoá cạnh, đột biến dịch chuyển tâm, đột biến thay thế cạnh tham lam, đột biến tối ưu hoá cây con như trong [12].

4. CÁC KẾT QUẢ THỰC NGHIỆM

4.1. Dữ liệu thử nghiệm

Dữ liệu thử nghiệm trong bài báo này để giải bài toán $BDMST$ đã được sử dụng trong [12, 21, 13, 14, 15, 16, 17, 20, 25, 28]. Các dữ liệu này là đồ thị đầy đủ Euclidean và có thể được tải về từ http://tomandtun.googlepages.com/phd. Bài báo chọn năm bộ dữ liệu đầu tiên cho các kích thước đồ thị $n = 100, 250, 500,$ và 1000 (số đỉnh của đồ thị). Đường kính bị chặn tương ứng là 10, 15, 20, 25 (tổng số có 20 bộ dữ liệu).

4.2. Các thực nghiệm

Bài báo thử nghiệm hai thực nghiệm. Đầu tiên, so sánh các giải thuật di truyền $EA-xyk$ với cùng số cha mẹ (k) nhưng phương pháp chọn cha mẹ và cách chọn cạnh trong cha mẹ khác nhau để giải bài toán $BDMST$. Bài báo thử nghiệm lần lượt với các giá trị k khác nhau 2, 5, 7, 9 và với ba phương pháp chọn cha mẹ và ba cách chọn cạnh trong cha mẹ như đã trình bày trong phần trên. Trong thực nghiệm thứ 2, bài báo so sánh các giải thuật di truyền $EA - xyk$ giải bài toán $BDMST$ với số cha mẹ khác nhau.

4.3. Tham số thực nghiệm

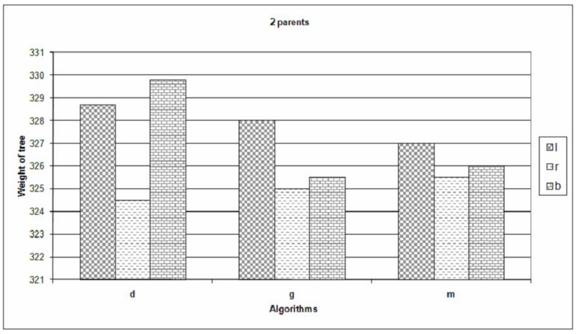
Các giải thuật di truyền thử nghiệm giải bài toán $BDMST$ được khởi tạo bằng các thuật toán heuristic $OTTC$, RGH, RGH_1, RGH_2 với tỷ lệ tương ứng là 25%. Kích thước quần thể và số thế hệ tương ứng là 100 và 500. Số cha mẹ (k) được chọn là 2, 5, 7, 9. Các quần thể được lựa chọn trận đấu kích thước 3 và xác suất lai ghép là $0,5$. Xác suất đột biến ứng với đột biến xoá cạnh, đột biến dịch chuyển tâm, đột biến thay thế cạnh tham lam, đột biến tối ưu hoá cây con là $0,7$, $0,2$, $0,8$ và $0,5$.

Mỗi thực nghiệm chạy 30 lần cho mỗi bộ dữ liệu. Các thực nghiệm được thực hiện trên máy Pentium 4 Centrino 3.06 GHz CPU, 512MB RAM.

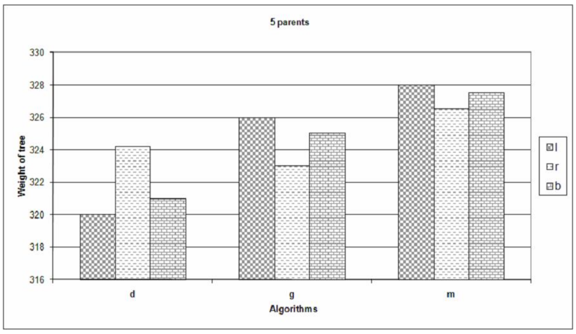
4.4. Các kết quả thực nghiệm

Các kết quả thực nghiệm cho thấy:

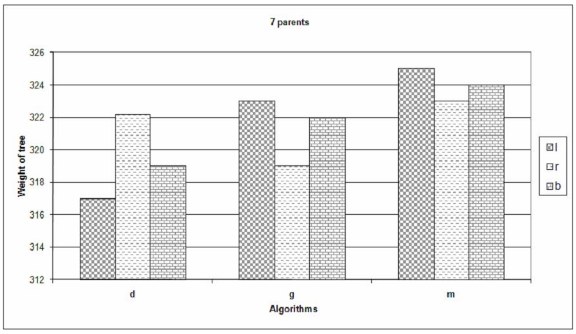
Đối với toán tử lai ghép 2 cha mẹ



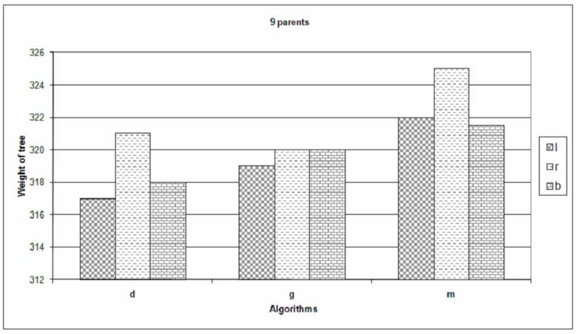
Hình 1. So sánh tổng các giá trị tốt nhất tìm được bằng các thuật toán *EA-xy*2 trên tất cả các bộ dữ liệu



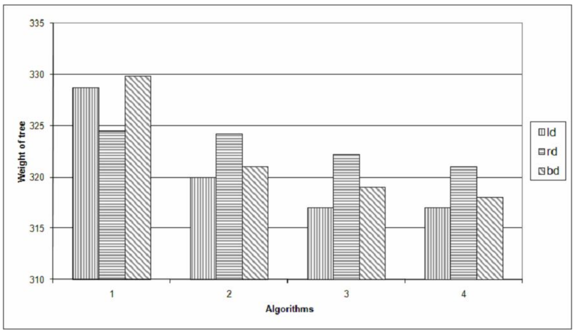
Hình 2. So sánh tổng các giá trị tốt nhất tìm được bằng các thuật toán *EA-xy5* trên tất cả các bộ dữ liệu



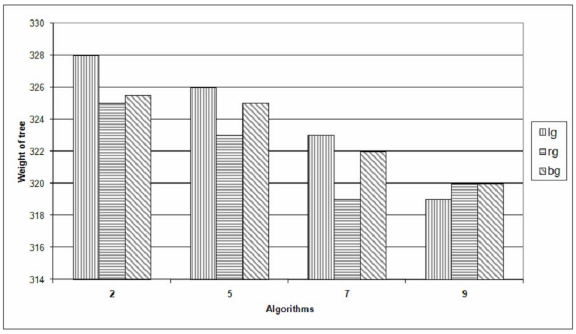
Hình 3. So sánh tổng các giá trị tốt nhất tìm được bằng các thuật toán *EA-xy7* trên tất cả các bộ dữ liệu



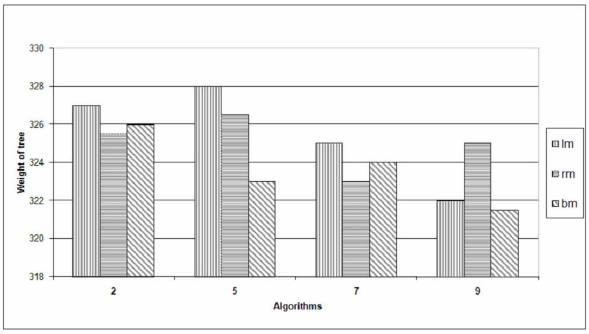
Hình 4. So sánh tổng các giá trị tốt nhất tìm được bằng các thuật toán *EA-xy9* trên tất cả các bộ dữ liệu



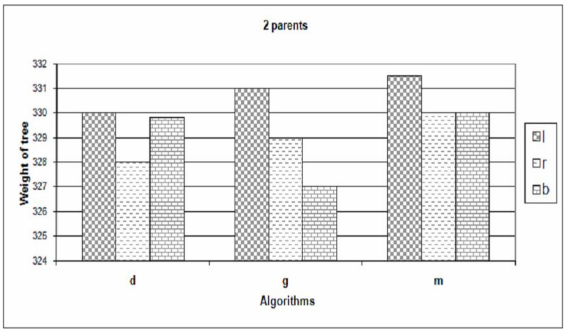
Hình 5. So sánh tổng các giá trị tốt nhất tìm được bằng các thuật toán *EA-xdk* trên tất cả các bộ dữ liệu ($x = \{b, r, l\}$, $k = \{2, 5, 7, 9\}$)



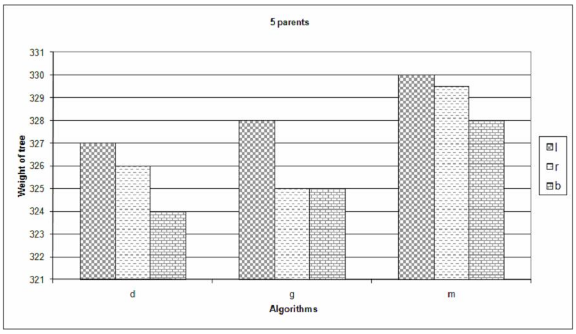
Hình 6. So sánh tổng các giá trị tốt nhất tìm được bằng các thuật toán EA-xgk trên tất cả các bộ dữ liệu ($x = \{b, r, l\}$, $k = \{2, 5, 7, 9\}$)



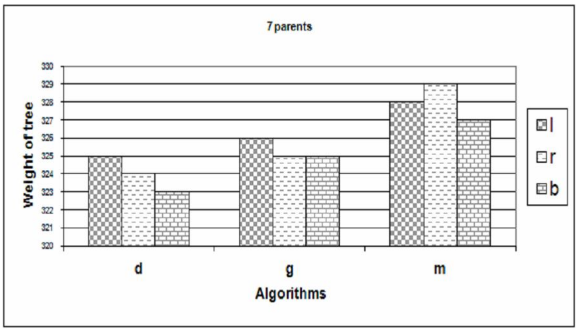
Hình 7. So sánh tổng các giá trị tốt nhất tìm được bằng các thuật toán *EA-xmk* trên tất cả các bộ dữ liệu ($x = \{b, r, l\}$, $k = \{2, 5, 7, 9\}$)



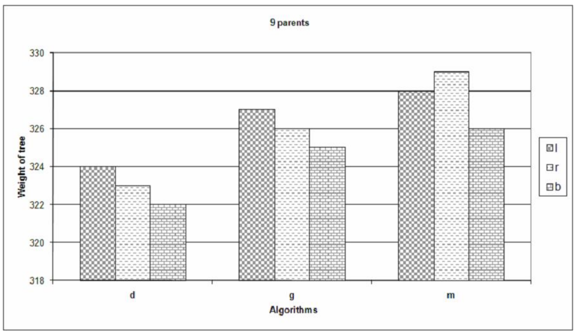
Hình 8. So sánh tổng các giá trị tốt nhất tìm được bằng các thuật toán *EA-xy*2 trên tất cả các bộ dữ liệu



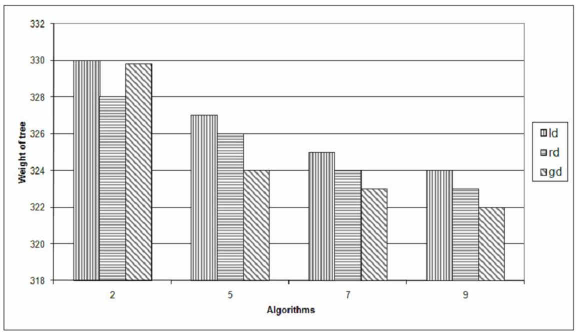
Hình 9. So sánh tổng các giá trị tốt nhất tìm được bằng các thuật toán *EA-xy*5 trên tất cả các bộ dữ liệu



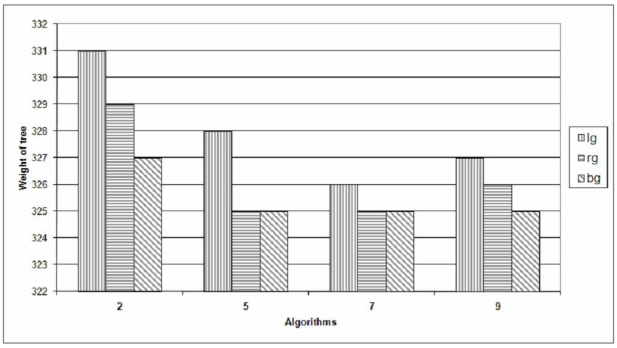
Hình 10. So sánh tổng các giá trị tốt nhất tìm được bằng các thuật toán *EA-xy*7 trên tất cả các bộ dữ liệu



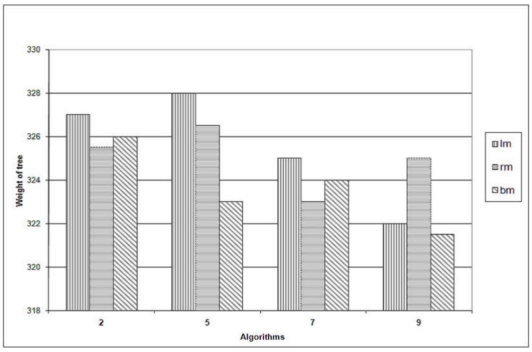
Hình 11. So sánh tổng các giá trị tốt nhất tìm được bằng các thuật toán *EA-xy*9 trên tất cả các bộ dữ liệu



Hình 12. So sánh tổng các giá trị tốt nhất tìm được bằng các thuật toán EA-xdk trên tất cả các bộ dữ liệu ($x = \{b, r, l\}$, $k = \{2, 5, 7, 9\}$)



Hình 13. So sánh tổng các giá trị tốt nhất tìm được bằng các thuật toán *EA-xgk* trên tất cả các bộ dữ liệu ($x = \{b, r, l\}$, $k = \{2, 5, 7, 9\}$)



Hình 14. So sánh tổng các giá trị tốt nhất tìm được bằng các thuật toán *EA-xmk* trên tất cả các bộ dữ liệu ($x = \{b, r, l\}$, $k = \{2, 5, 7, 9\}$)

- Hình 1 cho thấy kết quả tốt nhất tìm được từ giải thuật di truyền sử dụng
  - Toán tử lai ghép k_1: phương pháp chọn cạnh d là tồi nhất, phương pháp chọn cạnh g là tốt nhất.
  - Toán tử lai ghép k_2: phương pháp chọn cạnh d là tồi nhất, phương pháp chọn cạnh m là tốt nhất.
  - Toán tử lai ghép k_3: phương pháp chọn cạnh d là tồi nhất, phương pháp chọn cạnh m là tốt nhất.
  - Toán tử lai ghép k_2 và phương pháp chọn cạnh d là tốt nhất; toán tử lai ghép k_1 và phương pháp chọn cạnh d là tồi nhất.

- Hình 8 cho thấy kết quả trung bình tìm được từ giải thuật di truyền sử dụng
  - Toán tử lai ghép k_1, phương pháp chọn cạnh m là tồi nhất, phương pháp chọn cạnh g là tốt nhất.
  - Toán tử lai ghép k_2, phương pháp chọn cạnh m là tồi nhất, phương pháp chọn cạnh g là tốt nhất.
  - Toán tử lai ghép k_3, phương pháp chọn cạnh m là tồi nhất, phương pháp chọn cạnh d là tốt nhất.
  - Toán tử lai ghép k_1 và phương pháp chọn cạnh g là tốt nhất; toán tử lai ghép k_3 và phương pháp chọn cạnh m là tồi nhất.

Đối với toán tử lai ghép 5 cha mẹ

- Hình 2 cho thấy kết quả tốt nhất tìm được từ giải thuật di truyền sử dụng

- Toán tử lai ghép k_1, phương pháp chọn cạnh m là tồi nhất, phương pháp chọn cạnh d là tốt nhất.
- Toán tử lai ghép k_2, phương pháp chọn cạnh m là tồi nhất; phương pháp chọn cạnh g là tốt nhất.
- Toán tử lai ghép k_3, phương pháp chọn cạnh m là tồi nhất, phương pháp chọn cạnh d là tốt nhất.
- Toán tử lai ghép k_3 và phương pháp chọn cạnh d là tốt nhất; toán tử lai ghép k_3 và phương pháp chọn cạnh m là tồi nhất.

• Hình 9 cho thấy kết quả trung bình tìm được từ giải thuật di truyền sử dụng

- Toán tử lai ghép k_1, phương pháp chọn cạnh m là tồi nhất; phương pháp chọn cạnh d là tốt nhất.
- Toán tử lai ghép k_2, phương pháp chọn cạnh m là tồi nhất; phương pháp chọn cạnh g là tốt nhất.
- Toán tử lai ghép k_3, phương pháp chọn cạnh m là tồi nhất; phương pháp chọn cạnh d là tốt nhất.
- Toán tử lai ghép k_1 và phương pháp chọn cạnh d là tốt nhất; toán tử lai ghép k_3 và phương pháp chọn cạnh m là tồi nhất.

Đối với toán tử lai ghép 7 cha mẹ

• Hình 3 cho thấy kết quả tốt nhất tìm được từ giải thuật di truyền sử dụng

- Toán tử lai ghép k_1, phương pháp chọn cạnh m là tồi nhất; phương pháp chọn cạnh d là tốt nhất.
- Toán tử lai ghép k_2, phương pháp chọn cạnh m là tồi nhất; phương pháp chọn cạnh g là tốt nhất.
- Toán tử lai ghép k_3, phương pháp chọn cạnh m là tồi nhất; phương pháp chọn cạnh d là tốt nhất.
- Toán tử lai ghép k_3 và phương pháp chọn cạnh d là tốt nhất; toán tử lai ghép k_3 và phương pháp chọn cạnh m là tồi nhất.

• Hình 10 cho thấy kết quả trung bình tìm được từ giải thuật di truyền sử dụng

- Toán tử lai ghép k_1, phương pháp chọn cạnh m là tồi nhất; phương pháp chọn cạnh d là tốt nhất.
- Toán tử lai ghép k_2, phương pháp chọn cạnh m là tồi nhất; phương pháp chọn cạnh d là tốt nhất.

– Toán tử lai ghép k_3, bởi phương pháp chọn cạnh m là tồi nhất; phương pháp chọn cạnh d là tốt nhất.
– Toán tử lai ghép k_1 và phương pháp chọn cạnh d là tốt nhất; toán tử lai ghép k_3 và phương pháp chọn cạnh m là tồi nhất.

Đối với toán tử lai ghép 9 cha mẹ

- Hình 4 cho thấy kết quả tốt nhất tìm được từ giải thuật di truyền sử dụng
  – Toán tử lai ghép k_1, phương pháp chọn cạnh m là tồi nhất; phương pháp chọn cạnh d là tốt nhất.
  – Toán tử lai ghép k_2, phương pháp chọn cạnh m là tồi nhất; phương pháp chọn cạnh g là tốt nhất.
  – Toán tử lai ghép k_3, phương pháp chọn cạnh m là tồi nhất; phương pháp chọn cạnh d là tốt nhất.
  – Toán tử lai ghép k_3 và phương pháp chọn cạnh d là tốt nhất; toán tử lai ghép k_2 và phương pháp chọn cạnh m là tồi nhất.

- Hình 11 cho thấy kết quả trung bình tìm được từ giải thuật di truyền sử dụng
  – Toán tử lai ghép k_1, phương pháp chọn cạnh m là tồi nhất; phương pháp chọn cạnh d là tốt nhất.
  – Toán tử lai ghép k_2, phương pháp chọn cạnh m là tồi nhất; phương pháp chọn cạnh d là tốt nhất.
  – Toán tử lai ghép k_3, phương pháp chọn cạnh m là tồi nhất; phương pháp chọn cạnh d là tốt nhất.
  – Toán tử lai ghép k_1 và phương pháp chọn cạnh d là tốt nhất; toán tử lai ghép k_2 và phương pháp chọn cạnh m là tồi nhất.

So sánh kết quả tìm được giữa số cha mẹ chọn để lai ghép khác nhau và phương pháp chọn cha mẹ khác nhau trong cách chọn cạnh d để kết nạp cạnh vào cây con.

- Hình 5 cho thấy, kết quả tốt nhất tìm được khi sử dụng toán tử lai ghép k_3 và số cha mẹ là 7 hoặc 9 là tốt nhất. Kết quả tốt nhất tìm được khi sử dụng toán tử lai ghép k_1 và số cha mẹ là 2 là tồi nhất.
- Hình 12 cho thấy, kết quả trung bình tìm được khi sử dụng toán tử lai ghép k_3 và số cha mẹ là 7 hoặc 9 là tốt nhất. Kết quả tốt nhất tìm được khi sử dụng toán tử lai ghép k_1 và số cha mẹ là 2 là tồi nhất.

So sánh kết quả tìm được giữa số cha mẹ chọn để lai ghép khác nhau và phương pháp chọn cha mẹ khác nhau trong cách chọn cạnh g để kết nạp cạnh vào cây con.

- Hình 6 cho thấy, kết quả tốt nhất tìm được khi sử dụng toán tử lai ghép k_2 và số cha mẹ là 7 là tốt nhất. Kết quả tốt nhất tìm được khi sử dụng toán tử lai ghép k_3 và số cha mẹ là 2 là tồi nhất.

- Hình 13 cho thấy, kết quả trung bình tìm được khi sử dụng toán tử lai ghép k_1 hoặc toán tử lai ghép k_2 và số cha mẹ là 5 hoặc 7 là tốt nhất. Kết quả tốt nhất tìm được khi sử dụng toán tử lai ghép k_3 và số cha mẹ là 2 là tồi nhất.

So sánh kết quả tìm được giữa số cha mẹ chọn để lai ghép khác nhau và phương pháp chọn cha mẹ khác nhau trong cách chọn cạnh m để kết nạp cạnh vào cây con.

- Hình 7 cho thấy, kết quả tốt nhất tìm được khi sử dụng toán tử lai ghép k_2 và số cha mẹ là 7 là tốt nhất. Kết quả tốt nhất tìm được khi sử dụng toán tử lai ghép k_1 và số cha mẹ là 5 là tồi nhất.

- Hình 14 cho thấy, kết quả trung bình tìm được khi sử dụng toán tử lai ghép k_1 và số cha mẹ là 9 là tốt nhất. Kết quả tốt nhất tìm được khi sử dụng toán tử lai ghép k_3 và số cha mẹ là 2 là tồi nhất.

So sánh kết quả tìm được giữa số cha mẹ chọn để lai ghép khác nhau và phương pháp khác nhau để chọn cạnh kết nạp vào cây con trong toán tử lai ghép k_2.

- Hình 1, 2, 3, 4 cho thấy, kết quả tốt nhất tìm được khi sử dụng số cha mẹ lai ghép là 7 và phương pháp chọn cạnh g là tốt nhất. Kết quả tốt nhất tìm được khi sử dụng số cha mẹ lai ghép là 2 và phương pháp chọn cạnh m là tồi nhất.

- Hình 8, 9, 10, 11 cho thấy, kết quả trung bình tìm được khi sử dụng số cha mẹ lai ghép là 9 và phương pháp chọn cạnh d là tốt nhất. Kết quả trung bình tìm được khi sử dụng số cha mẹ lai ghép là 2 và phương pháp chọn cạnh m là tồi nhất.

So sánh kết quả tìm được giữa số cha mẹ chọn để lai ghép khác nhau và phương pháp khác nhau để chọn cạnh kết nạp vào cây con trong toán tử lai ghép k_3.

- Hình 1, 2, 3, 4 cho thấy, kết quả tốt nhất tìm được khi sử dụng số cha mẹ lai ghép là 7 hoặc 9 và phương pháp chọn cạnh d là tốt nhất. Kết quả tốt nhất tìm được khi sử dụng số cha mẹ lai ghép là 2 và phương pháp chọn cạnh d là tồi nhất.

- Hình 8, 9, 10, 11 cho thấy, kết quả trung bình tìm được khi sử dụng số cha mẹ lai ghép là 9 và phương pháp chọn cạnh d là tốt nhất. Kết quả trung bình tìm được khi sử dụng số cha mẹ lai ghép là 2 và phương pháp chọn cạnh m là tồi nhất.

KẾT LUẬN

Bài báo vừa trình bày ba phương pháp chọn cha mẹ và chọn cạnh để kết nạp vào con trong toán tử lai ghép của giải thuật di truyền giải bài toán $BDMST$. Các phương pháp đề xuất chọn cha mẹ đã cho thấy tính hiệu quả trên các bộ dữ liệu http://tomandtun.googlepages.com/phd.

Trong các nghiên cứu tiếp theo, tác giả sẽ thử nghiệm với các tham số khác nhau trong toán tử lai ghép đa cha mẹ để giải bài toán $BDMST$ và cũng sẽ áp dụng toán tử lai ghép đa cha mẹ để giải bài toán NP-khó khác trên đồ thị và hy vọng các phương pháp đề xuất không chỉ hiệu quả đối với bài toán này mà còn cả với nhiều bài toán NP-khó khác.

TÀI LIỆU THAM KHẢO

[1] A. Abdalla, N. Deo, Random-tree diameter and the diameter constrained MST, *International Journal of Computer Mathematics* **79** (6) (2002) 651–663.

[2] A. Abdalla, "Computing a Diameter-constrained Minimum Spanning Tree", PhD Dissertation, The School of Electrical Engineering and Computer Science, University of Central Florida, 2001.

[3] N.R.Achuthan, L.Caccetta, P.Caccetta, and A. Geelen, Computational methods for the diameter restricted minimum weight spanning tree problem, *Australian Journal of Combinatorics* **10** (1994) 51–71.

[4] A. Bookstein and S. T. Klein, Compression of correlated bit-vectors, *Information Systems* **16** (4) (1996) 387–400.

[5] K. Bala, K. Petropoulos, and T. E. Stern, Multicasting in a linear lightwave network, *Proceedings of IEEE INFOCOM93* **3** (1993) (350–1358).

[6] M.R. Garey and D.S.Johnson, *Computers and Intractability: A Guide to the Theory of NP-Completeness*, W.H.Freeman, New York, 1979 (p.206).

[7] J.Gottlieb, B.A.Julstrom, F.Rothlauf, and G.R.Raidl, Prufer numbers: A poor representation of spanning trees for evolutionary search, *Proceedings of the Genetic and Evolutionary Computation Conference (GECCO2001)*, San Francisco, USA, 2001 (343–350).

[8] L Gouveia, T.L. Magnanti and C. Requejo, A 2-path approach for odd diameter constrained minimum spanning and steiner trees, *Network* **44** (4) (2004) 254–265.

[9] M. Gruber and G.R. Raidl, A new 0-1 ILP approach for the bounded diameter minimum spanning tree problem, *Proceedings of the 2$^{\text{nd}}$ International Network Optimization Conference*, Portugal, 2005 (178–185).

[10] M. Gruber and G.R. Raidl, Variable neighbourhood search for the bounded diameter minimum spanning tree problem, *Proceedings of the 18$^{\text{th}}$ Mini Euro Conference on Variable Neighborhood Search*, Spain, 2005.

[11] M. Gruber, J. Hemert, and G.R. Raidl, Neighbourhood searches for the bounded diameter minimum spanning tree problem embedded in a VNS, EA and ACO, *Proceedings of Genetic and Evolutionary Computational Conference (GECCO2006)*, Seattle, WA, USA, 2006 (1187–1194).

[12] Huynh Thi Thanh Binh, Nguyen Xuan Hoai, R.I McKay, A new hybrid genetic algorithm for solving the bounded diameter minimum spanning tree problem, *Proceedings of IEEE World Congress on Computational Intelligence (CEC2008)*, Hong Kong, 2008 (3127–3133).

[13] Huynh Thi Thanh Binh, Nguyen Duc Nghia, New multi-parent recombination in genetic algorithm for solving bounded diameter minimum spanning tree problem, *Proceedings of 1st Asian Conference on Intelligent Information and Database Systems, ACIIDS 2009*, Hue, Viet Nam (283–288).

[14] Huynh Thi Thanh Binh, Nguyen Xuan Hoai, R.I McKay, Nguyen Duc Nghia, New heuristic and hybrid genetic algorithm for solving the bounded diameter minimum spanning tree problem, *Proceeding of Genetic and Evolutionary Computation Conference (GECCO2009)*, Montreal, Canada, 2009 (373–380).

[15] B.A. Julstrom, G.R. Raild, A permutation coded evolutionary for the bounded diameter minimum spanning tree problem, *Proceedings of the Genetic and Evolutionary Computation Conference (GECCO2003)*, Chicago, IL, USA, 2003 (2–7).

[16] B.A. Julstrom, Encoding bounded diameter minimum spanning trees with permutations and random keys, *Lecture Notes in Computer Science* **3102**, Springer, Berlin Heidelberg New York, (1272–1281).

[17] B.A Julstrom, Greedy heuristics for the bounded diameter minimum spanning tree problem, *ACM Journal of Experimental Algorithmics* **14** (2009).

[18] G. Kortsarz and D. Peleg, Approximating shallow-light trees, *Proceedings of the 8th Symposium on Discrete Algorithms*, New Orleans, Louisianna, USA, 1997 (103–110).

[19] G. Kortsarz and D. Peleg, Approximating the weight of shallow steiner trees, *Discrete Application Mathematics* **93** (1999) 265–285.

[20] Nguyen Duc Nghia and Huynh Thi Thanh Binh, A new recombination operator for solving bounded diameter minimum spanning tree problem, *Proceedings of RIVF2007*, Ha Noi, Viet Nam, 2007 (108–113).

[21] Nguyen Duc Nghia, Huynh Thi Thanh Binh, Heuristic algorithms for solving bounded diameter minimum spanning tree problem and its application to genetic algorithm development, *Advances in Greedy Algorithms Book, I-Tech Education and Publishing*, Austria, 2008 (370–386).

[22] C.C. Palmer and A. Kershenbaum, Representing trees in genetic algorithms, *Proceedings of The First IEEE Conference on Evolutionary Computation*, Orlando, Florida, USA, 1994 (379–384).

[23] R. Prim, Shortest connection networks and some generalization, *Bell System Technical Journal* **36** (1957) 1389–1401.

[24] G.R. Raidl and B.A. Julstrom, Edge-sets: an effective evolutionary coding of spanning trees, *IEEE Transactions on Evolutionary Computation* **7** (2003) 225–239.

[25] G.R. Raidl and B.A. Julstrom, Greedy heuristics and an evolutionary algorithm for the bounded-diameter minimum spanning tree problem, *Proceeding of the ACM Symposium on Applied Computing*, Chicago, USA, 2003 (747–752).

[26] K. Raymond, A tree-based algorithm for distributed mutual exclusion, *ACM Transactions on Computer Systems* **7** (1) (1989) 61–77.

[27] F. Rothlauf, *Representations for Genetic and Evolutionary Algorithms*, 2nd Edition, Springer-Verlag, 2006.

[28] A. Singh and A.K. Gupta, Impoved heuristics for the bounded diameter minimum spanning tree problem, *Journal Soft Computing* **11** (2007) 911–921.

Nhận bài ngày 5 - 6 - 2009
Nhận lại sau sửa ngày 14 - 1 -2010