Kỷ yếu Hội nghị Quốc gia lần thứ VIII về Nghiên cứu cơ bản và ứng dụng Công nghệ thông tin (FAIR); Hà Nội, ngày 9-10/7/2015
DOI: 10.15625/vap.2015.000183

MỘT THUẬT TOÁN TỐI ƯU ĐÀN KIẾN DÓNG HÀNG TOÀN CỤC MẠNG TƯƠNG TÁC PROTEIN

Trần Ngọc Hà[1], Hoàng Xuân Huấn[2]

[1]Trường ĐH Sư phạm – Đại học Thái Nguyên.
[2]Trường ĐH Công nghệ - Đại học Quốc gia Hà Nội.
hatn@tnu.edu.vn, huanhx@vnu.edu.vn

TÓM TẮT - *Dóng hàng toàn cục mạng tương tác protein là một trong những bài toán quan trọng trong tin sinh học và đang được nhiều người quan tâm nghiên cứu. Các mạng được dóng hàng chính xác cho phép ta có thể xác định các orthologous protein. Bài viết này, chúng tôi giới thiệu một thuật toán dóng hàng toàn cục mạng tương tác protein dựa trên phương pháp tối ưu hoá đàn kiến. Các thực nghiệm cho thấy phương pháp đề xuất cho kết quả tốt hơn hẳn các phương pháp mới nhất hiện nay.*

Từ khóa - *Dóng hàng toàn cục, mạng tương tác protein, tối ưu đàn kiến.*

I. GIỚI THIỆU

Trước cách tiếp cận dóng hàng mạng, việc phát hiện nhóm các orthologous protein chỉ dựa trên các quan hệ tiến hóa, với tiêu chí thường được sử dụng là độ tương tự về mặt trình tự [1, 23]. Tuy nhiên, chỉ tính tương đồng trình tự thường không đủ để xác định các phức hợp protein được bảo tồn [12, 24, 26]. Sự phát triển của các kỹ thuật công nghệ sinh học trong hơn thập kỷ qua đã cho phép xây dựng được các mạng tương tác protein Protein-Protein Interraction Network – PPI Network) cho nhiều loài sinh vật. Từ các dữ liệu này, một số bài toán về phân tích mạng PPI đã được đặt ra (xem [3, 7, 15-17]), chẳng hạn như: phân tích cấu trúc tô pô mạng [9], phát hiện mô-đun [2]... Trong đó, đặc biệt quan trọng là các bài toán dóng hàng mạng PPI dựa trên kết hợp thông tin về sự tương tác giữa các protein cùng với mối quan hệ tiến hóa giữa các trình tự. Việc so sánh tính tương đồng của các mạng PPI này cung cấp nhiều thông tin hữu ích cho dự đoán các chức năng chưa biết hoặc kiểm định các chức năng đã biết của các proteins [8, 11, 25].

Các phương pháp dóng hàng mạng tương tác Protein được chia thành 2 hướng tiếp cận: dóng hàng cục bộ và dóng hàng toàn cục. Mục tiêu của dóng hàng cục bộ là xác định các mạng con gần nhau về cấu trúc mạng và/hoặc tương tự nhau về trình tự) (xem [13, 14, 21, 24]). Với mục tiêu đó, kết quả của dóng hàng cục bộ thường chứa nhiều mạng con chồng lấn nhau, vì vậy có thể dẫn tới sự nhập nhằng khi dóng hàng một protein với nhiều protein khác. Mục tiêu của phương pháp dóng hàng toàn cục giữa 2 mạng protein là tránh các nhập nhằng thường gặp ở phương pháp dóng hàng cục bộ. Bài toán này được Aladag và Erten chứng minh là bài toán NP khó [1].

Thuật toán dóng hàng toàn cục đáng chú ý đầu tiên là IsoRank [25] được Sing et al. (2008) đề xuất, phát triển dựa trên dóng hàng cục bộ. Sau IsoRank, một số thuật toán tương tự đã được đề xuất như PATH và GA [25], PISwap [4, 5] nhờ đưa thêm các nới lỏng thích hợp của hàm đánh giá trên tập các ma trận ngẫu nhiên hoặc ứng dụng tìm kiếm cục bộ trên dóng hàng thu được từ lời giải một thuật toán khác. MI-GRAAL [15,16] và các biến thể [19,20] dựa trên kết hợp kỹ thuật tham ăn với thông tin heuristics như: graphlet, hệ số phân nhóm, độ lập dị (eccentricities) và độ tương tự (giá trị E-values từ chương trình BLAST). Các thuật toán này đều đưa ra kết quả nhanh và tốt hơn so với các thuật toán trước đó. Tuy nhiên, những thuật toán đã nêu chỉ tối ưu cho độ chính xác (hàm mục tiêu) hoặc tính khả mở. Vì các mạng PPI có thường số đỉnh lớn nên cả tính chính xác và tính khả mở (thời gian chạy) cần được quan tâm. Aladag và Erten (2013) đề xuất thuật toánSPINAL [1] heuristic có thời gian đa thức cho kết quả tương đối tốt. Thuật toán này gồm hai pha: pha đầu tính điểm tương đồng cho tất cả cặp protein; pha sau xây dựng đơn ánh bằng cách cải tiến một cách cục bộ từng tập con của lời giải hiện có. Các thực nghiệm được chạy trên các bộ dữ liệu tiêu chuẩn là Saccharomyces cerevisiae, Drosophila melanogaster, Caenorhabditiselegans và Homo sapiens cho thấy SPINAL cho chất lượng lời giải tốt hơn 2 thuật toán tốt nhất khi đó là IsoRank và MI-GRAAL.

Gần đây, Đỗ Đức Đông và các cộng sự (2014) cũng đã giới thiệu một thuật toán ngẫu nhiên FastAn [6] gồm 2 giai đoạn; giai đoạn đầu là thủ tục xây dựng dóng hàng thô theo cách tiếp cận heuristic. Sau đó tiến hành thủ tục rebuild nhờ giữ lại một số cặp đỉnh tốt nhất đã được dóng hàng trong lời giải xây dựng được ở giai đoạn 1 và dóng hàng lại cho các cặp đỉnh còn lại để tăng chất lượng lời giải. Các thực nghiệm cho thấy FastAn có kết quả tốt hơn cả về thời gian chạy và chất lượng lời giải so với SPINAL.

Bài báo này đề xuất một phương pháp dóng hàng toàn cục mạng tương tác protein dựa trên tối ưu đàn kiến gọi là ACOGA. Thuật toán được thực hiện theo nhiều vòng lặp, trong mỗi vòng lặp, các kiến đi xây dựng lời giải, sau đó lời giải của kiến tốt nhất sẽ được lựa chọn để cập nhật vết mùi và sử dụng tìm kiếm cục bộ để tăng chất lượng lời giải. Kết quả thực nghiệm cho thấy thuật toán đề xuất cho chất lượng lời giải tốt hơn nhiều so với FastAn.

Ngoài kết luận, phần còn lại của bài báo có cấu trúc như sau: Phần 2 giới thiệu các khái niệm liên quan đến bài toán dóng hàng toàn cục mạng tương tác Protein. Thuật toán mới ACOGA được giới thiệu ở phần 3. Phần 4 trình bày các thực nghiệm so sánh hiệu quả của thuật toán đề xuất với các thuật toán FastAn.

II. BÀI TOÁN DÓNG HÀNG TOÀN CỤC MẠNG TƯƠNG TÁC PROTEIN

Giả sử $G_1 = (V_1, E_1)$ và $G_2 = (V_2, E_2)$ là 2 đồ thị mô tả 2 mạng tương tác protein, trong đó V_1, V_2 tương ứng là tập các đỉnh mô tả các protein trong các mạng G_1 và G_2 tương ứng;; E_1, E_2 là tập các cạnh mô tả các tương tác giữa các protein tương ứng trong G_1, G_2. Không mất tính tổng quát ta có thể giả thiết $|V_1| \leq |V_2|$, trong đó $|V|$ ký hiệu số phần tử của tập V.

Dóng hàng mạng toàn cục là tìm một đơn ánh từ tập V_1 vào tập V_2 tốt nhất theo một tiêu chuẩn đánh giá nào đó. Ở mỗi nghiên cứu, người ta đề xuất các tiêu chuẩn đánh giá khác nhau. Dưới đây là định nghĩa được sử dụng chủ yếu trong các nghiên cứu trước đây [1, 4, 5, 16, 25].

Định nghĩa 1. (Dóng hàng mạng) Đồ thị $A_{12} = (V_{12}, E_{12})$ được coi là dóng hàng mạng của 2 đồ thị $G_1 = (V_1, E_1)$ và $G_2 = (V_2, E_2)$ nếu nó thoả mãn các điều kiện sau:

i) Mỗi nút $< u_i, v_j > \in V_{12}$ tương ứng với một cặp $u_i \in V_1$ và $v_j \in V_2$.

ii) Hai nút phân biệt $< u_i, v_j >$ và $< u_i', v_j' >$ của V_{12} phải thoả mãn $u_i \neq u_i'$ và $v_j \neq v_j'$

iii) Cạnh $(< u_i, v_j >, < u_i', v_j' >)$ thuộc tập E_{12} khi và chỉ khi $(u_i, u_i') \in E_1$ và $(v_j, v_j') \in E_2$.

Định nghĩa 2. (Dóng hàng tối ưu toàn cục mạng tương tác protein) Một dóng hàng $A_{12} = (V_{12}, E_{12})$ là lời giải của bài toán dóng hàng toàn cục 2 mạng tương tác protein G_1, G_2 nếu nó làm cực đại tiêu chí GNAS cho bởi công thức (1):

$$GNAS(A_{12}) = \alpha |E_{12}| + (1-\alpha) \sum_{\forall < u_i, v_j >} similar(u_i, v_j) \quad (1)$$

trong đó $\alpha \in [0,1]$ là tham số thể hiện mối tương quan giữa sự tương đồng về cấu trúc mạng và độ tương đồng về trình tự. Giá trị $Similar(u_i, v_j)$ được tính xấp xỉ nhờ sử dụng BLAST bit-scores hay E-values.

Theo nghiên cứu của Aladag và Erten [1] bài toán tìm tối ưu toàn cục của dóng hàng mạng đã được chứng minh là NP khó.

III.THUẬT TOÁN ĐỀ XUẤT

A. Lược đồ chung

Cho các đồ thị G_1, G_2; tham số α và các độ tương tự của các cặp đỉnh $< u_i, v_j >$ trong đó $u_i \in V_1, v_j \in V_2$. Với mỗi tập con các cặp đỉnh V_{12} của tập $V_1 \times V_2$, ta ký hiệu $V_{12}^1 = \{u_i \in V_1 : < u_i, v_j > \in V_{12}\}, V_{12}^2 = \{v_j \in V_2 : < u_i, v_j > \in V_{12}\}$ (V_{12}^i là tập các cặp các đỉnh thuộc tập đỉnh V_i của đồ thị G_i đã được dóng hàng). Thuật toán ACOGA được xây dựng như dưới đây:

Bước 1. Khởi tạo ma trận vết mùi, và tập A gồm m kiến.

Bước 2. Thực hiện lặp trong khi chưa thoả mãn điều kiện dừng

Với mỗi kiến ta tiến hành các bước sau:

2.1. Khởi tạo tập $V_{12} = \{< u_i, v_j >\}$ là cặp đỉnh có độ tương đồng lớn nhất.

2.2 Thực hiện lặp với *k= 2* tới $|V_1|$

2.2.1. Tìm đỉnh $u_i \in V_1 - V_{12}^1$ có số cạnh tới các đỉnh trong V_{12}^1 lớn nhất;

*2.2.2.*Tìm đỉnh $v_j \in V_2 - V_{12}^2$ theo thủ tục bước ngẫu nhiên được đặc tả ở mục B theo công thức (5)

2.2.3. Bổ sung $< u_i, v_j >$ vào V_{12};

2.2.4. Cập nhật lại E_{12} dựa trên V_{12};

2.3. Thực hiện tìm kiếm cục bộ trên lời giải tốt nhất do các kiến tìm được để cải thiện chất lượng lời giải.

2.4. Cập nhật lại lời giải tốt nhất.

2.5. Cập nhật vết mùi theo quy tắc SMMAS dựa trên lời giải tốt nhất.

Bước 3. Lưu lại lời giải tốt nhất.

Thuật toán này được đặc tả trong hình 1.

Thuật toán 1: Thuật toán ACOGA
Input: Đồ thị 1: $G_1 = (V_1, E_1)$; Đồ thị 2: $G_2 = (V_2, E_2)$;
Độ tương đồng giữa các cặp đỉnh: $Similar$
Tham số cân bằngα
Output: Dóng hàng mạng $A_{12} = (V_{12}, E_{12})$
Begin
Khởi tạo; // Khởi tạo ma trận vết mùi và m kiến (A);
While (Chưa thỏa mãn điều kiện dừng) do
for each kiến a$\in$ A **do**
$V_{12} = \{< u_x, v_y >\}$ // $< u_x, v_y >$ là cặp đỉnh có độ tương tự lớn nhất.
for k=2 **to** $|V_1|$ **do**
$u_i = findMaxRelate(u_i \in V_1 - V_{12}^1)$
$v_j = antMove(u_i, V_2 - V_{12}^2)$
$V_{12} = V_{12} \cup < u_i, v_j >$
Update(E_{12})
end for
end for
Tìm kiếm cục bộ;
Cập nhật lời giải tốt nhất;
Cập nhật vết mùi theo quy tắc *SMMAS*;
end while
Ghi lại lời giải tốt nhất;
End.

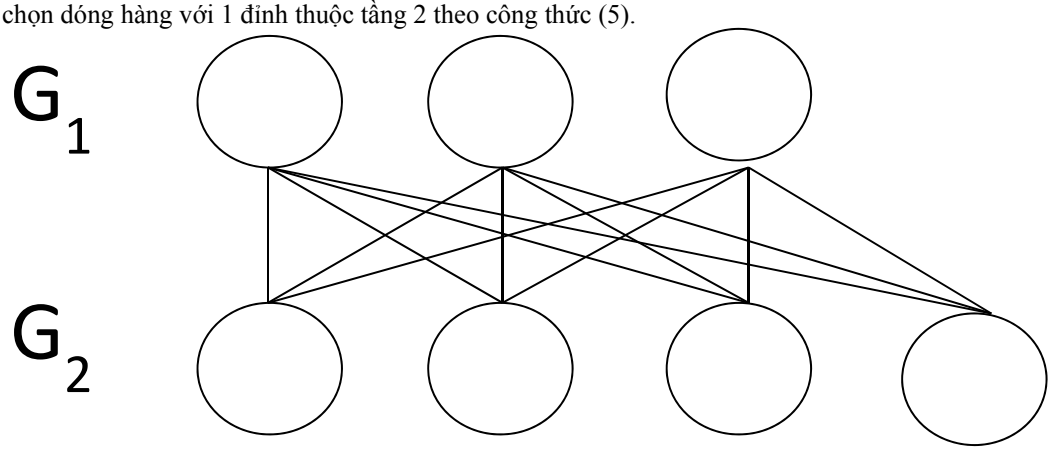
Hình 1. Đặc tả thuật toán ACOGA.

Chú ý rằng ở bước 2.2.1, việc tìm $u_i \in V_1 - V_{12}^1$ có số cạnh tới các đỉnh trong V_{12}^1 lớn nhất nhằm tăng số lượng các cạnh có thể được bảo toàn sau khi dóng hàng, nếu tìm được nhiều đỉnh tốt nhất thì sẽ lựa chọn ngẫu nhiên một đỉnh tìm được để dóng hàng.

B. Các thành phần của ACOGA

Đồ thị cấu trúc

Đồ thị cấu trúc của thuật toán ACOGA được biểu thị trong hình 2, gồm 2 tầng, tầng thứ *i* thể hiện đồ thị G_i. Các đỉnh ở tầng trên được kết nối với tất cả các đỉnh ở tầng dưới. Khi xây dựng lời giải, kiến sẽ xuất phát từ một đỉnh thuộc tầng 1 và lựa chọn dóng hàng với 1 đỉnh thuộc tầng 2 theo công thức (5).



Hình 2. Đồ thị cấu trúc của thuật toán ACOGA

Một dóng hàng toàn cục của 2 đồ thị theo định nghĩa 1 là một đường đi xuất phát từ 1 đỉnh của G_1 dóng với 1 đỉnh của G_2 sau đó quay lại G_1 rồi tiếp tục dóng với 1 đỉnh của G_2, lặp lại cho tới khi tất cả các đỉnh của G_1 đã được dóng hàng.

Vết mùi và thông tin heuristic

Vết mùi τ_j^i trên cạnh $< i, j >$ dóng đỉnh $u_i \in G_1$ với đỉnh $v_j \in G_2$ được khởi tạo bằng τ_{max} và sau đó được cập nhật lại sau mỗi vòng lặp theo công thức (6).

Thông tin heuristic η_j^i được tính theo công thức 4.

$$\eta_j^i = \alpha * M + (1-\alpha) * similar(u_i, v_j) \quad (4)$$

trong đó α là hằng số thể hiện mối tương quan giữa độ tương đồng về cấu trúc và tính tương đồng về trình tự. M là tổng số cạnh được bảo toàn sau dóng hàng nếu đỉnh u_i được dóng với đỉnh v_j, $similar(u_i, v_j)$ là độ tương đồng giữa 2 đỉnh u_i và v_j

Thủ tục bước ngẫu nhiên để xây dựng dóng hàng

Tại mỗi vòng lặp, đầu tiên kiến chọn một đỉnh $u_i \in V_1$, sau đó chọn đỉnh $v_j \in V_2$ theo xác suất được cho bởi công thức (5)

$$p_j^i = \frac{(\tau_j^i)^a * [\eta_j^i]^b}{\sum_{k \in R\_V_2} (\tau_k^i)^a * [\eta_k^i]^b} \quad (5)$$

trong đó $R\_V_2 = V_2 - V_{12}^2$ là các đỉnh của đồ thị G_2 chưa được dóng hàng.

Sau khi lựa chọn được đỉnh $v_j \in V_2$ để dóng với $u_i \in V_1$, kiến quay lại lựa chọn đỉnh tiếp theo của đồ thị G_1 để tiếp tục dóng hàng. Quá trình lặp lại cho đến khi tất cả các đỉnh của G_1 được dóng hàng với các đỉnh của G_2

Quy tắc cập nhật vết mùi

Sau khi tất cả các kiến đã xây dựng lời giải, lời giải của kiến tốt nhất được áp dụng thủ tục tìm kiếm cục bộ để tăng chất lượng lời giải. Lời giải tốt nhất này được sử dụng để cập nhật vết mùi trên các cạnh theo quy tắc cập nhật mùi SMMAS[10], như dưới đây:

$$\tau_j^i = (1-\rho)\tau_j^i + \Delta_j^i \quad (6)$$

$$\Delta_j^i = \begin{cases} \rho * \tau_{max} & (i,j) \in \text{ best solution} \\ \rho * \tau_{min} & (i,j) \notin \text{ best solution} \end{cases} \quad (7)$$

trong đó τ_{max} và τ_{min} là các tham số được cho trước, $\rho \in (0,1)$ là tham số bay hơi cho trước quy định 2 thuộc tính, ρ nhỏ thể hiện việc tìm kiếm quanh thông tin học tăng cường, ρ lớn thể hiện tính khám phá.

Thủ tục tìm kiếm cục bộ

Thuật toán 2: Thủ tục tìm kiếm cục bộ
Input: Đồ thị 1: $G_1 = (V_1, E_1)$; Đồ thị 2: $G_2 = (V_2, E_2)$;
Dóng hàng mạng A_{12}; n_{best}
Output: Dóng hàng mạng tốt hơn
Begin
Giữ lại n_{best} cặp $< u_t, v_j >$ tốt nhất của V_{12}
For=n_{best}+1 **to** |V_1| **do**
u_i = find_next_node();
v_j = choose_best_matched_node(u_t);
$V_{12}=V_{12} \cup < u_i, v_j >$
Cập nhật E_{12}
end-for
end

Hình 3. Đặc tả thủ tục tìm kiếm cục bộ

Trong mỗi vòng lặp, sau khi tất cả các kiến đã xây dựng xong lời giải. Lời giải tốt nhất A_{12} được kiến xây dựng sẽ được áp dụng tìm kiếm cục bộ. Thủ tục tìm kiếm cục bộ được đặc tả như trong hình 3.

Bước 1. Giữ lại n_{best} đỉnh thuộc tập A_{12} có score tốt nhất theo tiêu chí cho bởi công thức (3):

$$score(u_i) = \alpha \times w(u_i) + (1-\alpha) \times similar(u_i, f(u_i)) \quad (3)$$

trong đó $u_i \in V_1$ và $f(u_i)$ là đỉnh thuộc V_2 được ghép với u_i trong A_{12}, $w(u_i)$ là số lượng nút $u_j \in V_1$ mà $< u_i, u_j > \in E_1$ và $< f(u_i), f(u_j) > \in E_2$

Bước 2. Thực hiện lặp với $k = n_{best} + 1$ tới $|V_1|$:

2.1. Thủ tục **find_next_node**(): Tìm đỉnh $u_i \in V_1 - V_{12}^1$ có số cạnh tới các đỉnh trong V_{12}^1 lớn nhất.

2.2. Thủ tục **choose_best_matched_node**(u_i) tìm đỉnh $v_j \in V_2 - V_{12}^2$ mà khi bổ sung $< u_i, v_j >$ vào V_{12} thì *GNAS(A_{12})* tính bởi công thức (1) lớn nhất, trong đó A_{12} là đồ thị có đỉnh là tập V_{12} và các cạnh cảm sinh bởi G_1, G_2.

2.3. Bổ sung $< u_i, v_j >$ vào V_{12};

2.4. Cập nhậtE_{12} dựa trên V_{12};

Sau mỗi lần thực hiện thủ tục tìm kiếm cục bộ ta có một dóng hàng mới làm input A_{12} cho lần lặp tiếp theo, quá trình này lặp lại cho đến khi không cải tiến được *GNAS(A_{12})* nữa.

IV. THỰC NGHIỆM

Thực nghiệm được tiến hành để so sánh thuật toán ACOGA với thuật toán FastAn [6], là thuật toán mới nhất và đã được chứng tỏ tốt hơn GNAS, trên 4 tập dữ liệu benchmark được sử dụng trong [1]. Thuật toán được chạy với nhiều giá trị n_{best} khác nhau bao gồm 1%, 5%, 10%, 20% và 50%. Các kết quả thực nghiệm chỉ ra rằng với n_{best}=1% sẽ cho chất lượng lời giải tốt nhất.

Đối với thuật toán đàn kiến, các tham số rho được khởi tạo từ đầu và số lượng kiến trong mỗi vòng lặp ảnh hưởng nhiều đến chất lượng lời giải. Qua nhiều thực nghiệm chúng tôi thấy với rho=0.3 sẽ cho chất lượng lời giải tốt nhất. Đối với số lượng kiến, nếu sử dụng nhiều kiến để xây dựng lời giải có thể cho được kết quả tốt hơn, tuy nhiên lại khiến thuật toán chạy lâu. Để cân bằng giữa hai yếu tố này, trong thực nghiệm chúng tôi chọn số kiến là 6. Các tham số T_{max} được khởi tạo bằng 1 và $T_{\min} = \frac{1}{|V_1| + |V_2|}$. Vì vậy các kết quả được trình bày ở đây tương ứng với giá trị n_{best}=1%, $\rho = 0.3$, n_{ants}=6 như đã phân tích ở trên.

Các thuật toán được so sánh dựa trên 2 tiêu chí là tiêu chuẩn GNAS và EC (Edge Correctness – Số cạnh được bảo tồn qua dóng hàng, hay $|E_{12}|$). Các thực nghiệm được chạy trên cùng một máy tính có cấu hình như sau: CPU Intel Core 2 Duo 2.53GHz, RAM DDR2 3GB và hệ điều hành Windows 7 32 bit.

A. Dữ liệu

Bộ dữ liệu được sử dụng so sánh để so sánh các phương pháp là 4 tập dữ liệu tiêu chuẩn được dùng để đánh giá chất lượng lời giải của SPINAL và FastAn. Đó là các mạng tương tác protein sau: Saccharomyces cerevisiae (sc), Drosophila melanogaster (dm), Caenorhabditis elegans(ce), and Homo sapiens (hs). Các mạng tương tác này thu được từ [22]. Mô tả về các tập dữ liệu này được chỉ ra trong Bảng 1. Từ các bộ dữ liệu đó chúng tôi tạo ra sáu cặp mạng để dóng hàng (*ce-dm,ce-hs,ce-sc,dm-hs, dm-sc,hs-sc*).

Bảng 1. Mô tả bộ dữ liệu

Bộ dữ liệu	Số Protein	Số tương tác
ce	2805	4495
dm	7518	25635
sc	5499	31261
hs	9633	34327

B. Kết quả thực nghiệm

Vì thuật toán ACOGA và FastAn là thuật toán ngẫu nhiên nên chúng tôi tiến hành chạy thuật toán 10 lần và sử dụng kết quả trung bình của 10 lần chạy để so sánh. Các thực nghiệm so sánh các thuật toán với các giá trị α lần lượt là 0.3, 0.4, 0.5, 0.6 và 0.7 như trong [1].

Bảng 2. So sánh thuật toán ACOGA và thuật toán FastAn theo 2 tiêu chuẩn GNAS và EC với các giá trị α khác nhau.

Datasets	α = 0.3		α = 0.4		α = 0.5		α = 0.6		α = 0.7	
	FASTAn	**ACOGA**	**FASTAn**	**ACOGA**	**FASTAn**	**ACOGA**	**FASTAn**	**ACOGA**	**FASTAn**	**ACOGA**
ce-dm	778.46 2560.7	**798.67** **2629.20**	1034.20 2564.6	**1057.34** **2622.9**	1290.11 2567.2	**1327.15** **2642**	1545.86 2567.7	**1601.13** **2660.4**	1801.24 2567.6	**1861.08** **2653.4**
ce-hs	863.46 2842.8	**885.47** **2916.1**	1144.17 2838.1	**1177.49** **2922.40**	1429.89 2844.9	**1461.74** **2909.1**	1708.81 2838.0	**1758.37** **2921.1**	1994.87 2843.4	**2049.1** **2921**
ce-sc	834.79 2761.1	**857.45** **2837.3**	1109.93 2761.2	**1144.56** **2849.4**	1389.21 2769.7	**1435** **2861.6**	1663.39 2766.5	**1688.11** **2808**	1936.83 2763.1	**1996.96** **2849**
dm-hs	2260.31 7478.3	***2315.78*** ***7663***	3007.11 7481.9	**3052.08** **7597**	3755.36 7429.0	**3803.79** **7584.3**	4496.45 7478.2	**4574.12** **7607.8**	5242.32 7478.8	**5319** **7588.6**
dm-sc	1977.82 6569.7	**2023.60** **6721**	2631.85 6565.5	**2653.53** **6619**	3290.03 6570.7	**3337.87** **6666.6**	3950.16 6577.4	**3989.68** **6643.30**	4603.41 6572.3	**4651.2** **6641.1**
hs-sc	2268.21 7531.8	**2300.318** **7640**	3017.96 7528.5	**3048.78** **7609.12**	3772.96 7535.2	***3838.3*** ***7666.0***	4520.51 7527	**4640.28** **7726.90**	5279.88 7538.1	***5422.18*** ***7742***

Các kết quả so sánh được thể hiện trong Bảng 2. Mỗi ô trong bảng thể hiện 2 tiêu chuẩn so sánh là tiêu chuẩn GNAS và tiêu chuẩn EC. Các kết quả tốt hơn được chúng tôi thể hiện bằng chữ in đậm.

Qua bảng 2 ta có thể thấy rõ trong tất cả các trường hợp thì thuật toán ACOGA đều cho các kết quả tốt hơn so với thuật toán FastAn đối với cả 2 tiêu chuẩn là GNAS và EC.

Về mặt thời gian chạy, do ACOGA là thuật toán metaheuristic được thực hiện với nhiều vòng lặp, nên thời gian chạy lâu hơn so với thuật toán FastAn (là một thuật toán theo hướng tiếp cận heuristic), vì vậy ở đây chúng tôi không đưa ra các bảng so sánh.

V. KẾT LUẬN

Bài báo này đề xuất một thuật toán dóng hàng toàn cục mạng tương tác protein dựa trên giải thuật tối ưu đàn kiến. Trong mỗi vòng lặp của thuật toán, tất cả các kiến xây dựng lời giải, sau đó kiến có chất lượng lời giải tốt nhất được lựa chọn để cập nhật vết mùi và áp dụng tìm kiếm cục bộ để tăng chất lượng lời giải. Các thực nghiệm trên bộ dữ liệu chuẩn đã chỉ ra rằng thuật toán chúng tôi đề xuất cho kết quả tốt hơn các thuật toán mới đề xuất đối với 2 tiêu chuẩn GNAS và EC đối với tất cả các trường hợp.

Thủ tục tìm kiếm cục bộ được sử dụng trong thuật toán phụ thuộc nhiều vào giá trị $n_{best,}$ hiện được chọn $n_{bes\,t}$ một cách thủ cộng. Trong thời gian tới chúng tôi sẽ nghiên cứu để có thể xác định được giá trị nbest một cách tự động để có thể cho chất lượng lời giải tốt nhất.

Ngoài ra để tăng chất lượng lời giải còn có thể tăng số lượng kiến trong mỗi vòng lặp. Tuy nhiên để không tốn thời gian trong mỗi lần chạy thì cần phải tiến hành song song hoá thuật toán ACOGA.

VI. TÀI LIỆU THAM KHẢO

[1] Aladag, A.E. and Erten, C. (2013), *SPINAL: scalable protein interaction network alignment*. Bioinformatics, Vol. 29 no 7, 917–924

[2] Bader,G.D. and Hogue,C.W. (2002), *Analyzing yeast protein-protein interaction data obtained from different sources*. Nat. Biotechnol., 20, 991–997.

[3] Banks,E. et al., (2008),*NetGrep: fast network schema searches in interactomes*. Genome Biology, 9,R138

[4] Chindelevitch,L. et al. (2010), *Local optimization for global alignment of protein interaction networks*. In: Pacific Symposium on Biocomputing,Hawaii,USA, pp. 123–132

[5] Chindelevitch L. et al. (2013*), Optimizing a global alignment of protein interaction networks*, Bioinformatics ,Vol. 29 no. 21,2765–2773.

[6] Đỗ Đức Đông, Trần Ngọc Hà, Đặng Cao Cường, Đặng Thanh Hải, Hoàng Xuân Huấn. (2015), *An efficient algorithm for global alignment of protein-protein interaction networks*, Proceeding of ATC15 (to appear), có thể xem bản preprint tại: ftp://file.viasm.org/Web/TienAnPham-14/Preprint_1418.pdf.

[7] Dost, B. et al. (2008),*QNet: a tool for querying protein interaction networks*. J. Comput. Biol., 15, 913–925

[8] Dutkowski,J. and Tiuryn,J. (2007), *Identification of functional modules from conserved ancestral protein–protein interactions*. Bioinformatics, 23, i149–i158.

[9] Han,J.D. et al. (2004), *Evidence for dynamically organized modularity in the yeast protein-protein interaction network*. Nature, 430, 88–93.

[10] H. Hoang Xuan, T. Nguyen Linh, D. Do Duc, H. Huu Tue, "Solving the Traveling Salesman Problem with Ant Colony Optimization: A Revisit and New Efficient Algorithms", REV Journal on Electronics and Communications, Vol. 2, No. 3–4, July – December,2012, 121-129

[11] B.H. Junker and F. Schreiber, *Analysis of Bological Networks*, wiley, 2008

[12] Kelley,B.P. et al. (2003), *Conserved pathways within bacteria and yeast as revealed by global protein network alignment*. Proc. Natl Acad. Sci. USA, 100, 11394–11399.

[13] Kelley,B.P. et al. (2004*), Pathblast: a tool for alignment of protein interaction networks*. Nucleic Acids Res., 32,83–88.

[14] Koyuturk,M. et al. (2006),*Pairwise alignment of protein interaction networks*. J. Comput. Biol., 13, 182–199.

[15] Kuchaiev,O. et al. (2010), *Topological network alignment uncovers biological function and phylogeny*. J. R. Soc. Interface., 7, 1341–1354.

[16] Kuchaiev,O. and Przulj, N. (2011) *Integrative network alignment reveals large regions of global network similarity in yeast and human*. Bioinformatics, 27, 1390–1396.

[17] Kuhn HW: *The Hungarian Method for the assignment problem*. Naval Res Logistics Q 1955, 2:83-97.

[18] Liao,C.S. et al. (2009) *IsoRankN: spectral methods for global alignment of multiple protein networks*. Bioinformatics, 25, i253–i258.

[19] Memisevic,V. and Przulj,N. (2012), *C-graal: common-neighbors-based global graph alignment of biological networks*. Integr. Biol., 4, 734–743.

[20] Milenkovic,T. et al. (2010), *Optimal network alignment with graphlet degree vectors*. Cancer Inform.,Vol.9, 121–137.

[21] Narayanan,M. and Karp,R.M. (2007), *Comparing protein interaction networks via a graph match-and-split algorithm.* J. Comput. Biol., Vol. 14, 892–907.

[22] Park,D. et al. (2011) *IsoBase: a database of functionally related proteins across PPI networks*. Nucleic Acids Res., 39, 295–300

[23] Remm,M. et al. (2001), *Automatic clustering of orthologs and in-paralogs from pairwise species comparisons*. J. Mol. Biol., 314, 1041–1052.

[24] Sharan,R. et al. (2005), *Conserved patterns of protein interaction in multiple species*. Proc. Natl Acad. Sci. USA, 102, 1974–1979.

[25] Singh,R. et al. (2008), *Global alignment of multiple protein interaction networks*. In: Pacific Symposium on Biocomputing. pp. 303–314.

[26] Zaslavskiy,M. et al. (2009) *Global alignment of protein-protein interaction networks by graph matching methods*. Bioinformatics, Vol.25, 259–267

AN EFFICIENT ANT BASED ALGORITHM FOR GLOBAL ALIGNMENT OF PROTEIN-PROTEIN INTERACTION NETWORKS

Tran Ngoc Ha, Hoang Xuan Huan

ABSTRACT - *Global alignment of two protein-protein interaction networks is an important problem in bioinformatics/ computational biology. This is studied by many researchers. Accuracy aligned networks allow us to identify functional orthologous proteins. In this article, we introduce an ant-based algorithm for global network alignment called ACOGA. The experiments show that the proposed method outperforms the state-of-the-art algorithms*

.