

# **Chương 1. Trí tuệ bầy đàn (Swarm Intelligence)**

Pham Duy Hung, PhD

Faculty of Electronics and Telecommunications

VNU-University of Engineering and Technology

Email: [hungpd@vnu.edu.vn](mailto:hungpd@vnu.edu.vn)

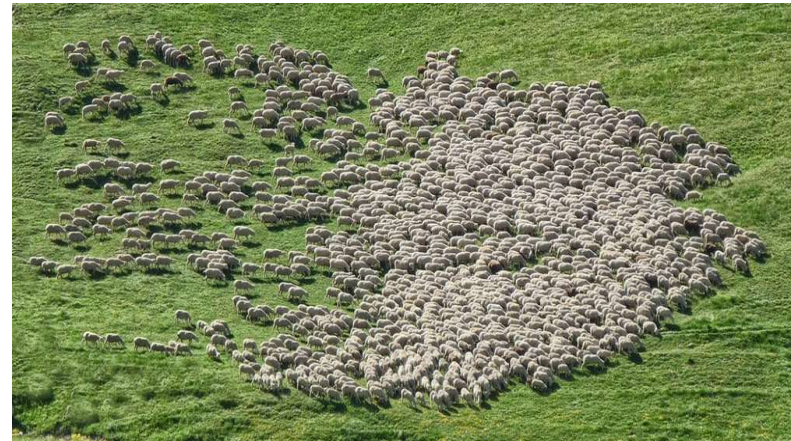
# Nội dung

- Trí tuệ bầy đàn SI
- Chiến lược kiếm mồi của kiến và thí nghiệm cầu đôi
- Thuật toán ACO
- Ứng dụng ACO

**Đọc:** [1] Bonabeau E., Dorigo M., and Theraulaz G., “Swarm Intelligence: From Natural to Artificial Systems”, SantaFe Studies in the Sciences of Complexity, Oxford University Press, 1999, Ch. 1 (pp.1-23) and Ch. 2 (pp. 25-36).

# **Trí tuệ bầy đàn SI**

Ví dụ, hiện tượng và các nguyên lý



**Một số hiện tượng bầy đàn tự nhiên hàm ý nói đến sự liên kết chặt chẽ giữa các cá thể ...**





# Flocking trong xã hội động vật

- **Có vẻ xảy ra ở**

- ✓ Tất cả các môi trường (không khí, nước, mặt đất)
- ✓ Nhiều họ động vật (côn trùng, cá, chim, động vật có vú,...)
- ✓ Từ nhóm nhỏ (2 con ngỗng) đến nhóm lớn (đàn cá trích dài 17 dặm)
- ✓ Động vật ở các lứa tuổi, kích cỡ khác nhau
- ✓ Ở một số loài động vật, chỉ trong hoàn cảnh đặc biệt (ví dụ di cư)

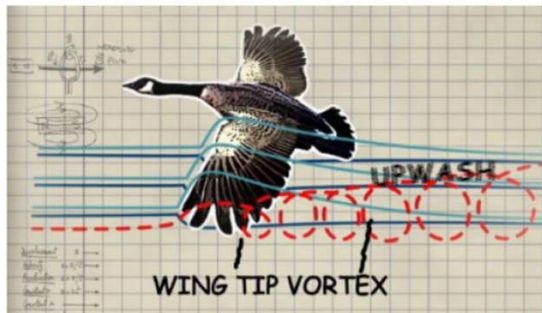
# Hiện tượng flocking

- Chuyển động nhanh của cả bầy được định hướng
- Khả năng phản ứng với động vật ăn thịt (...)
- Khả năng phản ứng với chướng ngại vật
- Không có va chạm giữa các thành viên trong đàn
- Kết dính và phân chia bầy
- Chịu được sự di chuyển trong bầy, mất hoặc tăng thành viên bầy
- Không có leader
- Các loài khác nhau có thể có các đặc tính bầy đàn khác nhau
- Dễ dàng nhận dạng nhưng không phải lúc nào cũng dễ miêu tả.

# Lợi ích của flocking

## 1. Tiết kiệm năng lượng

- Ví dụ: đội hình chữ V (V-formation)
  - ✓ Ngỗng bay trong đội hình chữ V có thể mở rộng phạm vi bay hơn 70%
  - ✓ Các loài chim theo đàn thường bay nhanh hơn khi bay một mình
- Lý do: mỗi con chim “cưỡi” trên dòng xoáy do đầu cánh của con chim phía trước tạo ra
- Người đi xe đạp tiết kiệm năng lượng theo cách tương tự

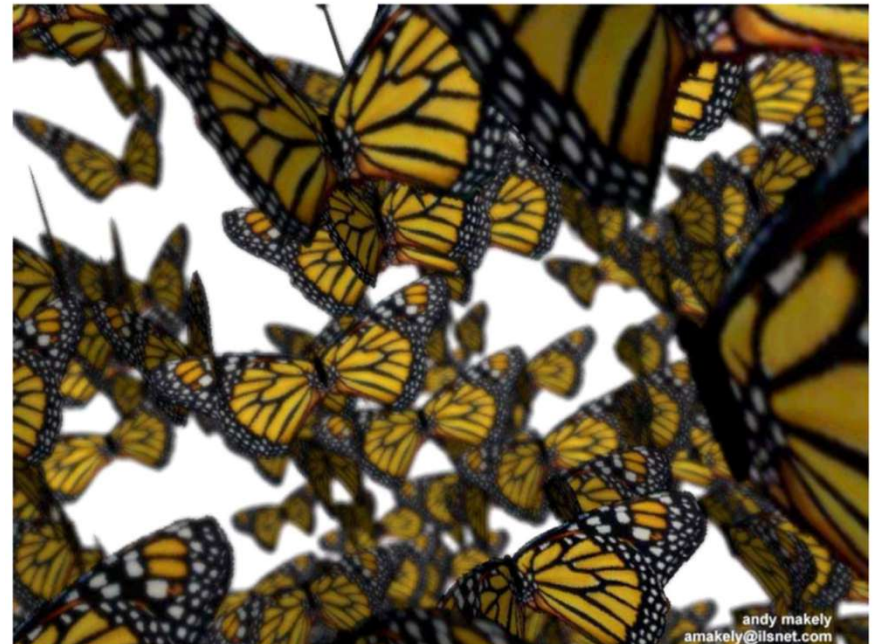




# Lợi ích của flocking

## 2. Dẫn đường chính xác

- Vài ví dụ
  - Bướm vua (Monarch Butterflies) tìm đến cùng cây hằng năm
  - Chim gõ kiến di cư (Wrynecks) di cư từ châu Phi đến Valais hằng năm
  - Cá tìm đến cũng một bãi đẻ trứng



# Hiện tượng bầy đàn (collective phenomena)

- **Thông tin cục bộ bị hạn chế (limited local information)**

Mỗi cá thể trong nhóm **chỉ có thể có thông tin cục bộ** và **không hiểu biết về cấu trúc toàn cục** (được tạo ra bởi các thành viên trong bầy).

- **Tập các quy tắc cá thể đơn giản (set of simple individual rules)**

Mỗi cá thể tuân theo **một bộ quy tắc hành vi đơn giản**. Các quy tắc này cho phép bầy phối hợp hoạt động để tạo ra cấu trúc toàn cục.

- **Các cấu trúc toàn cục (global structure) xuất hiện thực hiện một số chức năng**

Các cấu trúc này thường cho phép nhóm giải quyết một số bài toán. Chúng có tính **linh hoạt** (dễ thích nghi với môi trường mới), tính **bền vững** (robust – nếu một hoặc vài cá thể bị lỗi, cấu trúc tự hình thành lại).

**Từ tự nhiên đến các hệ thống nhân tạo  
và hơn thế...**

Fish School

# Trí tuệ bầy đàn (collective/swarm intelligence)

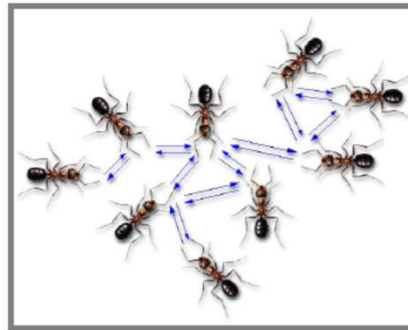
## Một vài câu hỏi ...

- Làm thế nào để xã hội động vật có thể thực hiện được các nhiệm vụ trong môi trường động mà **không có bất cứ chỉ dẫn hay điều khiển từ bên ngoài** ?
- Làm thế nào để **một số lượng lớn các thực thể chỉ có một phần thông tin** về môi trường có thể giải quyết được bài toán?
- Làm thế nào để **năng lực nhận thức tập thể (collective cognitive capacity)** có thể xuất hiện từ những cá thể có năng lực nhận thức hạn chế?

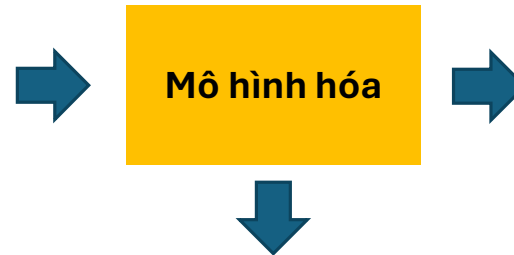


# Từ hệ thống tự nhiên đến nhân tạo

- Mô hình hóa để hiểu sự chuyển đổi từ vi mô (microscopic) sang vĩ mô (macroscopic).
- Mô hình hóa như giao diện (interface) với hệ thống nhân tạo.



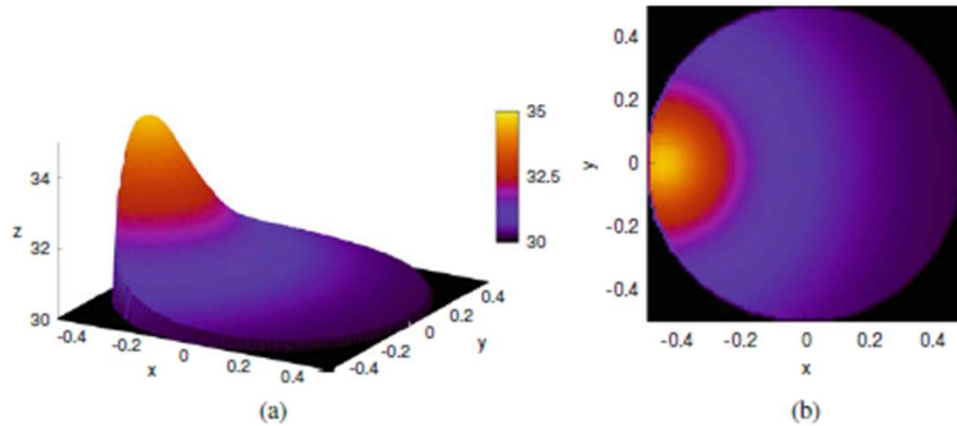
Hành vi **cá thể** và  
**tương tác cục bộ**



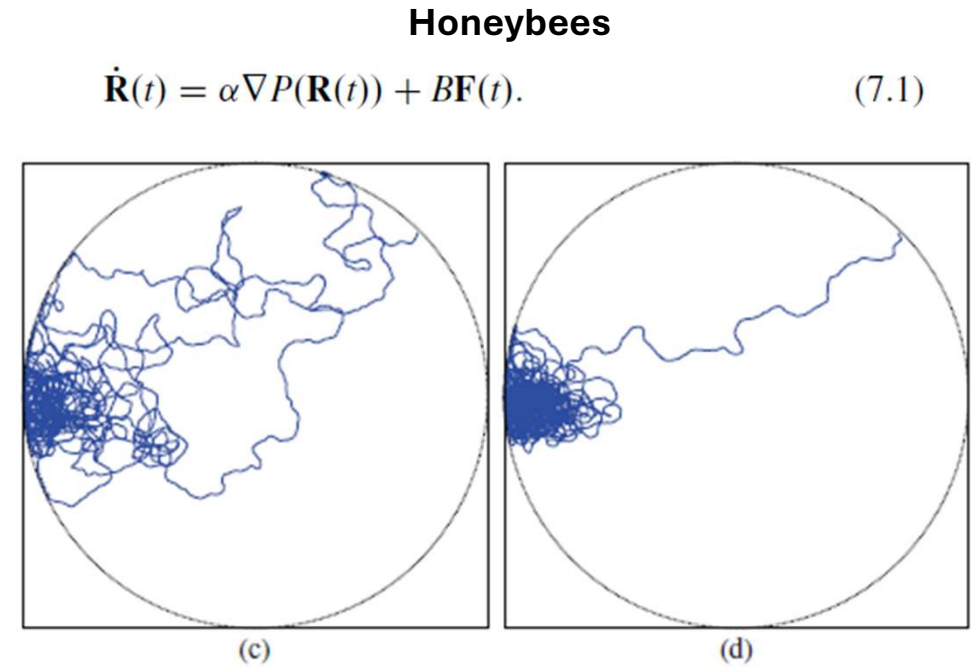
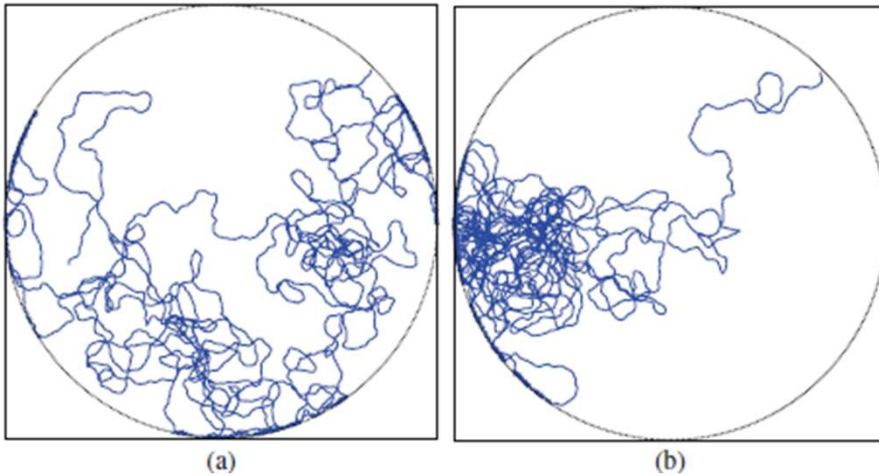
Ý tưởng cho các  
**hệ thống nhân tạo**



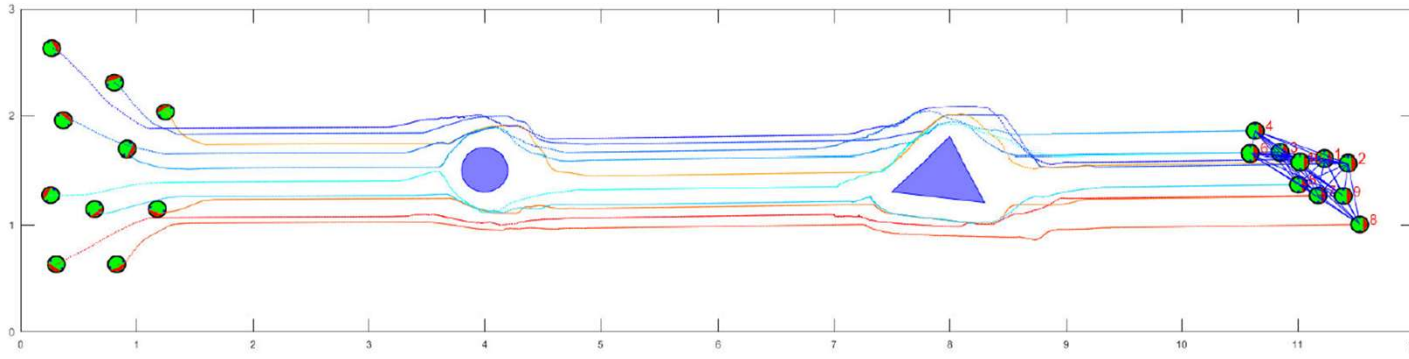
Cấu trúc **toàn cục**  
và **các quyết định**  
**tập thể**



**Fig. 7.1** Spatial temperature distribution for aggregation at the warmest spot. (a) Side view (b) Top view



**Fig. 7.3** Example trajectories sampled from the Langevin equation, Eq. (7.1), with four settings of parameter  $\alpha \in \{0.01, 0.025, 0.05, 0.1\}$  regulating the intensity of the robot's goal-oriented behavior. The desired warm spot is at the left border. (a)  $\alpha = 0.01$  (b)  $\alpha = 0.025$  (c)  $\alpha = 0.05$  (d)  $\alpha = 0.1$



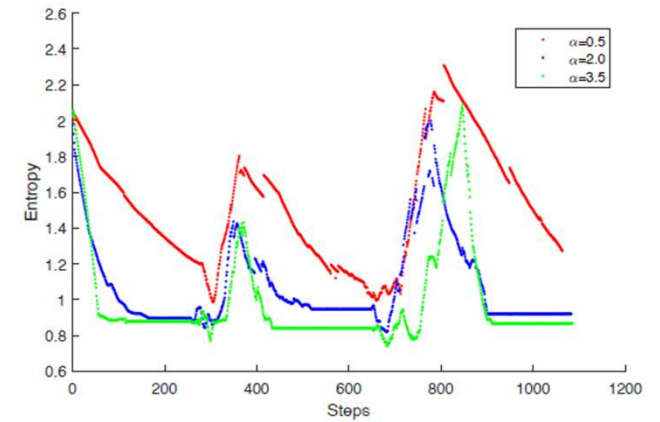
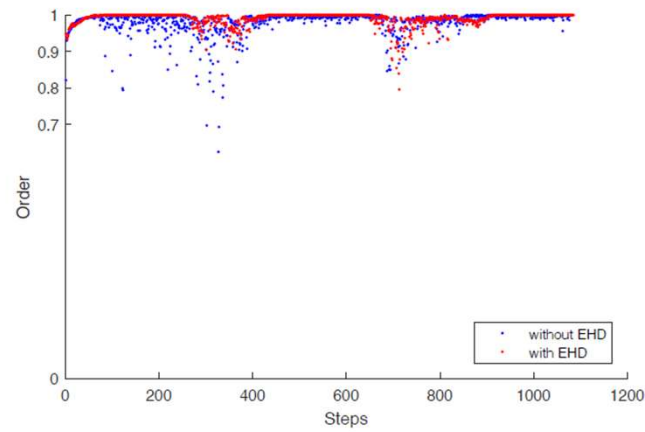
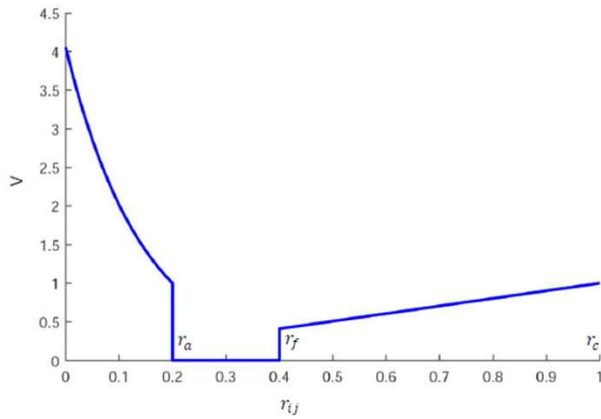
✓ Microscopic level

$$\vec{v}_i = \alpha \vec{v}_i^c + \beta \vec{v}_i^s + \gamma \vec{v}_i^a \quad (1)$$

✓ Macroscopic level

$$\Phi = \frac{1}{N} \left\| \sum_{j \in N} \hat{n}_j \right\| \quad (12)$$

$$S = \int_0^\infty H(h) dh \quad (13)$$



P. D. Hung and T. D. Ngo, "A Variant of Boids Rules: Reducing Jerky Motion through Eliminating Heading Disturbance," *2021 IEEE International Symposium on Robotic and Sensors Environments (ROSE)*, FL, USA, 2021, pp. 1-5, doi: 10.1109/ROSE52750.2021.9611763.

# Hệ thống trí tuệ bầy đàn số

(**Digital**-swarm intelligent systems)

- Trong thế giới ảo, hầu hết các cơ chế được thể hiện bởi SI tự nhiên đều có thể dễ dàng tái tạo.
- Các cơ chế được sửa đổi có chủ đích và có thể được thêm một số cơ chế khác để cải thiện hiệu suất của thuật toán.

# Hệ thống trí tuệ bầy đàn vật lý

## Physical Swarm-Intelligent systems

- **Lấy cảm hứng sinh học**
  - ✓ Tổ chức xã hội côn trùng
  - ✓ Bầy đàn ở động vật có xương sống

➡ **Kết hợp các nguyên lý tự nhiên với các kiến thức về kỹ thuật và công nghệ.**
- **Phối hợp**
  - ✓ Điều khiển hoàn toàn phân tán (fully distributed control)
  - ✓ Tự trị cá thể (individual autonomy)
  - ✓ Tự tổ chức (self-organization)
- **Truyền thông**
  - ✓ Truyền thông cục bộ explicit/implicit
  - ✓ Truyền thông gián tiếp thông qua ký hiệu trong môi trường (stigmergy)
- **Khả năng mở rộng (scalability)**
- **Tính bền vững (robustness)**

➡ **Thỏa hiệp về hiệu quả**
- **Hiệu quả chi phí hệ thống**
  - Sự đơn giản của cá thể
  - Sản xuất hàng loạt





# Một vài định nghĩa về trí tuệ bầy đàn SI

- Beni and Wang (1989)
  - ✓ Điều khiển **phân tán, không đồng bộ**, các **thành viên đơn giản**, (gần như) giống hệt nhau, **tự tổ chức**.
- Bonabeau, Dorigo and Theraulaz (1999)
  - ✓ Bất kỳ sự nỗ lực nào để thiết kế các **thuật toán hoặc thiết bị xử lý phân tán** đều được **lấy cảm hứng từ các hành vi bầy đàn** của xã hội côn trùng hoặc xã hội động vật.

# Một vài định nghĩa về trí tuệ bầy đàn SI

- Beni (2004)
  - ✓ Intelligent swarm = a group of **non-intelligent robots** (“machines”) capable of universal computation.
  - ✓ Thuật ngữ “intelligence” -> non predictable order from disorder, creativity.
- Dorigo and Sahin (2004)
  - ✓ Swarm robotics là nghiên cứu về cách thức một **số lượng lớn các tác nhân vật lý** (physical agent) **đơn giản** có thể được thiết kế sao cho một **hành vi bầy đàn** mong muốn **xuất hiện từ các tương tác cục bộ** giữa các tác nhân và giữa các tác nhân với môi trường.

# **Các cơ chế chính ảnh sau trí tuệ bẩm sinh tự nhiên**

# Hai cơ chế chính ẩn sau SI tự nhiên

1. **Self-Organization** (tự tổ chức)
2. **Stigmergy** (truyền thông gián tiếp)

# Tự tổ chức (self-organization)

- Tập hợp các cơ chế động, theo đó **cấu trúc** xuất hiện ở **mức toàn cục (global level)** là kết quả của sự **tương tác giữa các thành phần mức thấp hơn**.
- Các quy tắc xác định tương tác giữa các đơn vị cấu thành hệ thống được thực thi trên cơ sở **thông tin cục bộ thuần túy**, không tham chiếu đến mẫu tổng thể (global pattern). Trong trường hợp này, mẫu tổng thể **là thuộc tính mới xuất hiện của hệ thống** chứ không phải là thuộc tính được áp đặt lên hệ thống bởi sự tác động từ bên ngoài.
- Gắn với quan hệ microscopic level và macroscopic level.



# Đặc trưng của hệ thống tự tổ chức

- **Tạo ra các cấu trúc không gian – thời gian**
- **Đa trạng thái ổn định (Multistability):** khả năng tồn tại cùng một lúc nhiều trạng thái ổn định.
- **Tồn tại các cấu trúc khác nhau khi một số thông số thay đổi:** ví dụ như các con mồi chuyển từ giai đoạn không phối hợp sang giai đoạn phối hợp chỉ khi mật độ của chúng cao hơn giá trị ngưỡng

# Các thành phần của hệ thống tự tổ chức

- **Đa tương tác** (multiple interactions)
- **Tính ngẫu nhiên (Randomness)**
- **Phản hồi dương** (Positive feedback):
  - ✓ Ví dụ: recruitment và reinforcement
- **Phản hồi âm** (negative feedback):
  - ✓ Ví dụ: số lượng có hạn các foragers có sẵn và việc bay hơi các vết mùi.
- Cân bằng giữa khai thác và khám phá (Exploitation and exploration)

# Truyền thông gián tiếp (Stigmergy)

- Là một loại cơ chế được khai thác bởi xã hội côn trùng để điều phối và kiểm soát hoạt động của chúng thông qua **tương tác gián tiếp (indirect interactions)**.
- Cơ chế **stigmergy** có thể được chia thành 2 loại khác nhau:
  - ✓ Quantitative (continuous) stigmergy
  - ✓ Qualitative (discrete) stigmergy

# **Chiến lược kiếm tiền của đàn kiến và thí nghiệm cầu đôi**

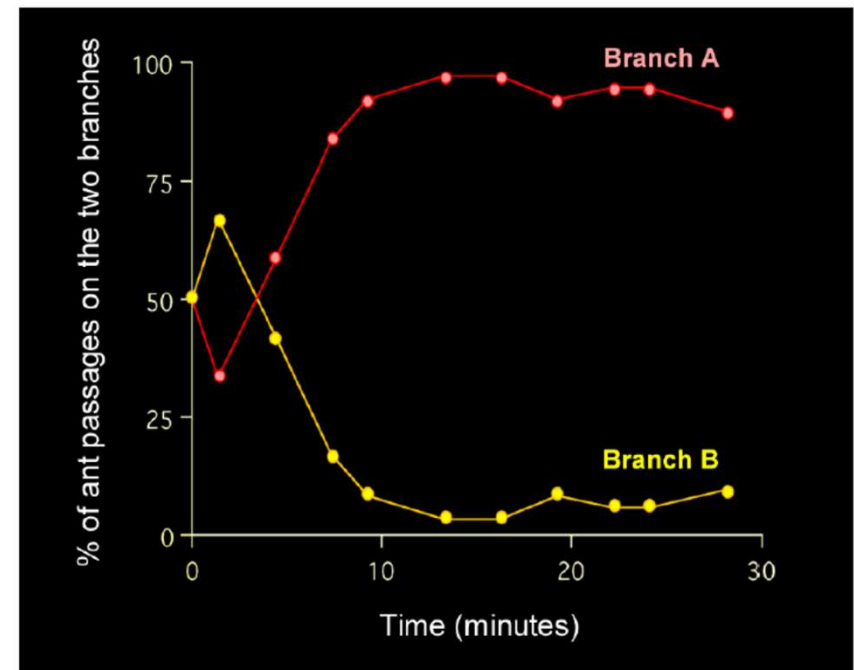
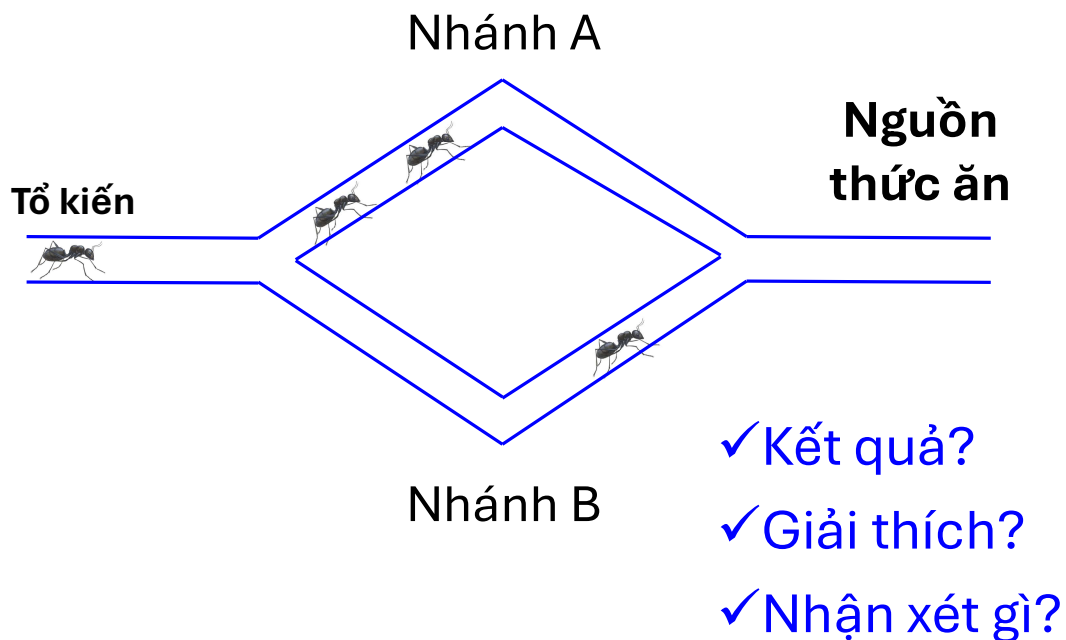
# Chiến lược kiếm mồi của đàn kiến

- Mỗi con kiến để lại một chất hóa học để đánh dấu đường đi gọi là **vết mùi (pheromon)**.
- Quá trình kiếm mồi, kiến có thể lần theo đường đi đến nguồn thức ăn được các con kiến khác khám phá theo phương thức chọn ngẫu nhiên, định hướng **theo nồng độ vết mùi**.
- Thí nghiệm cầu đôi của **Deneubourg**, 1990



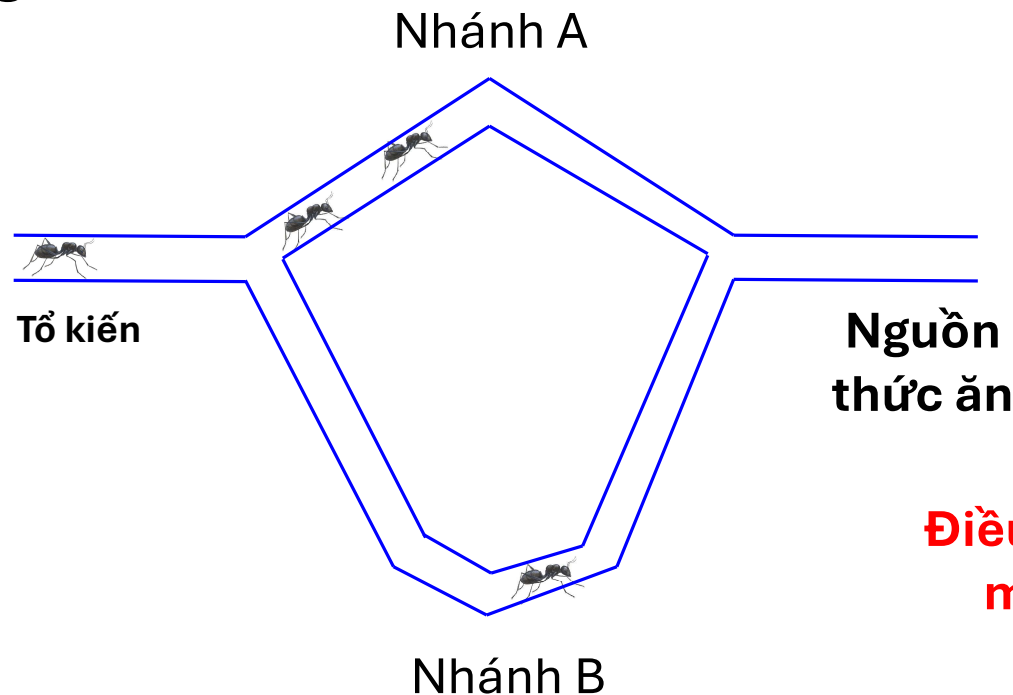
# Thí nghiệm cầu đôi (Deneubourg, 1990)

- **Thí nghiệm 1:** 2 nhánh A và B cùng độ dài nối tổ kiến và nguồn thức ăn



# Thí nghiệm cầu đôi (Deneubourg, 1990)

- **Thí nghiệm 2:** 2 nhánh A và B có độ dài khác nhau nối tổ kiến với nguồn thức ăn

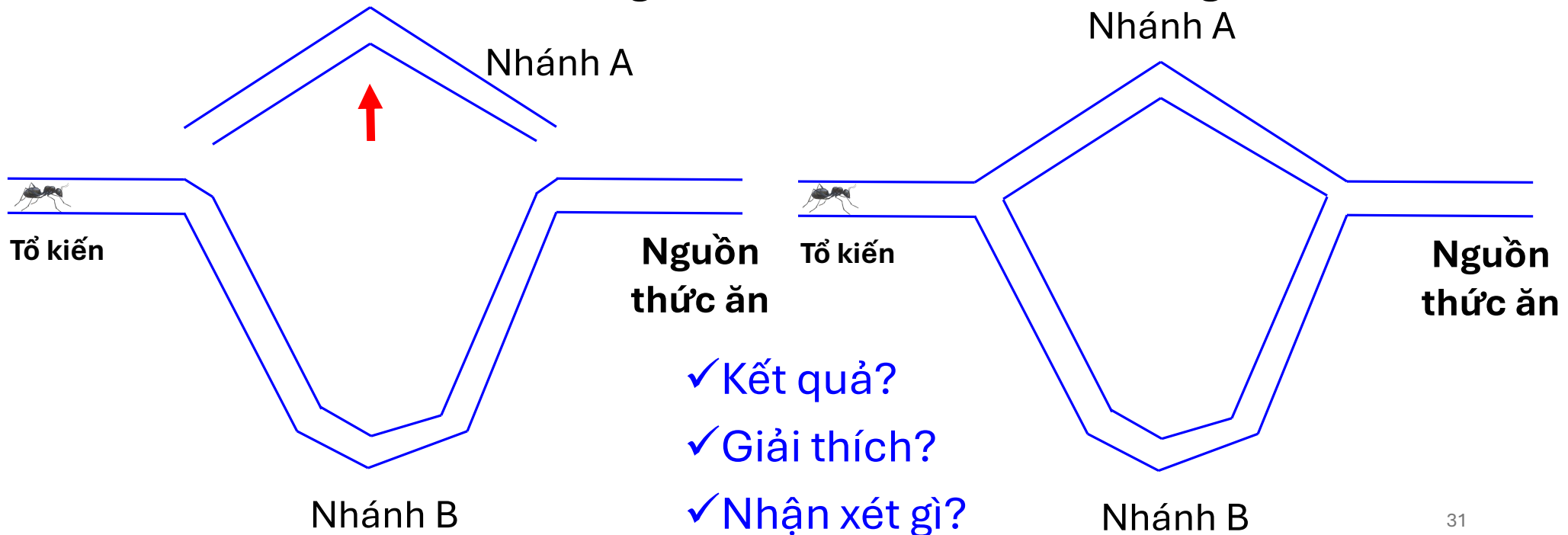


- ✓ Kết quả?
- ✓ Giải thích?
- ✓ Nhận xét gì?

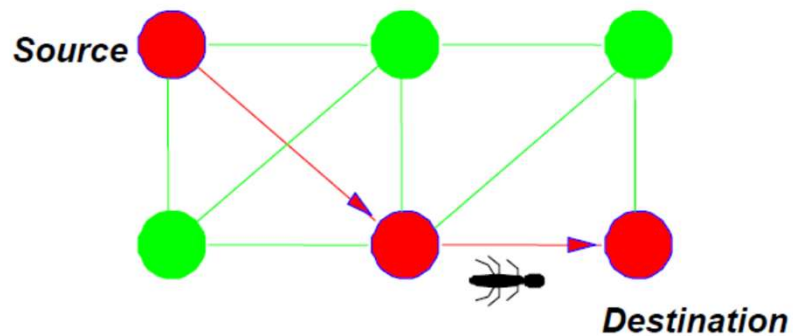
**Điều gì xảy ra nếu đàn kiến đã chọn một đường, lại xuất hiện thêm đường mới tốt hơn ?**

# Thí nghiệm cầu đôi (Deneubourg, 1990)

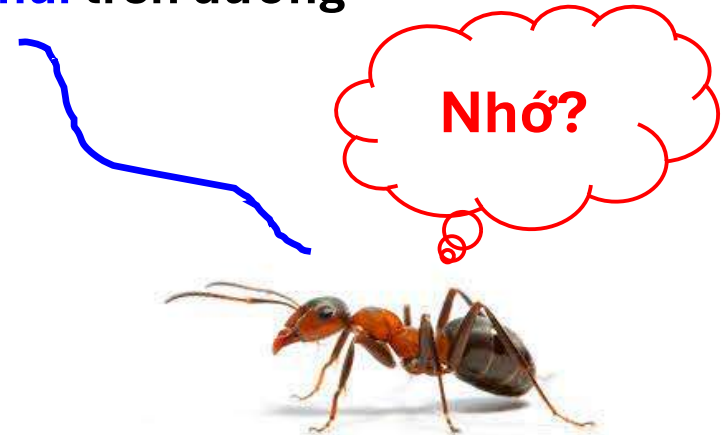
- **Thí nghiệm 3:** Ban đầu chỉ có 1 nhánh dài nối tổ kiến và nguồn thức ăn, và sau một thời gian đủ lớn thì thêm nhánh ngắn.



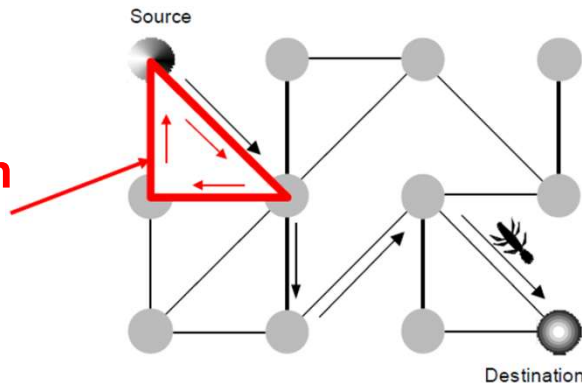
# **Kiến nhân tạo và bài toán tìm đường ngắn nhất**



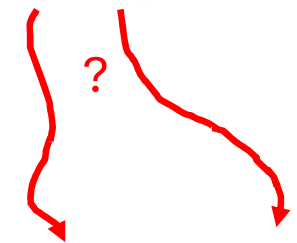
Đặt vết mùi trên đường



Vấn đề kiến  
đi luân  
quần



Hệ thống đa tác tử  
(Multi-Agent System):  
Kiến nhân tạo có nhiều  
khả năng hơn kiến tự  
nhiên



Quy tắc xác  
suất để chọn  
đường

# Hệ kiến AS (Ant System): Dorigo, 1991

- Kiến **có bộ nhớ** lưu các nút đã thăm
- Kiến **xây dựng lời giải theo xác suất**
- Kiến **cập nhật vết mùi**
- Kiến **đặt một lượng mùi** dựa trên chất lượng lời giải đã tạo.
- Vết mùi **bốc hơi** nhanh hơn trong tự nhiên

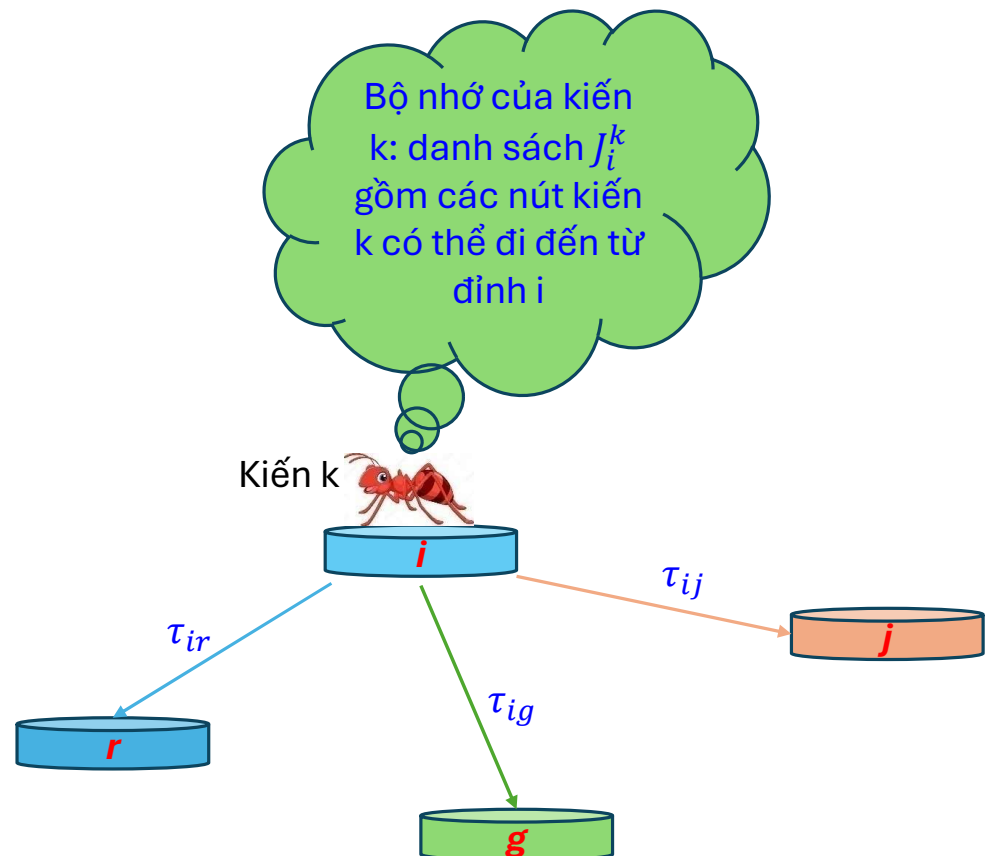
# Sử dụng Pheromone và bộ nhớ để chọn nút tiếp theo

$\tau_{ij}$ : lượng mùi ảo (virtual pheromon)  
được đặt trên link giữa nút i và j.

$$p_{ij}^k = f(\tau_{ij})$$

Quy tắc chuyển đổi dựa vào xác suất:

$$p_{ij}^k(t) = \frac{[\tau_{ij}(t)]^\alpha}{\sum_{l \in J_i^k} [\tau_{il}(t)]^\alpha}$$



# Quy tắc cập nhật vết mùi và bay hơi

- Cập nhật vết mùi:

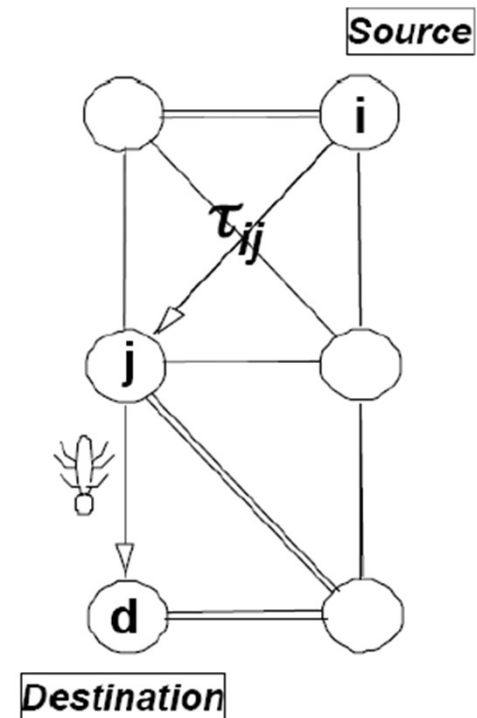
$$\tau_{ij}(t+1) \leftarrow (1 - \rho)\tau_{ij}(t) + \Delta\tau_{ij}(t)$$

với  $\rho$  là hệ số bay hơi.

$$\Delta\tau_{ij}(t) = \sum_{k=1}^m \Delta\tau_{ij}^k$$

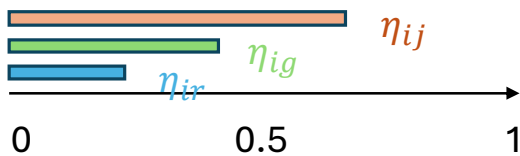
Tại bước  $t=0$ , mỗi link được khởi tạo với một lượng mùi nhỏ giống nhau  $\tau_0$ .

- Quy tắc cập nhật mùi thể hiện chiến lược **học** của thuật toán, các phiên bản khác nhau có quy tắc khác nhau, thường gắn với tên gọi thuật toán.





# Sử dụng Pheromone và heuristic để chọn nút tiếp theo

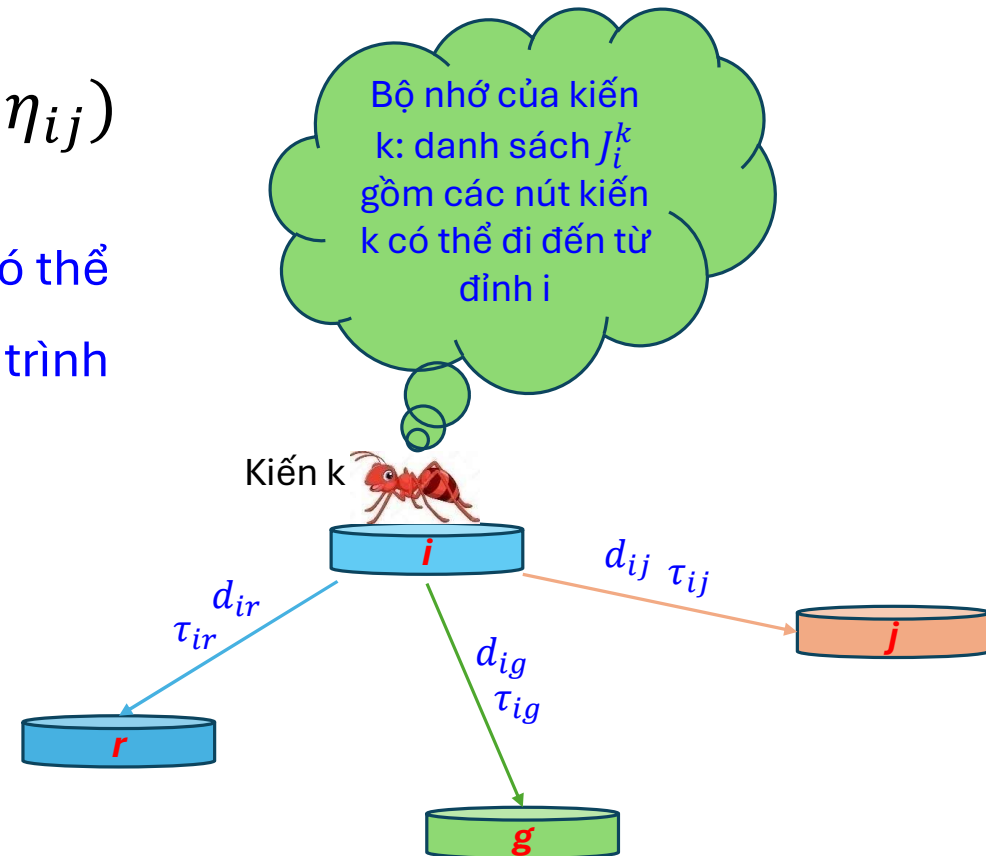


$$p_{ij}^k = f(\tau_{ij}, \eta_{ij})$$

$\eta_{ij} = \frac{1}{d_{ij}}$ : thông tin heuristic giữa nút  $i$  và nút có thể thăm  $j$ .  $\eta_{ij}$  tĩnh, không thay đổi trong suốt quá trình giải bài toán.

$$p_{ij}^k(t) = \frac{[\tau_{ij}(t)]^\alpha [\eta_{ij}]^\beta}{\sum_{l \in J_i^k} [\tau_{il}(t)]^\alpha [\eta_{il}]^\beta}$$

- ✓  $\alpha$  hệ số ảnh hưởng của pheromone
- ✓  $\beta$  hệ số ảnh hưởng của heuristic



# Thuật toán ACO

Loop  $\backslash^* t = 1^* \backslash$

Place one ant on each node  $\backslash^* \text{there are } n = |N| \text{ nodes } \backslash^*$

For  $k := 1$  to  $m$   $\backslash^* \text{each ant builds a tour, in this case } m=n \backslash^*$

For step  $:= 1$  to  $n$   $\backslash^* \text{each ant adds a node to its path } \backslash^*$

Choose the next node to move by applying a probabilistic *state transition rule*

End-for

End-for

*Update pheromone trails*

Until End\_condition  $\backslash^* \text{e.g., } t = t_{\max} \backslash^*$

- ✓ Mỗi kiến được đặt ngẫu nhiên vào một nút
- ✓  $m$  kiến đồng thời xây dựng lời giải bài toán
- ✓ Mỗi kiến  $k$  lần lượt khám phá các nút, sử dụng quy tắc xác suất (heuristic+vết mùi) để quyết định chọn nút thăm tiếp theo.
- ✓ Cập nhật vết mùi
- ✓ Dừng khi đạt điều kiện thời gian

# Quy tắc chuyển đổi

- $J_i^k$ : danh sách các nút lân cận nút  $i$  (nơi kiến  $k$  đang đứng) mà kiến  $k$  có thể thăm.
- $T$ : tour, gồm  $n=N$  steps với  $N$  là số nút trong bài toán, trong đó quy tắc chuyển đổi xác suất như dưới đây được áp dụng.
- $t$ : chỉ số vòng lặp, số lần thuật toán chạy  $1 \leq t \leq t_{max}$ 
  - ✓ Trong một tour  $T$ , kiến  $k$  ở nút  $i$  quyết định di chuyển về phía nút  $j$  với xác suất sau.

$$p_{ij}^k(t) = 0$$

Nếu nút  $j$  đã được thăm bởi kiến  $k$

$$p_{ij}^k(t) = \frac{[\tau_{ij}(t)]^\alpha [\eta_{ij}]^\beta}{\sum_{l \in J_i^k} [\tau_{il}(t)]^\alpha [\eta_{il}]^\beta}$$

Nếu nút  $j$  chưa được thăm bởi kiến  $k$

✓ **Quy tắc chọn nút:  
Roulette wheel**

$r = rand$

$C = cumsum(P)$

$J = find(r \leq C, 1, 'first')$

$\alpha$ : tham số điều khiển ảnh hưởng của vết mùi (ảo)

$\beta$ : tham số điều khiển ảnh hưởng của heuristic

# Cập nhật vết mùi

- Kết thúc mỗi tour  $T$  của cả đàn kiến, mỗi kiến  $k$  đặt một lượng mùi (ảo)  $\Delta\tau_{ij}^k$  trên link  $e_{ij}$ .
  - ✓  $\Delta\tau_{ij}^k = 0$ , khi  $e_{ij}$  không được sử dụng trong tour  $T$
  - ✓  $\Delta\tau_{ij}^k = \frac{Q}{L^k(t)}$ , khi  $e_{ij}$  được sử dụng trong tour  $T$Với  $L^k(t)$  là tổng chiều dài tour  $T$  đã thực hiện bởi kiến  $k$  ở vòng lặp  $t$ ;  $Q$  là tham số hiệu chỉnh bởi heuristic
- Lượng mùi được cập nhật trên cạnh  $e_{ij}$  bằng tổng lượng mùi của tất cả các kiến  $k$

$$\Delta\tau_{ij}(t) = \sum_{k=1}^m \Delta\tau_{ij}^k$$

# Các biến thể của ACO

- Ant System – AS (Dorigo 1991)
- Elitist AS (EAS) (Dorigo 1996)
- Rank-Based AS (ASrank) (Bullnheimer, 1997, 1999)
- Max-Min AS (MMAS) (Stutzle and Hoos, 1997)
- Ant Colony System (ACS) (Dorigo and Gambarbella, 1997)
- Approximate Nondeterministic Tree Search ANTS (Maniezzo, 1999)
- Hypercube AS (Blum, Roli and Dorigo, 2001)

# Ứng dụng ACO

- ✓ Bài toán người bán hàng (TSP)
- ✓ Lập lịch (Scheduling), Phân nhiệm (Task allocation)
- ✓ Định tuyến mạng (Network Routing)
- ✓ Lập kế hoạch robot (Robot Path Planning)

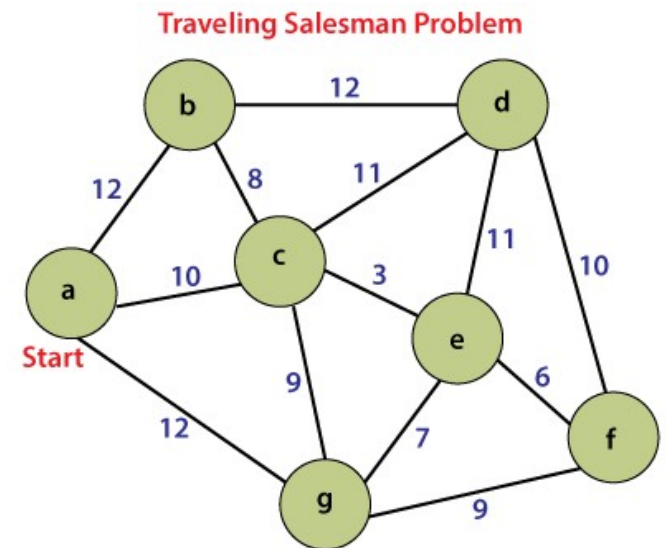
# ACO cho bài toán người giao hàng TSP

(Traveling Salesman Problem)

- **Bài toán:** tìm đường đi ngắn nhất cho phép người giao hàng thăm quan mỗi thành phố đúng 1 lần.
- **TSP:** bài toán tìm chu trình **Hamilton** có độ dài ngắn nhất trên đồ thị đầy đủ  $G(N,E)$ .

$$f(S) = \sum_{(i,j) \in S} d_{ij}$$

- Độ khó: bài toán NP-hard, thời gian tính toán đường đi ngắn nhất tăng theo cách phi đa thức với số lượng thành phố trong mạng -> Metaheuristics/học máy (như ACO, GA) cung cấp lời giải gần tối ưu.



## Đồ thị $G(N,E)$

N: tập các thành phố (nút)

E: tập các đường kết nối (cạnh)

$d_{ij}$  khoảng cách giữa thành phố  $i$  và  $j$

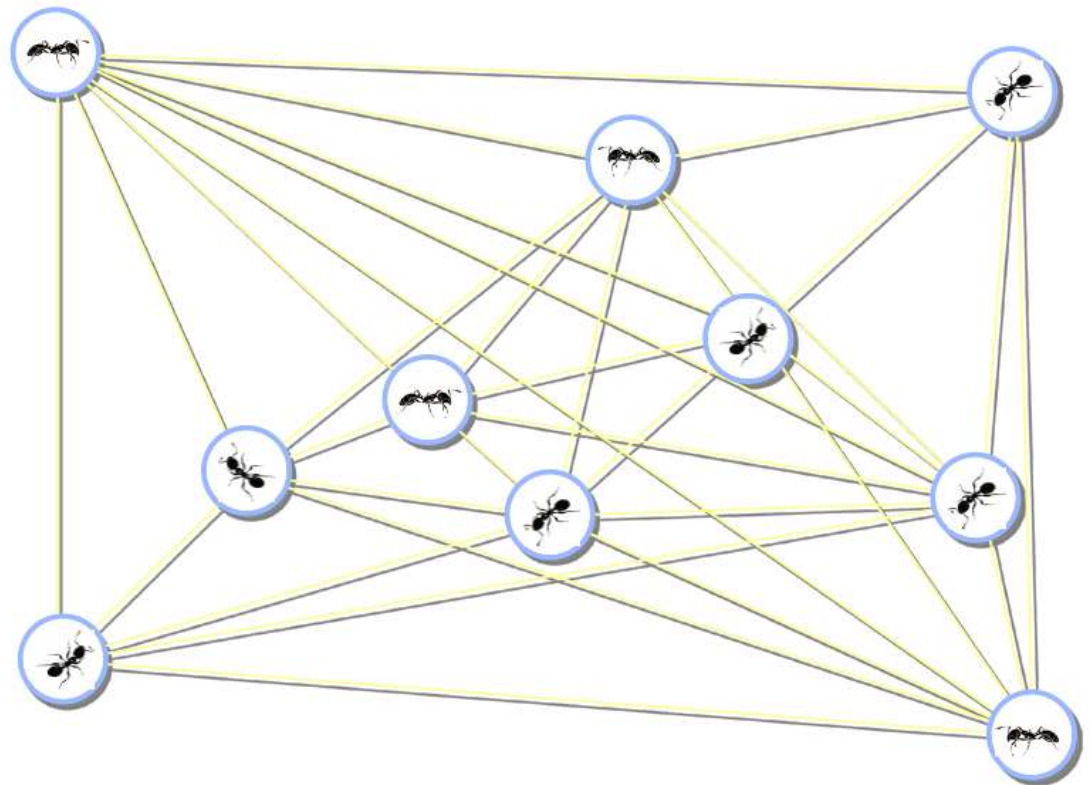
# Hình dung về độ khó bài toán NP-hard

- TSP- thuật toán Brute force:
  - ✓ TSP với 30 thành phố sẽ phải đo tổng khoảng cách là  $2,65 \times 10^{32}$  tour khác nhau. Giả sử nghìn tỷ (trillion) phép tính/giây, thì lời giải này cần 252.333.390.232.297 năm.
  - ✓ Thêm 1 thành phố nữa sẽ làm cho thời gian tăng thêm một hệ số 31.
- TSP – Thuật toán metaheuristic
  - ✓ 10 nghìn thành phố (Applegate và các tác giả 2006)
  - ✓ Metaheuristic methods: hàng triệu thành phố

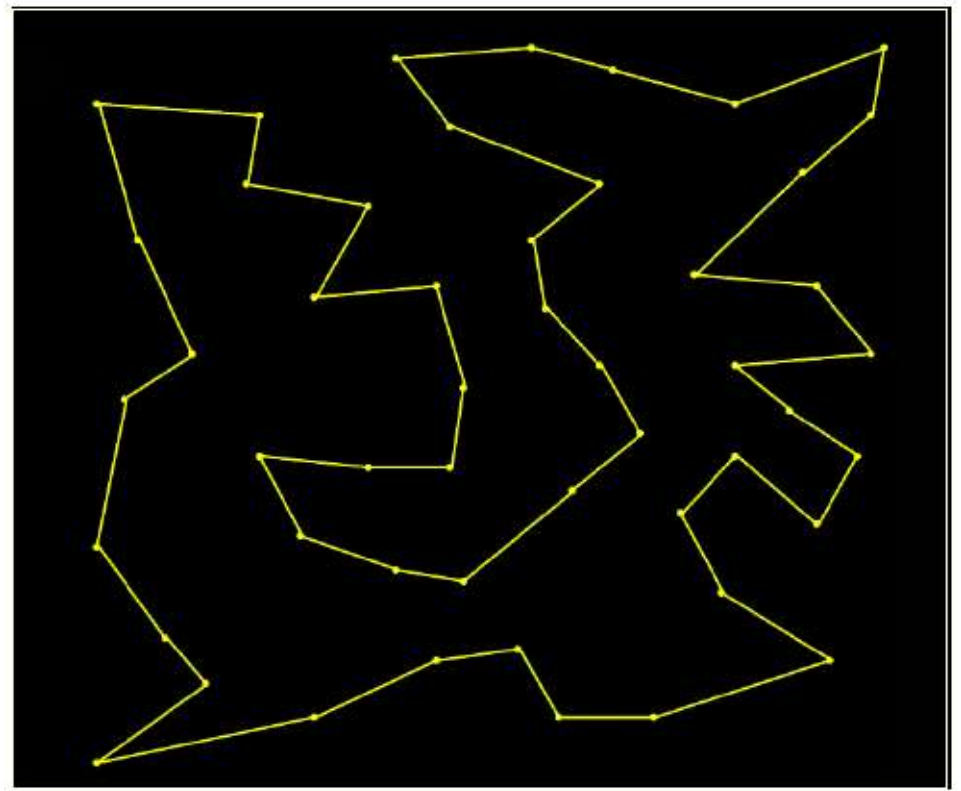
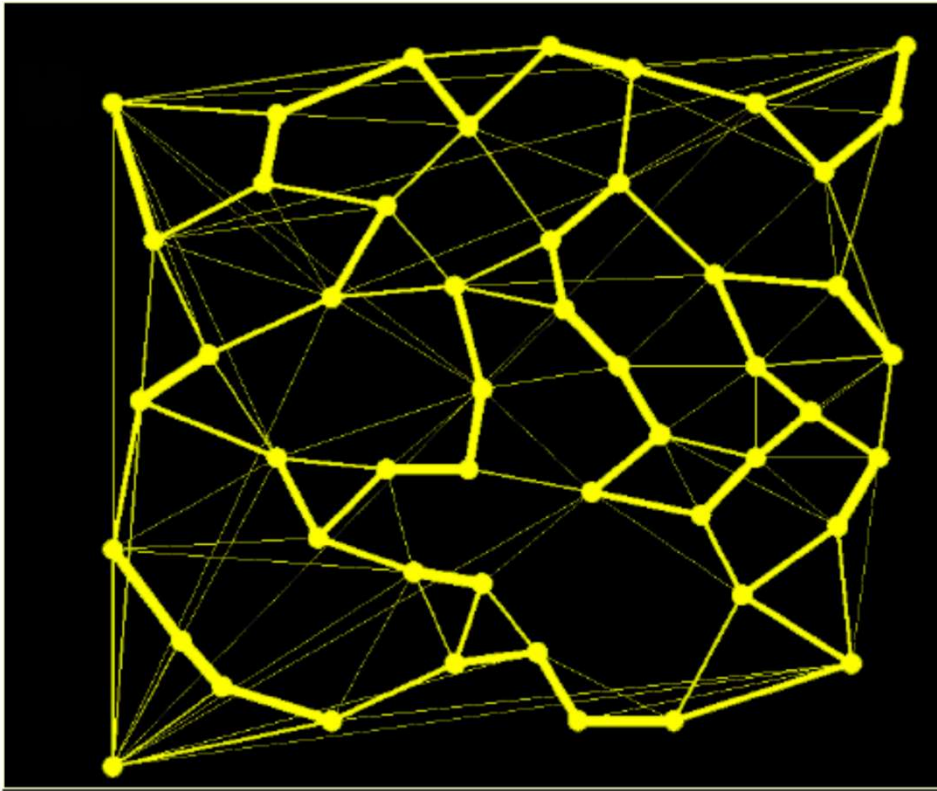


# AS cho TSP

- m con kiến nhân tạo (m Agents) đồng thời xây dựng lời giải.
- Ban đầu các con kiến được đặt ngẫu nhiên trên các thành phố.

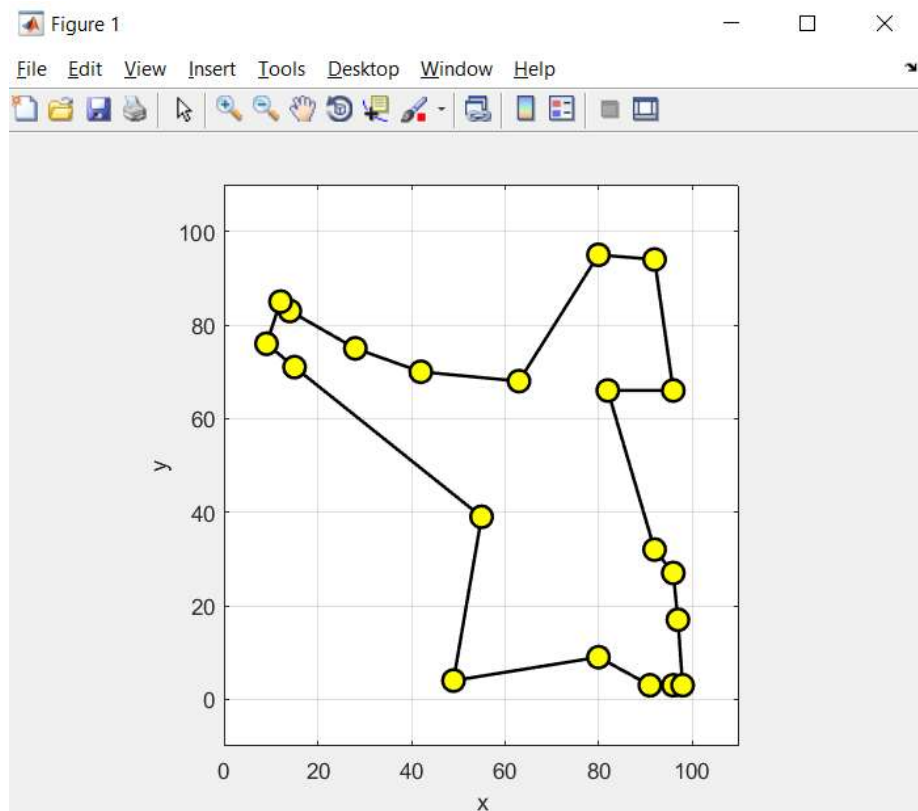


## AS cho TSP: phân tích kết quả

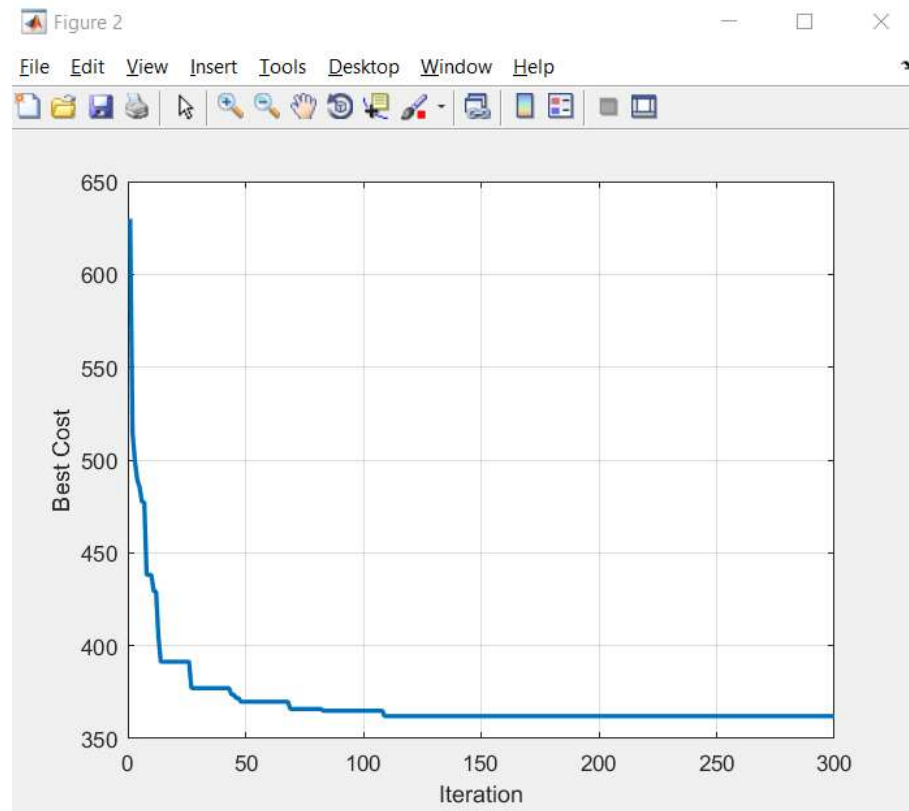


Lời giải với 50 cities

# AS cho TSP: phân tích kết quả



Lời giải với 20 cities



Phát triển của chiều dài tour tốt nhất, 20 cities

# AS cho TSP: phân tích kết quả

- Hiệu suất là hàm của chiều bài toán

Network	n (dimension)	best solution	Mean number of iterations for to the near-optimal solution	Simulation time (seconds)
4 X 4	16	160	5,6	8
5 X 5	25	254,1	13,6	75
6 X 6	36	360	60	1020
7 X 7	49	494,1	320	13440
8 X 8	64	640	970	97000

# Bài toán lập lịch (scheduling)

- Lập lịch trên máy song song không đồng nhất, tối thiểu makespan (thời gian hoàn thành muộn nhất)
- ✓ Có m Máy (M1, M2,...Mm)
- ✓ Có n công việc (J1, J2,... Jn)
- ✓ Mỗi công việc có thời gian xử lý khác nhau trên mỗi máy.

Phân công công việc cho các máy sao cho makespan nhỏ nhất.

$$f(S) = \max_i C_i$$

# Bài toán lập lịch (scheduling)

- Ví dụ

Công việc	M1	M2	M3
J1	10	12	8
J2	5	7	6
J3	8	4	9
J4	6	5	7
J5	7	8	6

# Bài toán định tuyến mạng

- Tìm đường đi tối ưu cho các gói dữ liệu từ nguồn đến đích trong mạng máy tính hoặc mạng viễn thông sao cho các tiêu chí tối ưu (chi phí, độ trễ, băng thông,...) được thỏa mãn.
- $G(V,E)$ 
  - ✓  $V$  là tập đỉnh (routers, switches, servers,...);
  - ✓  $E$  tập kết nối với trọng số cạnh  $w_{ij}$

⇒ Tìm đường đi tốt nhất giữa nút  $S$  và  $D$

- Hàm mục tiêu 
$$f(S) = \sum_{(i,j) \in S} w_{ij}$$

# Bài tập

- Áp dụng ACO trong bài toán tìm đường đi của robot, so sánh kết quả với các giải thuật tìm đường khác.