

Swarm Intelligence (SI)

- Collective behavior of decentralized, self-organized systems. (G.Beni and J.Wang (1989, cellular robotics))



Swarm Intelligence

- Collective system capable of accomplishing difficult tasks in dynamic and varied environments **without any external guidance or control and with no central coordination**
- Achieving a collective performance which could not normally be achieved by an individual acting alone

(ex)

Ant colony optimization (ACO) algorithm,
Artificial bee colony (ABC) algorithm,
Artificial immune systems (AIS),
Particle swarm optimization (PSO),
.....

- به گروه موجودات ازدحام گفته می شود.
- یک ازدحام ، گروهی از عاملها هستند که با یکدیگر از طریق فعالیت در محیط محلی شان به صورت مستقیم یا غیر مستقیم ارتباط برقرار می کنند.
- هوش ازدحامی SA به رفتار حل مسئله ای گفته می شود که از تعامل چنین موجوداتی پدید می آید.
- در این ازدحام ها موجودات ساختار ساده ای دارند
- اما رفتار جمعی آنها بسیار پیچیده است.
- رفتار پیچیده ازدحام در نتیجه الگوهای ارتباطی بین آن موجودات در طول زمان است.
- ای رفتار پیچیده ویژگی هیچ موجود به تنهایی نیستو معمولا به آسانی از رفتار ساده موجودات قابل پیش بینی و نتیجه گیری نیست.
- به این پدیده ظهور می گویند . / (Emergence)
- هوش جمعی به صورت غیر خطی از رفتار موجودات یک ازدحام بوجود می آید.
- **مهمترین بخش هوش جمعی تعامل یا همکاری است.**
- **تعامل می تواند مستقیم از طریق ارتباط فیزیکی یا ورودی های ادراکی بینایی شنوایی ... باشد و**
- **یا غیر مستقیم از طریق تغییر محیط باشد. (stigmercy)**

روشهای هوش جمعی

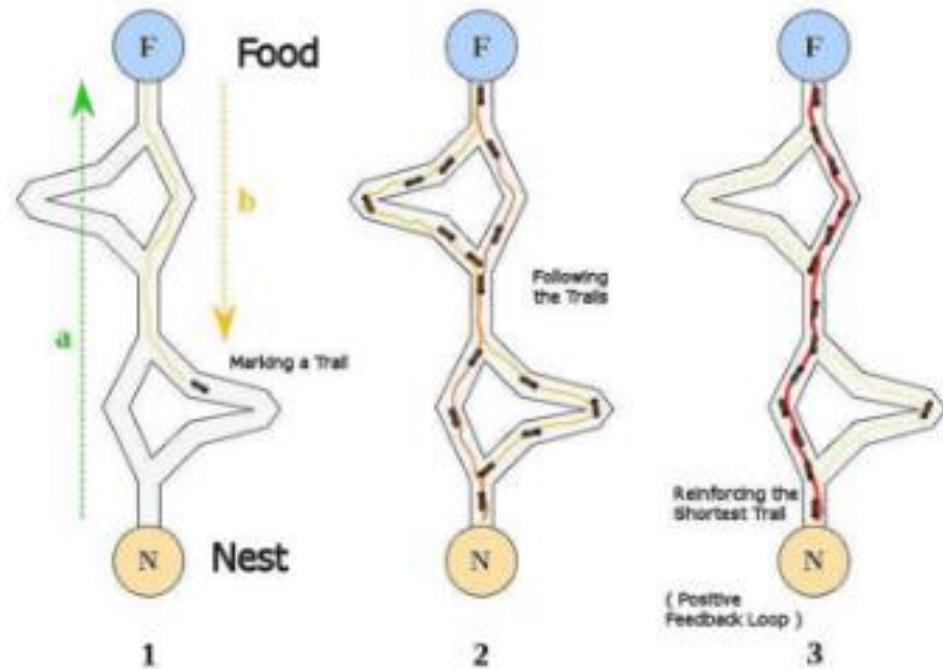
• روش های علامت محور

- یک سیستم خودسازمانده هستند.
- از یک حافظه محیطی مشترک برای برقراری ارتباط غیر مستقیم میان موجودات استفاده می کنند.
- ارتباط میان موجودات به صورت غیر مستقیم بوده و فاقد رفتارهای تقلید گونه مستقیم هستند.
- محیط آنها گسسته است.
- عدم وجود رهبر و یا واحد کنترل مرکزی

• روشهای تقلید محور

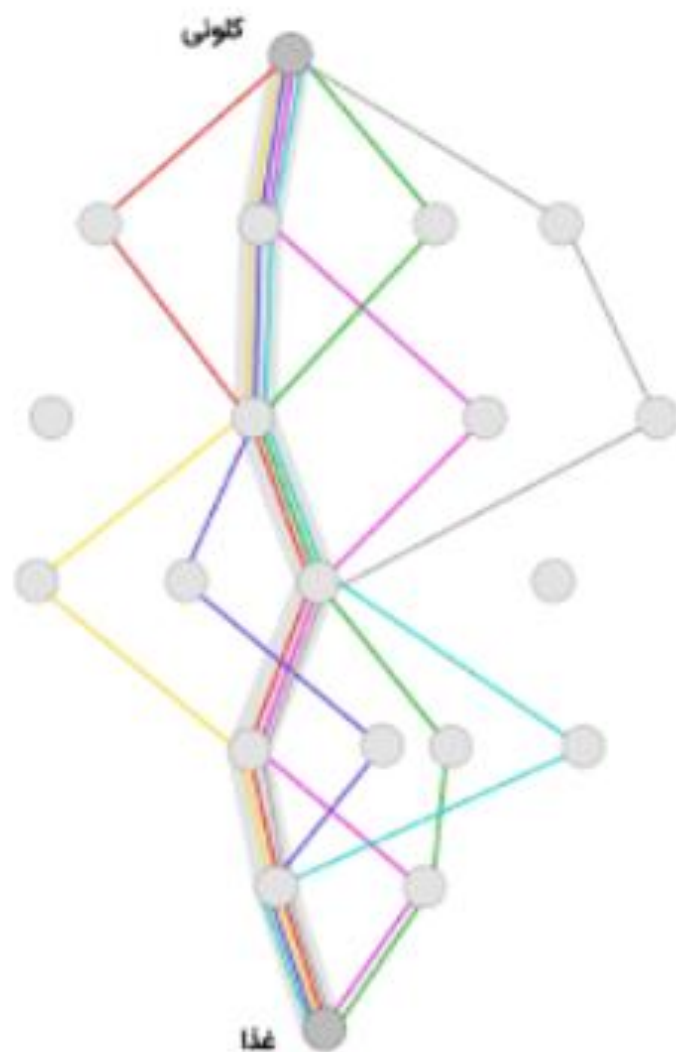
- از مفهوم سرعت و اینرسی و یا تمایل حرکتی در فرآیند جستجو استفاده می کنند.
- ارتباط مستقیم بین موجودات است و حافظه مشترک ندارند.
- هر موجود یک حافظه دارد که در آن بهترین مکان یافته شده خود (پاسخ برازنده محلی) و بهترین موجودات جمعیت (پاسخ برازنده سراسری) را نگه می دارد.
- موجود تمایل به سمت بهترین پاسخ های محلی و سراسری دارد.

Ant colony Optimization Algorithm (ACO)



Ant Colony Optimization

- الگوریتم کلونی مورچگان یا الگوریتم بهینه سازی کلونی مورچه Ant Colony Optimization الهام گرفته شده از مطالعات و مشاهدات روی کلونی مورچه هاست.
- این مطالعات نشان داده که مورچه‌ها حشراتی اجتماعی هستند که در کلونی‌ها زندگی می‌کنند و رفتار آنها بیشتر در جهت بقاء کلونی است تا در جهت بقاء یک جزء از آن.
- یکی از مهمترین و جالبترین رفتار مورچه‌ها، رفتار آنها برای یافتن غذا است و به ویژه چگونگی پیدا کردن کوتاهترین مسیر میان منابع غذایی و آشیانه.
- از آنجایی که مورچه‌ها اصلاً چشم ندارند، تعاملات آن‌ها از طریق ماده شیمیایی فرومون Pheromone که از آن برای نشان گذاری مسیر استفاده می‌شود، انجام می‌گیرد.
- هرچه فرومون‌های بیشتری در مسیر قرار گیرد مابقی مورچه‌ها از این مسیر بیشتر استفاده می‌کنند؛ بنابراین، چنین کمیتی نشان می‌دهد که این مسیر به عنوان یکی از بهینه‌ترین و کوتاه‌ترین راه می‌باشد.



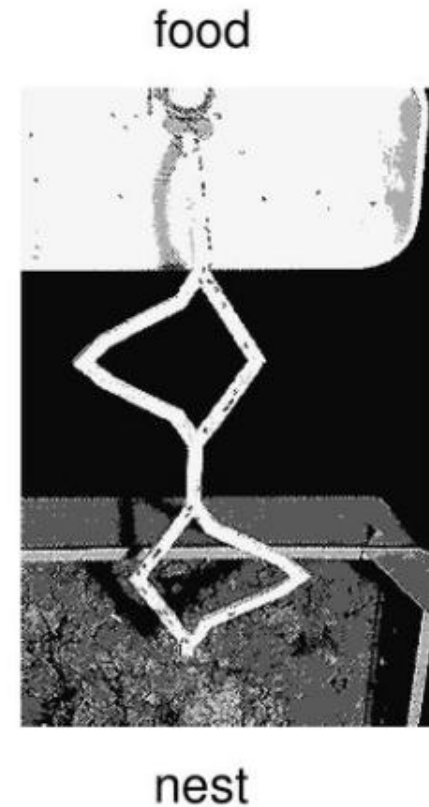
- این فرومون ها جذاب هستند و مورچه های مجاور تمایل دارند کم و بیش مستقیم مسیر را دنبال کنند.
- با بازگشت به کلونی این مورچه ها مسیر را بوسیله فرومون تقویت می کنند.
- اگر دسترسی به یک منبع غذایی به دو طریق امکان پذیر باشد، مسیر کوتاهتر در همان لحظه نسبت به مسیر طولانی تر توسط تعداد بیشتری مورچه طی خواهد شد.
- رفت و آمد از مسیر کوتاه به طور فزاینده ای افزایش می یابد از آنجایی که فرومون ها فرار هستند، در نهایت مسیر طولانی ناپدید می شود.

Ant's Foraging Behavior

- Experiments with Argentine ants

(Goss et.al, 1989)

- Ants go from the nest to the food source and backwards
- after a while, the ants prefer the shortest path from the nest to the food source
- **stigmergy**:
 - the ants communicate **indirectly** laying **pheromone** trails and following trails with higher pheromone
 - length gradient → pheromone will accumulate on the shortest path



Ant Colony Optimization

- A **probabilistic** technique for solving computational problems which can be reduced to finding good paths through **graphs**.
- A member of swarm intelligence methods, and it constitutes some **metaheuristic** optimization
- Initially proposed by Marco Dorico in 1992 in his PhD thesis
 - Aims to search for an optimal path in a graph, based on the behavior of ants seeking a path between their colony and a source of food.

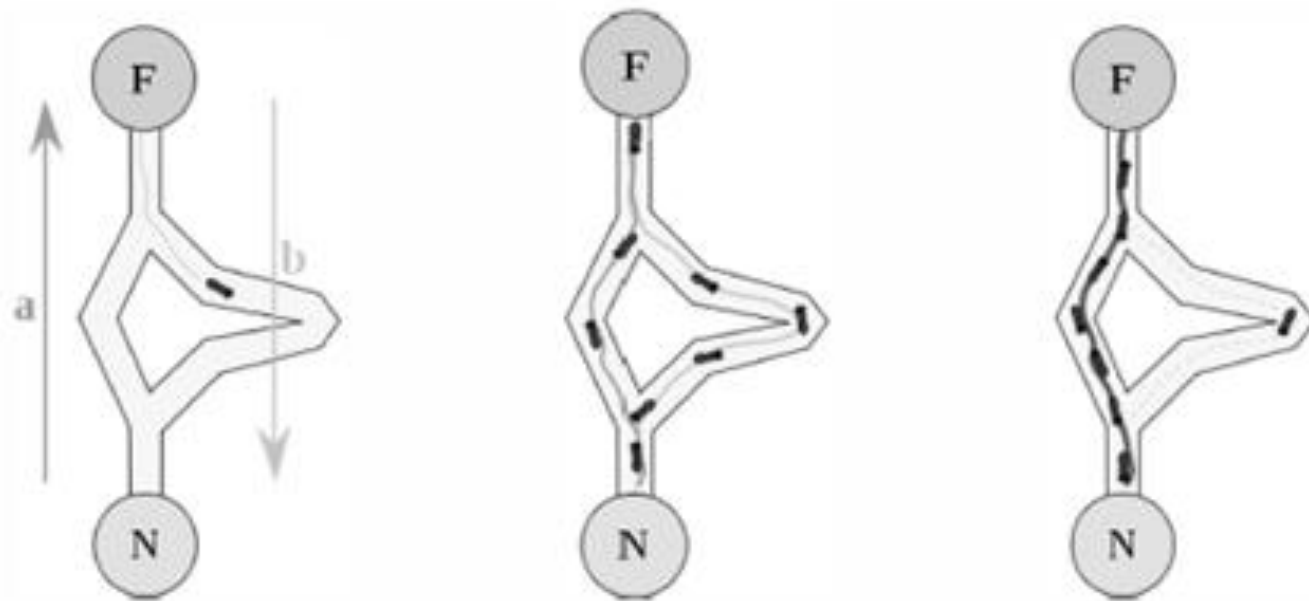
Ant Colony Optimization

Press Esc to exit full screen

Problem name	Authors	Algorithm name	Year
Traveling salesman	Dorigo, Maniezzo & Colorni	AS	1991
	Gamberdella & Dorigo	Ant-Q	1995
	Dorigo & Gamberdella	ACS & ACS 3 opt	1996
	Stutzle & Hoos	MMAS	1997
	Bullnheimer, Hartl & Strauss	AS _{rank}	1997
	Cordon, et al.	BWAS	2000
Quadratic assignment	Maniezzo, Colorni & Dorigo	AS-QAP	1994
	Gamberdella, Taillard & Dorigo	HAS-QAP	1997
	Stutzle & Hoos	MMAS-QAP	1998
	Maniezzo	ANTS-QAP	1999
	Maniezzo & Colorni	AS-QAP	1994
Scheduling problems	Colorni, Dorigo & Maniezzo	AS-JSP	1997
	Stutzle	AS-SMTTP	1999
	Barker et al	ACS-SMTTP	1999
	den Besten, Stutzle & Dorigo	ACS-SMTWTP	2000
	Merkle, Middenderf & Schmeck	ACO-RCPS	1997
Vehicle routing	Bullnheimer, Hartl & Strauss	AS-VRP	1999
	Gamberdella, Taillard & Agazzi	HAS-VRP	1999

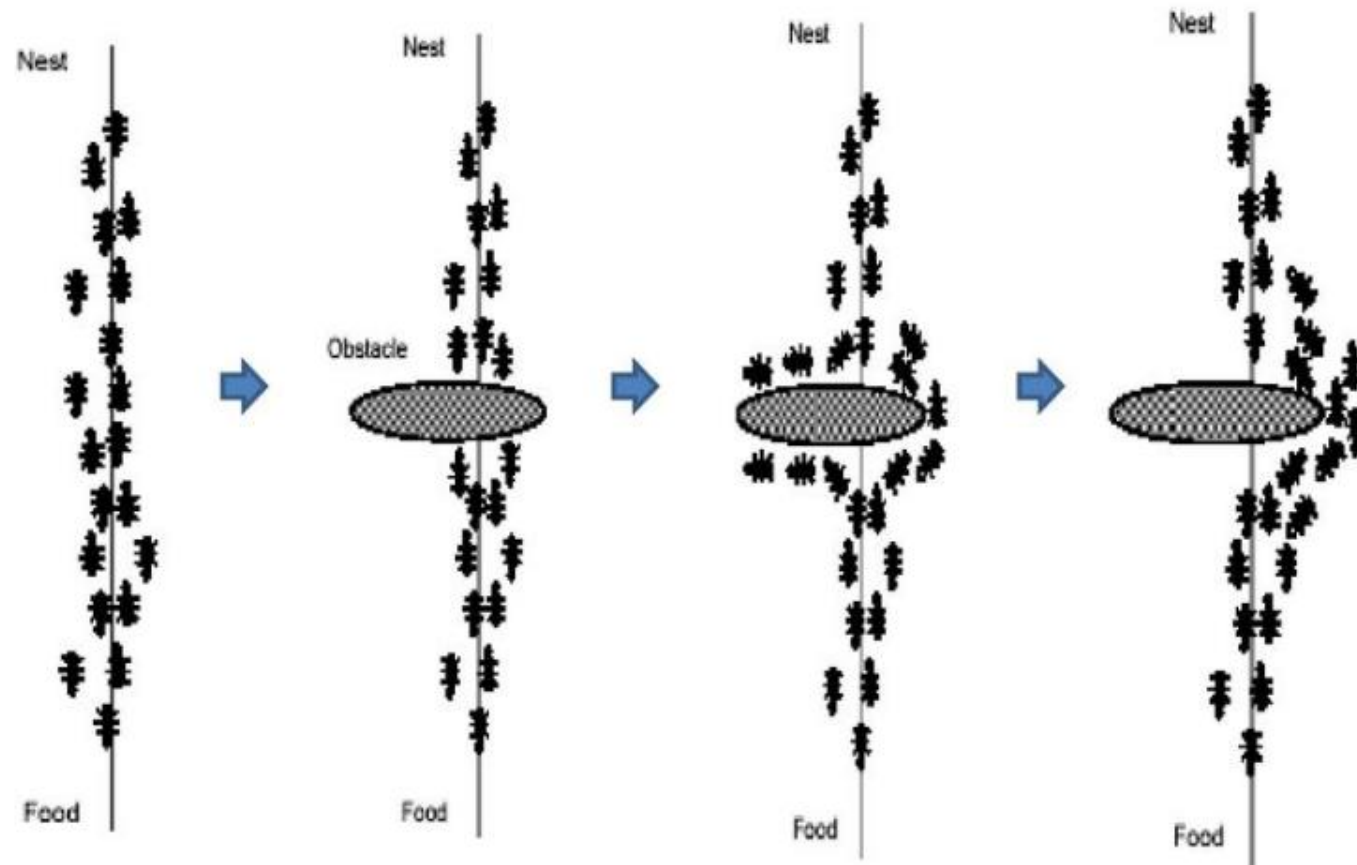
Ant Colony Optimization

Problem name	Authors	Algorithm name	Year
Connection-oriented	Schoonderwood et al.	ABC	1996
network routing	White, Pagurek & Oppacher	ASGA	1998
	Di Caro & Dorigo	AntNet-FS	1998
	Bonabeau et al.	ABC-smart ants	1998
Connection-less	Di Caro & Dorigo	AntNet & AntNet-FA	1997
network routing	Subramanian, Druschel & Chen	Regular ants	1997
	Heusse et al.	CAF	1998
	van der Put & Rethkrantz	ABC-backward	1998
Sequential ordering	Gamberdella & Dorigo	HAS-SOP	1997
Graph coloring	Costa & Hertz	ANTCOL	1997
Shortest common supersequence	Michel & Middendorf	AS_SCS	1998
Frequency assignment	Maniezzo & Carbonaro	ANTS-FAP	1998
Generalized assignment	Ramalhinho Lourenco & Serra	MMAS-GAP	1998
Multiple knapsack	Leguizamón & Michalewicz	AS-MKP	1999
Optical networks routing	Navarro Varela & Sinclair	ACO-VWP	1999
Redundancy allocation	Liang & Smith	ACO-RAP	1999
Constraint satisfaction	Solnon	Ant-P-solver	2000



- ممکن است این حدس زده شود که احتمال برای مورچه ای که مسیر درست را می پیماید برابر با همان احتمالی می باشد که مسیر اشتباه را انتخاب کند.
- نکته در اینجا اینست که مورچه ای که کوتاه ترین مسیر را می پیماید، اولین مورچه ای است که به نقطه مقصد رسیده و سپس به آشیانه (نقطه مبدا حرکت) بر می گردد.
- بنابراین در این کوتاه ترین مسیر فرمون های بیشتری وجود دارد.
- از این رو فرمون دقیقا همان چیزی است که نشان می دهد که مورچه باید از چه مسیری برود و در پایان کوتاه ترین راه، بهترین مسیر می باشد.
- تبخیر شدن فرمون و احتمال-تصادف به مورچه ها امکان پیدا کردن کوتاهترین مسیر را می دهد. این دو ویژگی باعث ایجاد انعطاف در حل هر گونه مسئله بهینه سازی می شوند.

Ant's Foraging Behavior

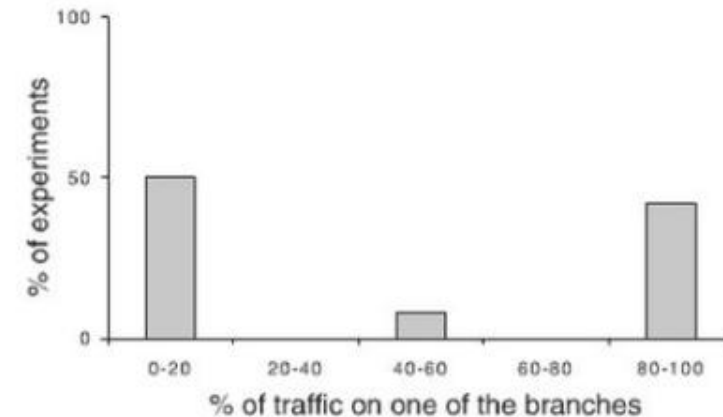
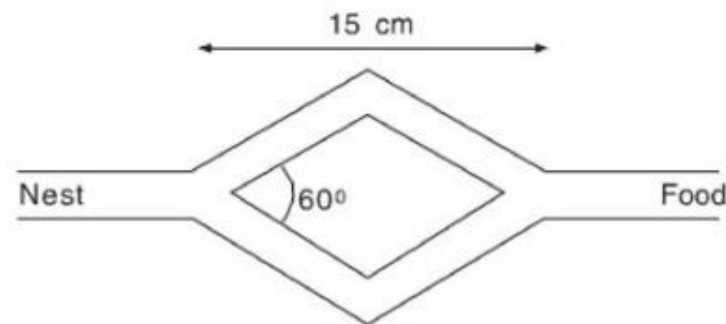


Double Bridge Experiments

- Deneubourg et. al., 1990
 - 1) Branches of equal length
 - $r = l_l / l_s = 1$
 - ants converges to **one** single path

➤ Initial random fluctuation 의 효과

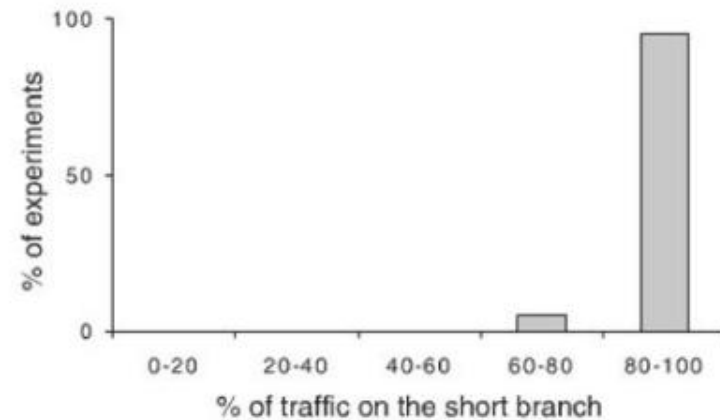
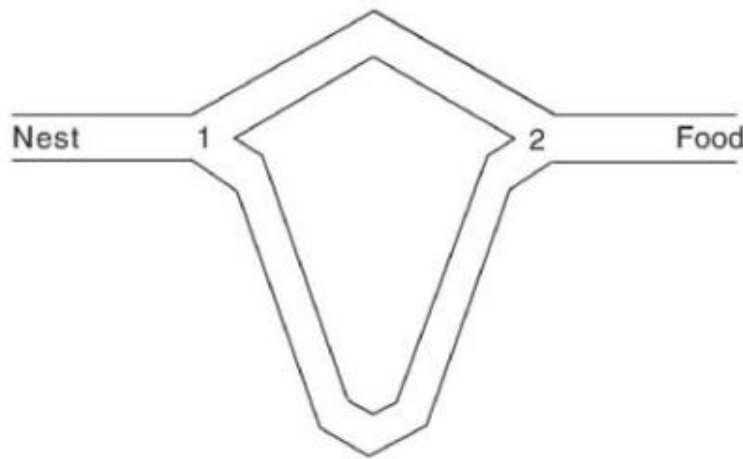
➤ Autocatalytic or **positive feedback** process



Double Bridge Experiments

2) Branches of different length (1)

- $r = l_l / l_s = 2$
- ants converges to the **short** branch
 - Initial random fluctuation 의 효과가 줄어듬
 - Positive feedback to shorter branch
 - Shorter branch -> Higher level of pheromone -> faster accumulation of pheromone on shorter branch ->

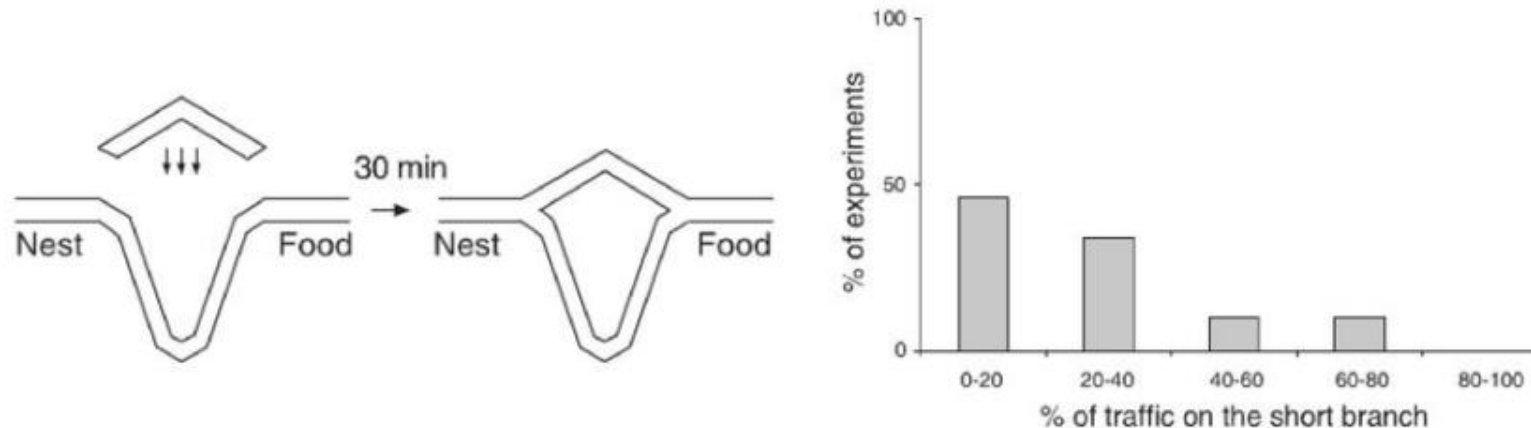


Double Bridge Experiments

3) Branches of different length (2)

- $r = l_l / l_s = 2$
- Short branch were added after 30 min
- the great majority of ants select the **long** branch
 - High pheromone concentration
 - Slow evaporation of pheromone

➤ Autocatalytic (自動觸媒) behavior



Double Bridge Experiments

- Stochastic Model
 - Probability of choosing a branch

$$p_{is}(t) = \frac{(t_s + \varphi_{is}(t))^\alpha}{(t_s + \varphi_{is}(t))^\alpha + (t_l + \varphi_{il}(t))^\alpha}$$

$$p_{is}(t) + p_{il}(t) = 1$$

$p_{ia}(t)$: probability that an ant arriving at decision point $i \in \{0,1\}$ selects branch $a \in \{s,l\}$ (s : short, l : long)

$\varphi_{ia}(t)$: total amount of pheromone at instant t

t_s : traverse time on short branch

$r t_s$: traverse time on long branch

$\alpha = 2$

- Pheromone evaporation

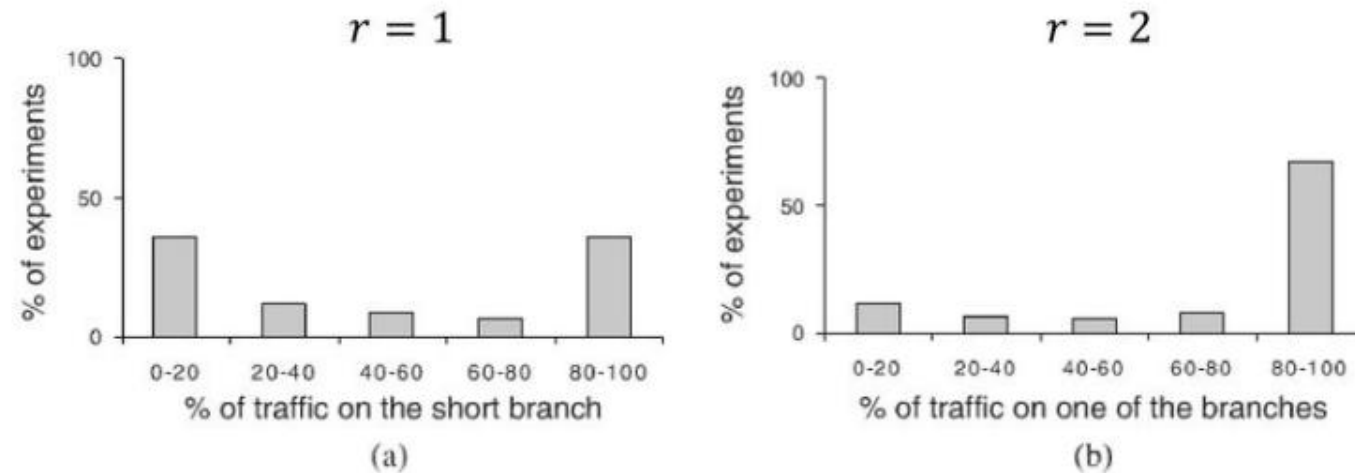
$$d\varphi_{is}/dt = \psi p_{js}(t - t_s) + \psi p_{is}(t), \quad (i = 1, j = 2; i = 2, j = 1),$$

$$d\varphi_{il}/dt = \psi p_{jl}(t - r \cdot t_s) + \psi p_{il}(t), \quad (i = 1, j = 2; i = 2, j = 1)$$

ψ : ants per second cross the bridge in each direction

Double Bridge Experiments

- Monte Carlo simulation



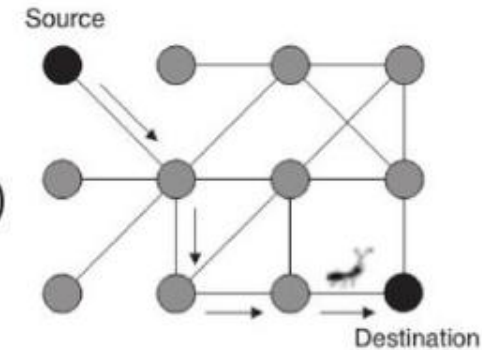
کلونی مورچه ساده (Simple Ant Coloni Optimization(S-ACO)

- هنگامی که مورچه از لانه به سمت غذا حرکت می کند : وضعیت پیش رو
- هنگامی که به لانه برمی گردد : وضعیت پس رو
- مورچه های پیش رو راه حل را با انتخاب احتمالی گره بعدی از میان گره های همسایه در گراف می سازند.
- احتمال انتخاب بر اساس دنباله فرمونی است که توسط مورچه های دیگر قبلا بر روی گراف گذاشته شده است.
- مورچه های پیش رو فرمونی از خود به جا نمیگذارند.
- اختصاص حافظه به هر مورچه اجازه می دهد که مسیر حرکت خود را تا گره مقصد بازیابی کند.
- مورچه ها گره هایی را که در مسیر رفت دیده اند به همراه هزینه یالها در گراف وزن دار را در حافظه نگه می دارند. پس می توانند ارزش راه حلی را که تولید کرده اند را محاسبه کنند.
- آپدیت فرمون بر اساس برآزش مسیر در برگشت انجام می شود. مسیر برگشت همان مسیر رفت است.
- تبخیر فرمون با سرعت ثابت برای همه یالها انجام می شود.
- هر مورچه راه حلی با تصمیم گیری مرحله به مرحله ایجاد می کند.
- در هر گره اطلاعات محلی ذخیره شده در گره و یال خوانده می شود و در انتخاب گره بعدی استفاده می شود

S-ACO Algorithm

Graph $G = (N, A)$, N : set of nodes, A :set of arcs

τ_{ij} : pheromone trail on arc (i, j)
 $= \text{const.} , \forall (i, j) \in A$ (at the beginning stage)



- Ant's path-searching behavior
 - If an ant k is located at a node i , then the probability of choosing j as next node is

$$p_{ij}^k = \begin{cases} \frac{\tau_{ij}^\alpha}{\sum_{l \in N_i^k} \tau_{il}^\alpha}, & \text{if } j \in N_i^k \\ 0, & \text{if } j \notin N_i^k \end{cases}$$

N_i^k : neighborhood of ant k
 when in node i

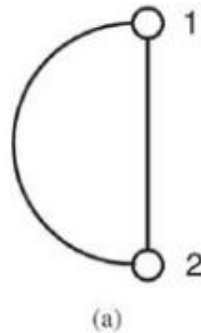
All nodes directly connected to node i
 except for the predecessor of node i

S-ACO Algorithm

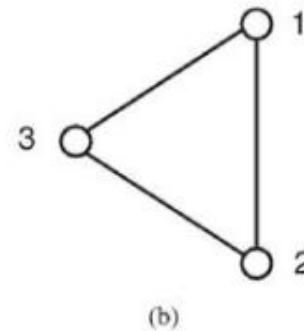
- Pheromone update
 - k -th ant deposits an amount $\Delta\tau^k$ of pheromone on arcs it has visited

$$\tau_{ij} = \tau_{ij} + \Delta\tau^k$$

<Equivalent graph model>



$\Delta\tau^k$: non-increasing function
of path length



$\Delta\tau^k$: constant

S-ACO Algorithm

- Pheromone evaporation
 - After each ant k has moved to a next node, pheromone trails are evaporated by

$$\tau_{ij} \leftarrow (1 - \rho)\tau_{ij} \quad \forall (i, j) \in A \quad \rho \in [0, 1]$$

- Exponentially decreasing

S-ACO

- Shortest Path Algorithm
 - Given a directed acyclic graph $G = (N, A)$
 , where $N = \{0, \dots, G\}$ 0: source (nest), G: destination (food)

```
for all ant  $k$ ,  
    ant position  $n^k \leftarrow 0$   
    ant state  $s^k \leftarrow \text{forward}$   
for all arcs  $(i, j)$ ,  
     $\tau_{ij} \leftarrow \text{const}$   
  
repeat  
    for all ant  $k$ ,  
         $\text{ant\_step}(k)$   
    for all arcs  $(i, j)$ ,  
         $\tau_{ij} \leftarrow (1 - \rho)\tau_{ij}$ 
```

```
 $\text{ant\_step}(k)$   
  
if  $n^k = G$ ,  $s^k \leftarrow \text{backward}$   
if  $n^k = 0$ ,  $s^k \leftarrow \text{forward}$   
  
if  $s^k = \text{forward}$ ,  
    choose next ant position  $n^k \leftarrow j$   
    with the probability of  $p^k_{ij}$   
    update pheromone  $\tau_{ij} = \tau_{ij} + \epsilon$   
  
if  $s^k = \text{backward}$ ,  
    choose next ant position  $n^k \leftarrow j$   
    with the probability of  $p^k_{ji}$   
    update pheromone  $\tau_{ji} = \tau_{ji} + \epsilon$ 
```

در الگوریتم S-ACO، $x^k(t)$ راه حل یافته شده در زمان t می باشد و $f(x^k(t))$ کیفیت راه حل را مشخص می کند.

دوریگو و دیکاریو [۳۱، ۳۵] در آزمایشات خود دریافتند که:

- S-ACO برای گراف های خیلی ساده، بسیار خوب کار می کند و کوتاه ترین مسیر اغلب اوقات انتخاب می شود.

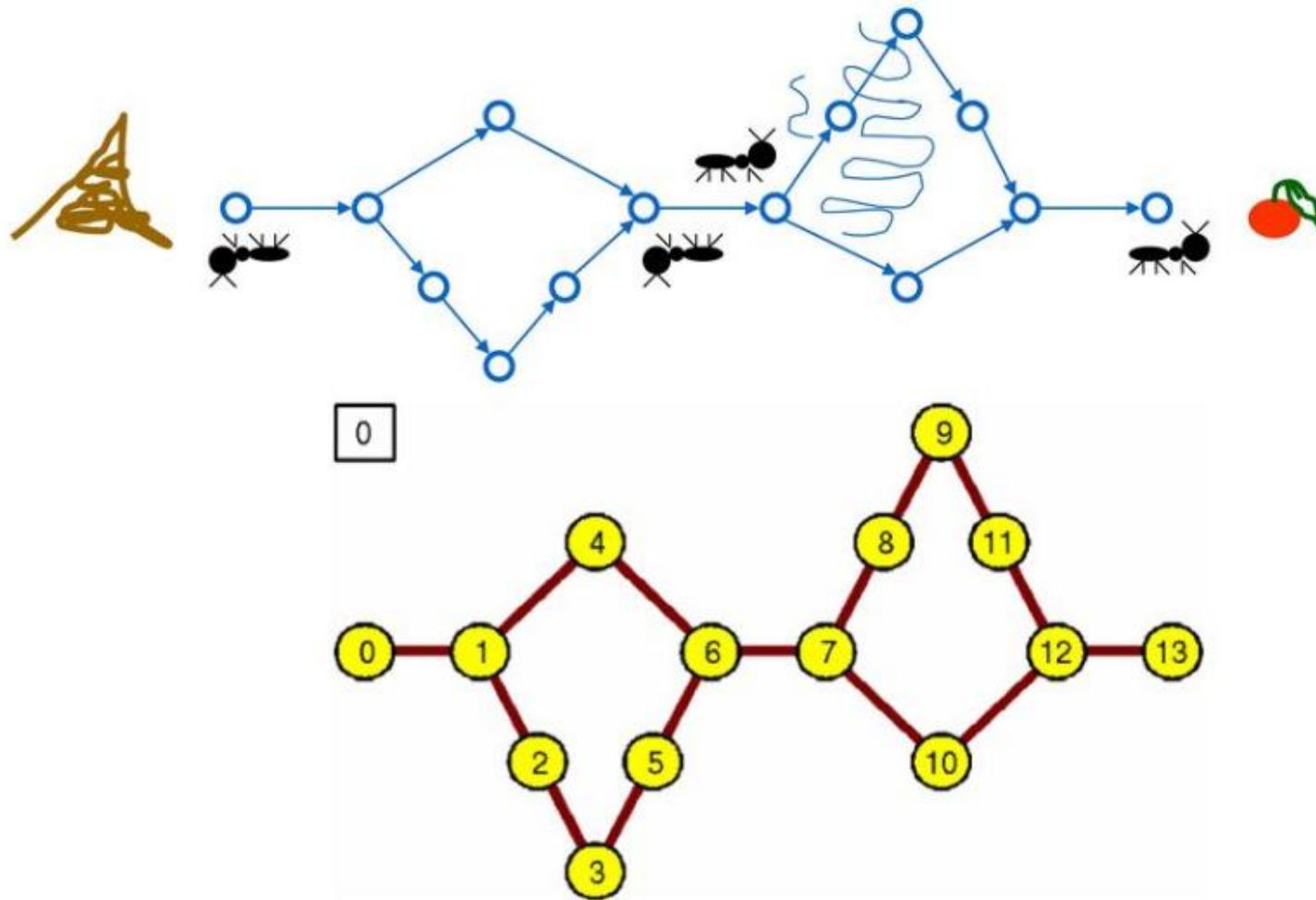
- در گراف های بزرگ تر، کارایی کاهش می یابد. زیرا که الگوریتم نسبت به انتخاب پارامترها حساس تر می شود.

- تعداد زیاد مورچه ها منجر به رفتار ناهمگرایی در الگوریتم می شود.

- تبخیر برای گراف های پیچیده اهمیت بیشتری دارد. اگر $\rho = 0$ باشد، یعنی تبخیری انجام نگرفته است و همگرایی رخ نمی دهد. اگر تبخیر زیاد باشد (استفاده از ρ بزرگ)، معمولاً الگوریتم برای مسائل پیچیده به راه حل های بهینه محلی همگرا می شود.

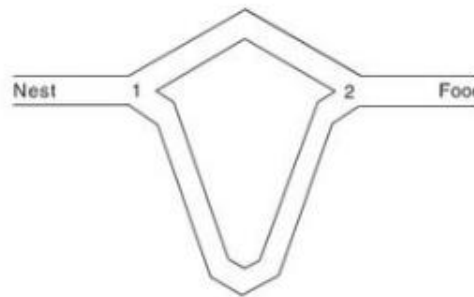
- برای مسائل پیچیده، مقدار بزرگ α (اهمیت فرومون) معمولاً منجر به عدم همگرایی الگوریتم می شود. برای این مسائل، مقادیر کوچک تر α مناسب تر است.

S-ACO



Experiments with Double Bridge

- Number of ants
 $m = 1, \dots, 512$



- Pheromone update
 $\Delta\tau^k = \text{constant}$ (without considering path length)
 $= 1/L^k$ (with considering path length)

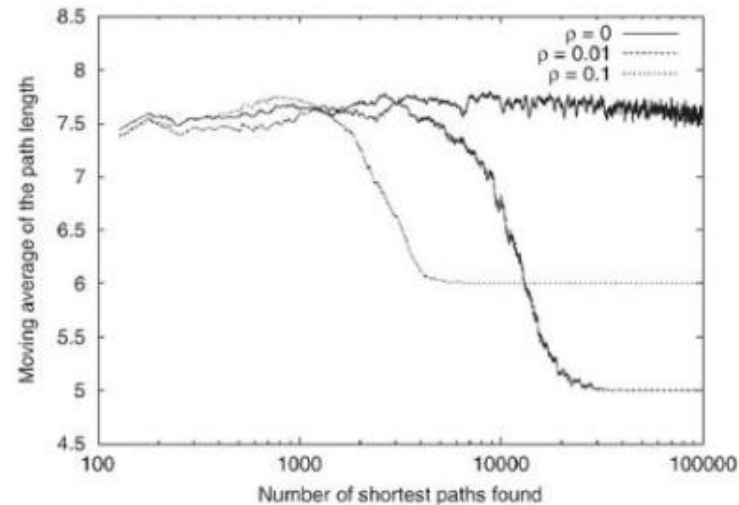
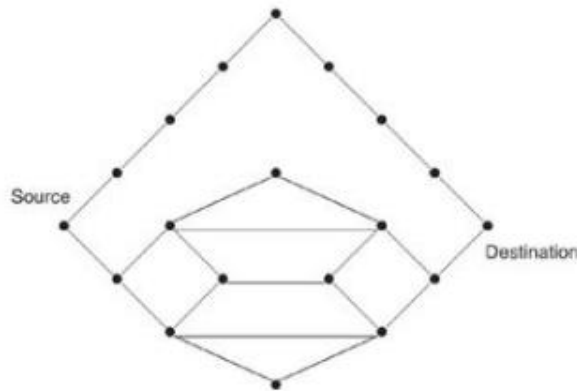
Table 1.1

Percentage of trials in which S-ACO converged to the long path (100 independent trials for varying values of m , with $\alpha = 2$ and $\rho = 0$)

m	1	2	4	8	16	32	64	128	256	512
without path length	50	42	26	29	24	18	3	2	1	0
with path length	18	14	8	0	0	0	0	0	0	0

Experiments with Extended Double Bridge

- Pheromone evaporation
 - Evaporation rate $\rho = \{0, 0.01, 0.1\}$ $\tau_{ij} \leftarrow (1 - \rho)\tau_{ij} \quad \forall (i, j) \in A$
 - $\rho = 0$: no evaporation
 - $\rho = 0.01$: 10% of pheromone evaporated after 10 iterations
 - $\rho = 0.1$: 65% of pheromone evaporated after 10 iterations
 - The larger ρ , the faster converges to suboptimal solution



سیستم مورچه

- دو فاز اصلی شامل ساخت راه حل و به هنگام سازی است.
- روش خوب مقداردهی اولیه دنباله فرومون دادن مقداری بیشتر از مقدار فرومون مورد انتظاری است که در یک تکرار توسط مورچه گذاشته می شود.
- اگر فرومون اولیه کم باشد خیلی زود جستجو براساس تورهای اولیه تولید شده مورچه ها انجام می شود که به دلیل تصادفی بودن آنها الگوریتم نتایج خوبی در بر نخواهد داشت.
- اگر این مقدار خیلی زیاد باشد خیلی از تکرارها تا زمانی که فرومون به اندازه کافی تبخیر شود. بیهوده خواهد بود.

$$p_{ij}^k = \frac{[\tau_{ij}]^\alpha [\eta_{ij}]^\beta}{\sum_{l \in N_i^k} [\tau_{il}]^\alpha [\eta_{il}]^\beta} \quad \text{if } j \in N_i^k$$

$$\tau_{ij} \leftarrow (1 - \rho) \tau_{ij}, \quad \forall (i, j) \in L$$

$$\tau_{ij} \leftarrow \tau_{ij} + \sum_{k=1}^m \Delta \tau_{ij}^k \quad \forall (i, j) \in L$$

$$\Delta \tau_{ij}^k = \begin{cases} 1/C^k & \text{if arc } (i, j) \text{ belongs to } T^k \\ 0 & \text{otherwise} \end{cases}$$

مرحله ۱: مقداردهی اولیه به پارامترهای الگوریتم.

مرحله ۲: تکرار مرحله ۳ برای همه n_k مورچه.

مرحله ۳: ساخت مسیر $x^k(t)$ برای مورچه k ام با توجه به رابطه (۴-۴).

مرحله ۴: تکرار مراحل ۵ و ۶ برای همه n_k مورچه.

مرحله ۵: اعمال فرآیند تبخیر فرومون برای همه یال‌های (i, j) در گراف مسیرهای ممکن با توجه به رابطه (۴-۵).

مرحله ۶: افزایش مقدار فرومون هر یال (i, j) در مسیر ساخته شده توسط مورچه k ام با توجه به رابطه (۴-۶). مقدار $\Delta\tau^k$ را در این رابطه با توجه به رابطه (۴-۷) محاسبه می‌شود.

مرحله ۷: تکرار مراحل ۲ الی ۶ تا ارضاء شرط خروج.

سیستم مورچه نخبه

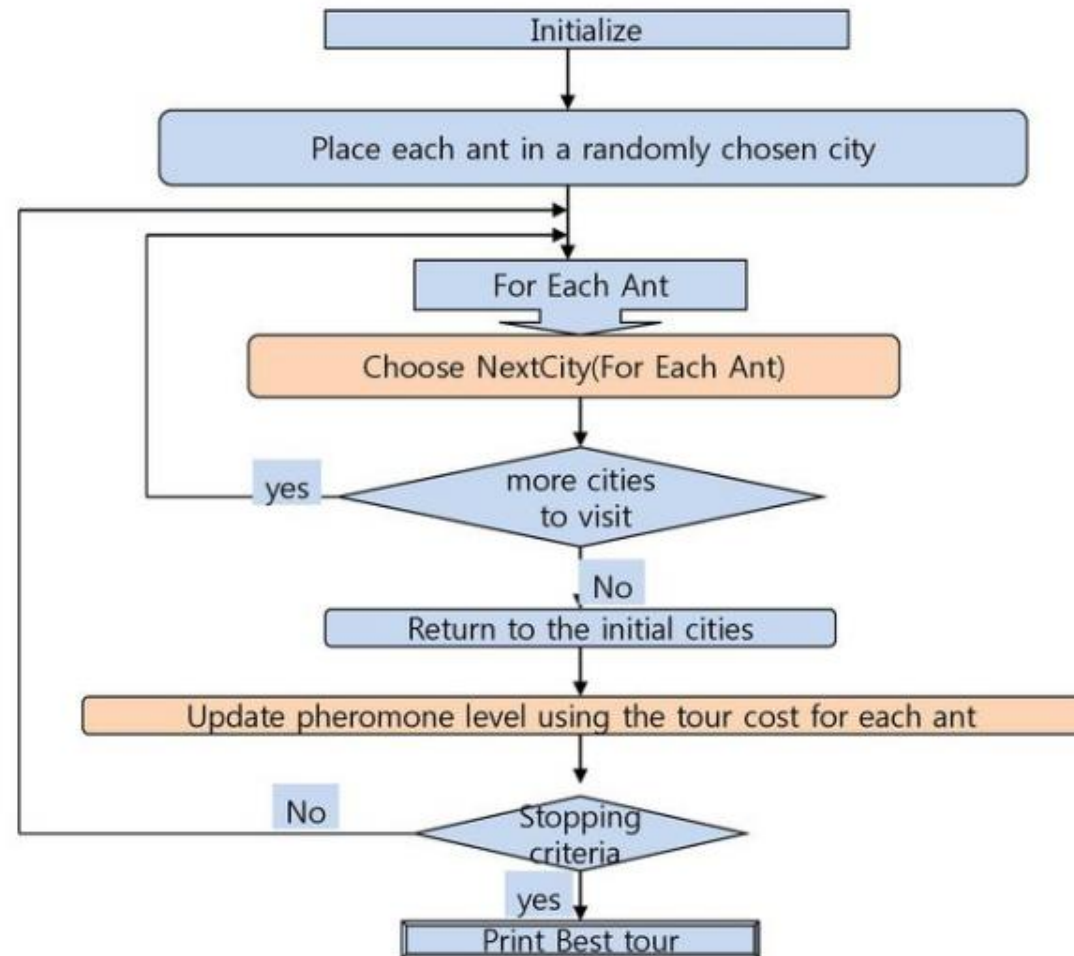
$$\tau_{ij} \leftarrow \tau_{ij} + \sum_{k=1}^m \Delta\tau_{ij}^k + e\Delta\tau_{ij}^{bs} \quad (۸-۴)$$

که $\Delta\tau_{ij}^k$ همانند رابطه (۷-۴) تعریف می شود و $\Delta\tau_{ij}^{bs}$ بر اساس رابطه زیر تعریف می شود:

$$\Delta\tau_{ij}^{bs} = \begin{cases} 1/C^{bs} & \text{if arc } (i,j) \text{ belongs to } T^{bs} \\ 0 & \text{otherwise} \end{cases} \quad (۹-۴)$$

Ant System (AS) for TSP

- Algorithm



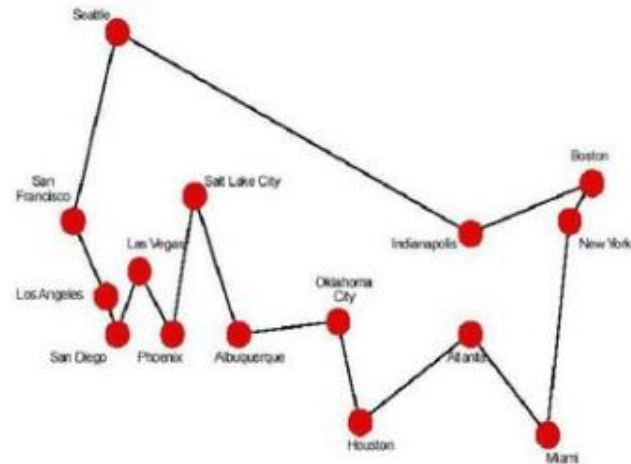
Traveling Salesman Problem

- TSP PROBLEM

Given N cities, and a distance function d between cities, find a tour that:

1. Goes through every city once and only once
2. Minimizes the total distance.

- Problem is NP-hard
- Classical combinatorial optimization problem to test.



Ant System (AS) for TSP

- Tour Construction
 - Probability of ant k , currently at city i , choosing a next city j

$$p_{ij}^k(t) = \begin{cases} \frac{[\tau_{ij}(t)]^\alpha [\eta_{ij}]^\beta}{\sum_{l \in N_i^k} [\tau_{il}(t)]^\alpha [\eta_{il}]^\beta} & \text{if } j \in N_i^k \\ 0 & \text{if } j \notin N_i^k \end{cases}$$

N_i^k : feasible neighborhood of ant k when being at city i
set of cities that ant k has not visited yet

$\eta_{ij} = 1/d_{ij}$: heuristic value priori to closer city

d_{ij} : length of arc (i, j)

α, β : parameters

$\alpha = 0, \beta > 1$: the closest city is selected

$\alpha > 1, \beta = 0$: only pheromone is used

$\Rightarrow$ stagnation

$\Rightarrow$ strongly suboptimal solution

Ant System (AS) for TSP

- Pheromone trail update

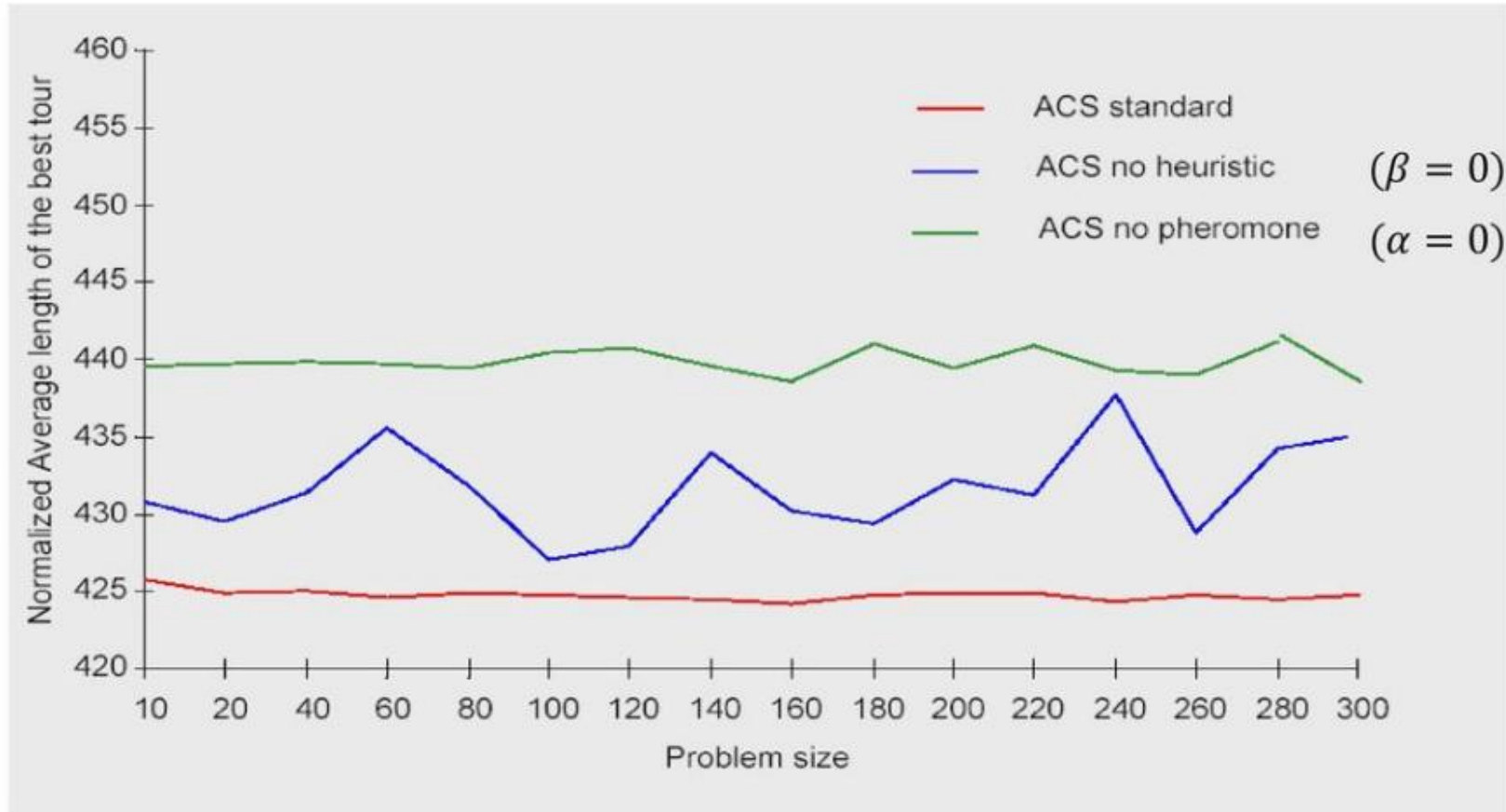
$$\tau_{ij} \leftarrow \tau_{ij} + \sum_{k=1}^m \Delta\tau_{ij}^k \quad , \forall (i,j) \in A$$
$$\Delta\tau_{ij}^k = \begin{cases} 1/C^k, & \text{if } (i,j) \in T^k \\ 0, & \text{otherwise} \end{cases}$$

T^k : tour of ant k , tabu

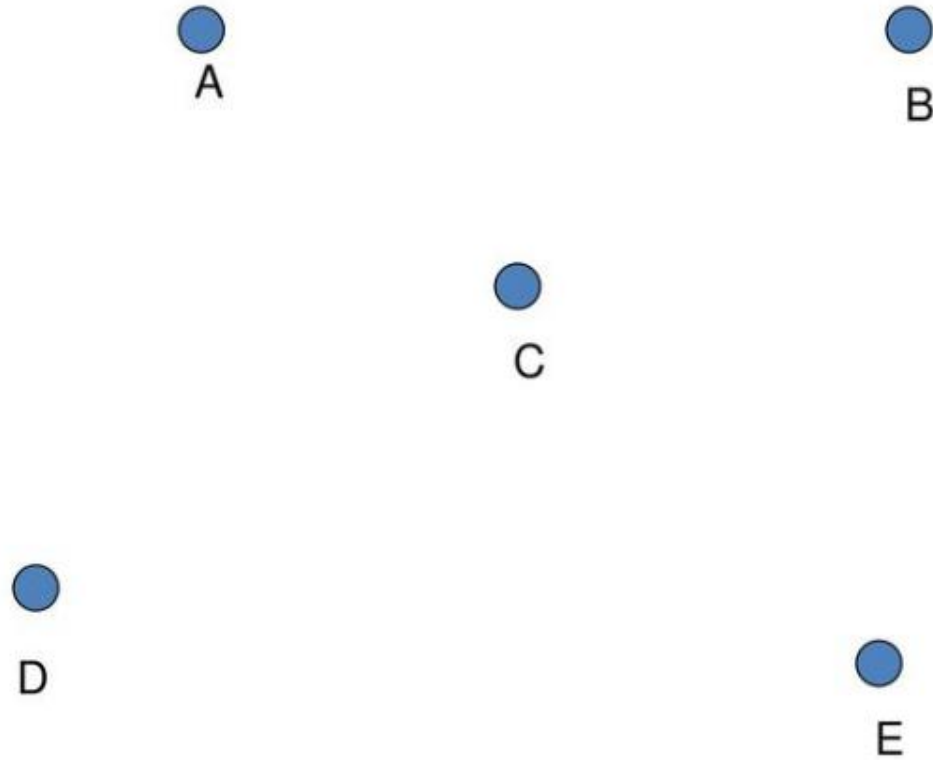
- Pheromone evaporation

$$\tau_{ij} \leftarrow (1 - \rho)\tau_{ij} \quad \forall (i,j) \in A$$

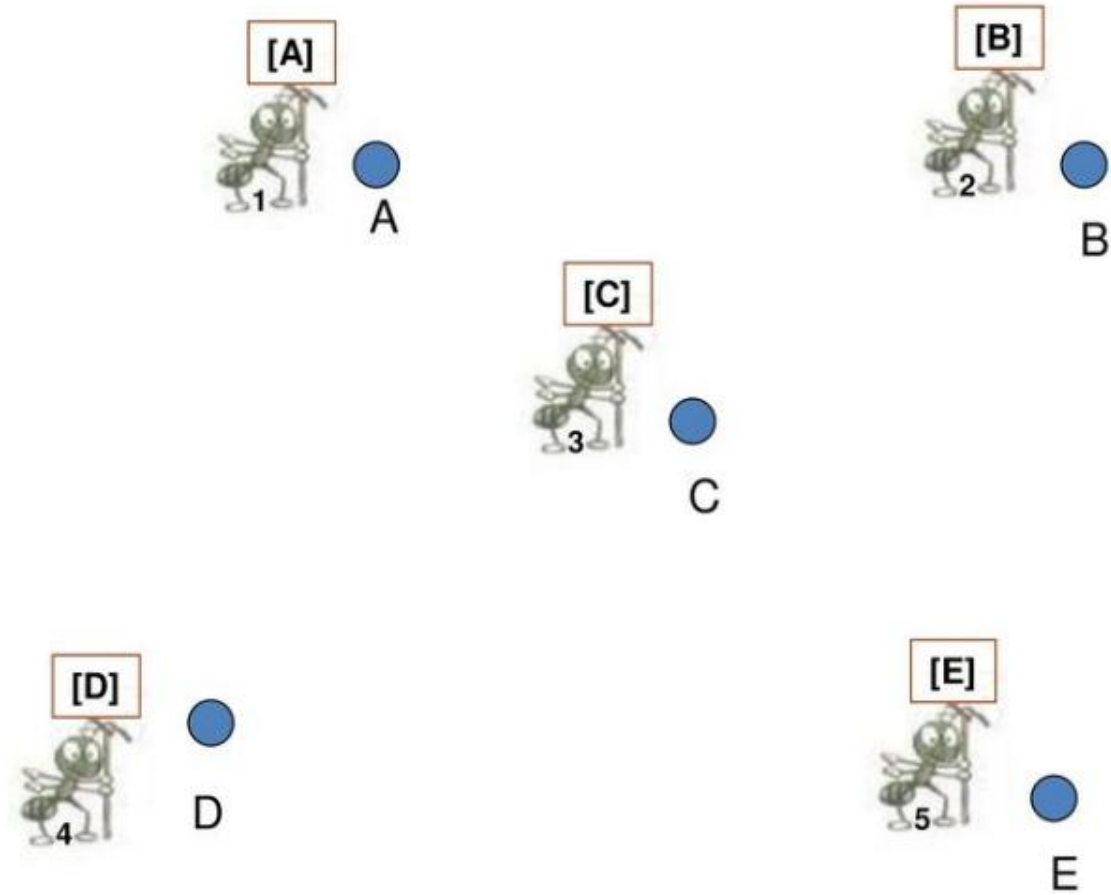
Press Esc to exit full screen



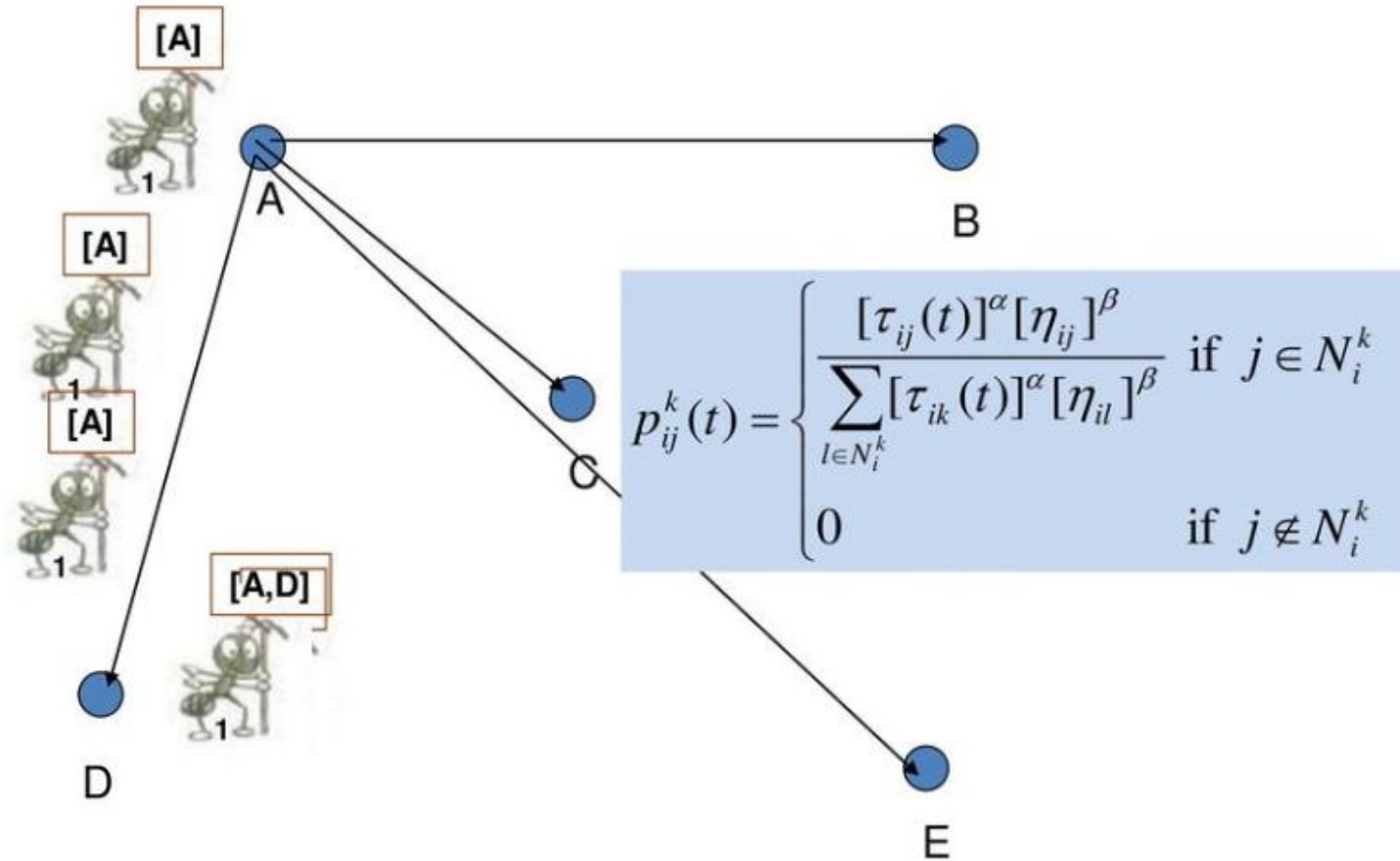
A simple TSP example



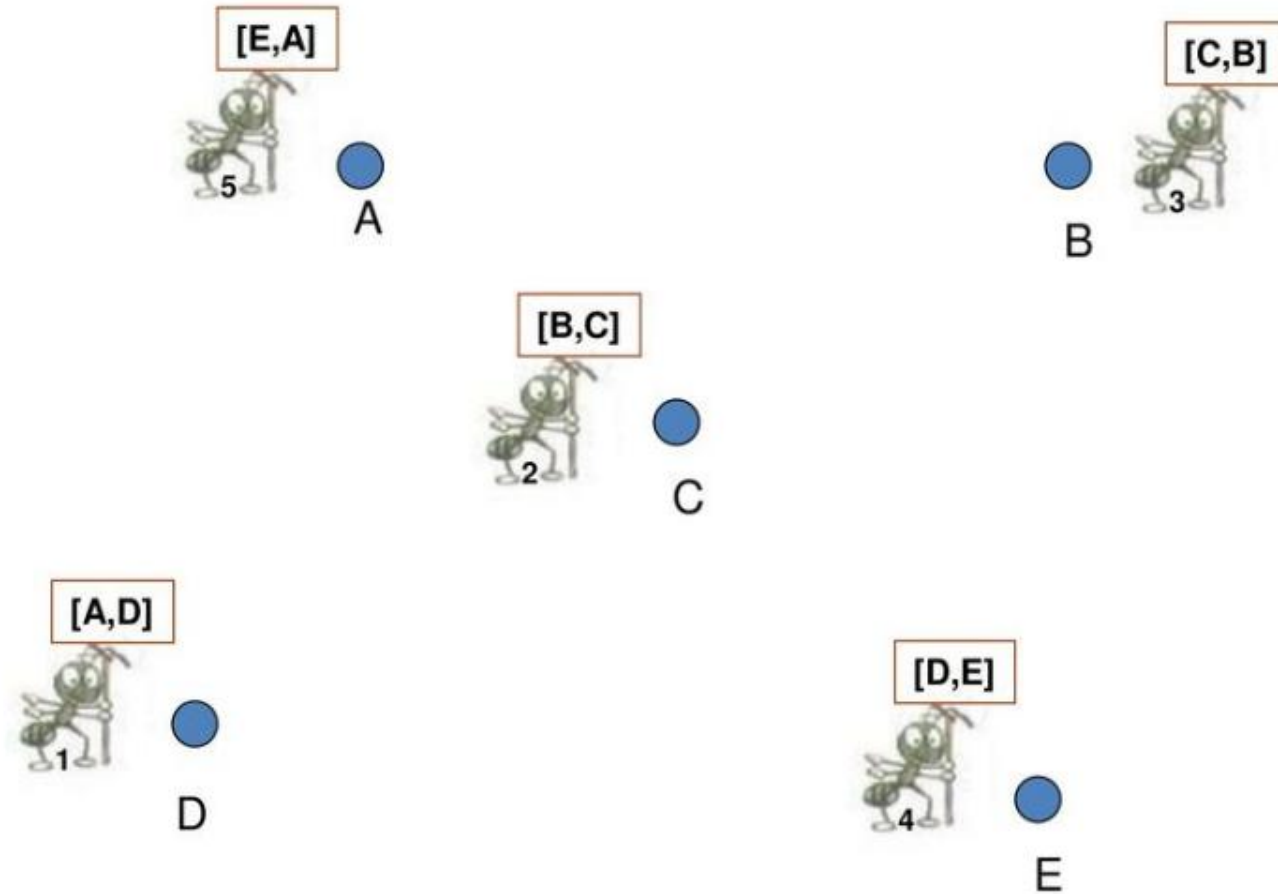
Iteration 1



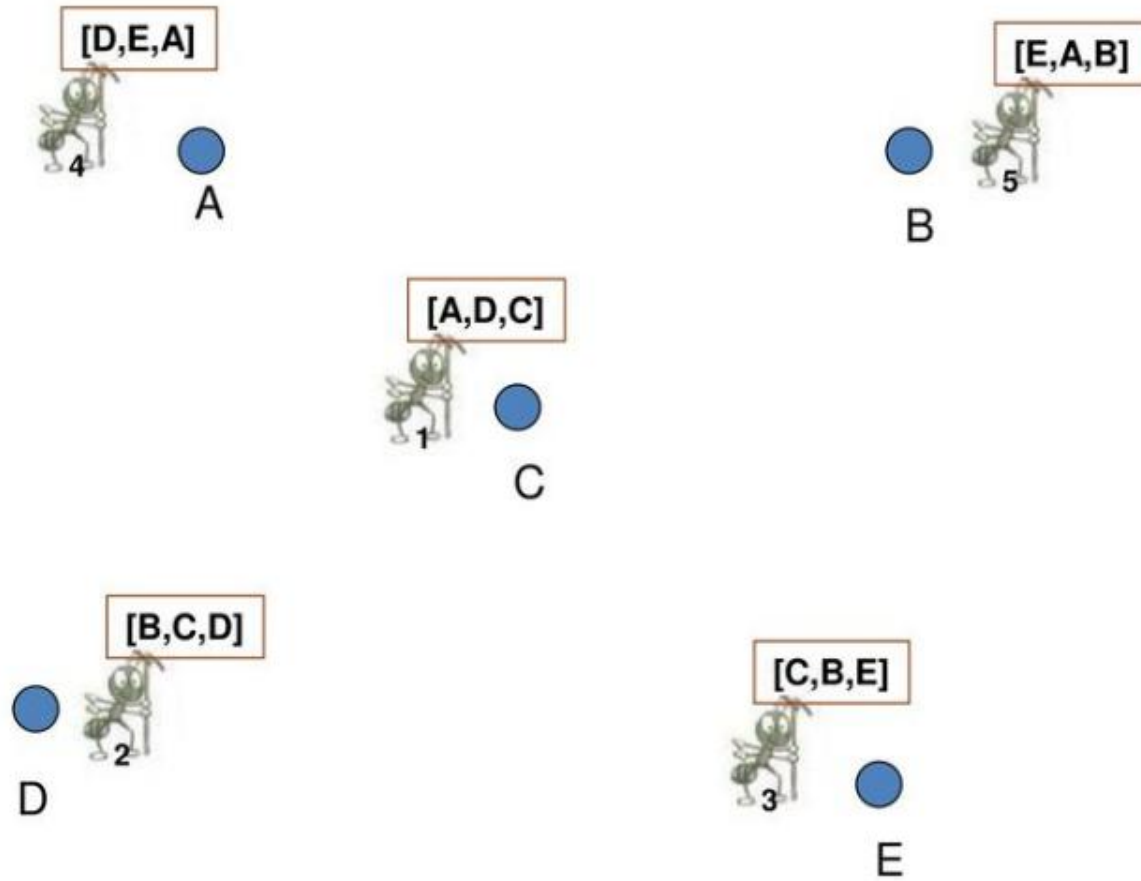
How to build next sub-solution?



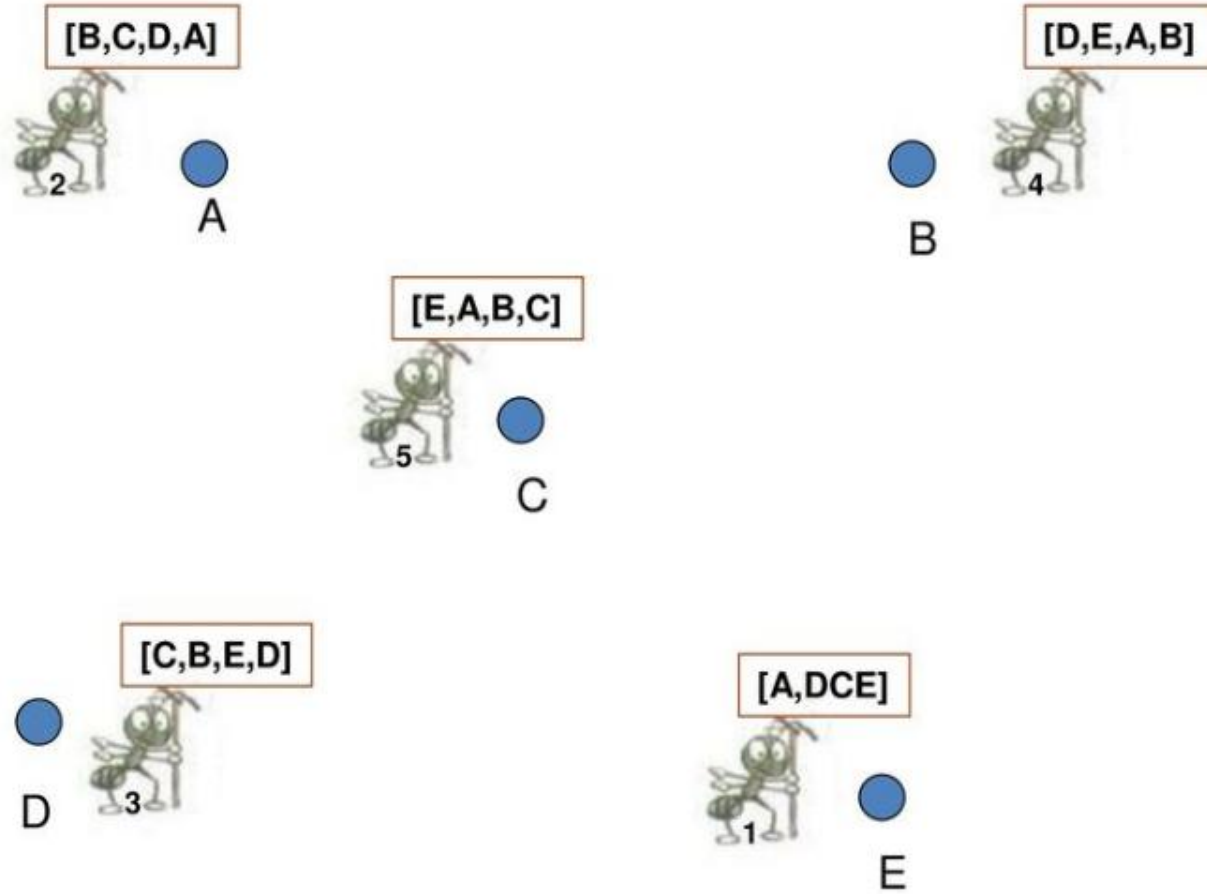
Iteration 2



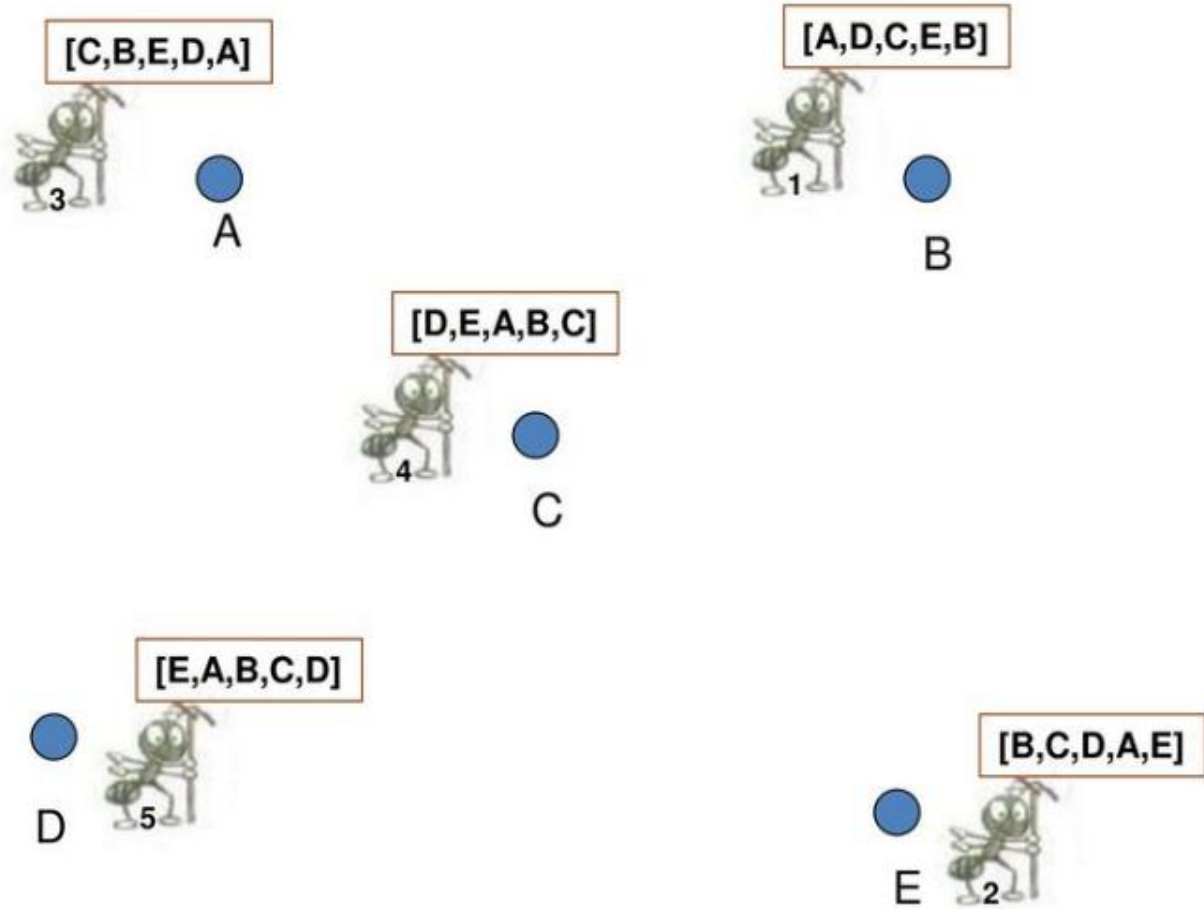
Iteration 3



Iteration 4



Iteration 5



Pheromone Update

 1 [A,D,C,E,B] $C^1 = 300$

 2 [B,C,D,A,E] $C^2 = 450$

 3 [C,B,E,D,A] $C^3 = 260$

 4 [D,E,A,B,C] $C^4 = 280$

 5 [E,A,B,C,D] $C^5 = 420$

$$\Delta\tau_{i,j}^k = \begin{cases} \frac{1}{C^k} & \text{if } (i, j) \in \text{tour} \\ 0 & \text{otherwise} \end{cases}$$

$$\tau_{A,B} \leftarrow \tau_{A,B} + \Delta\tau_{A,B}^1 + \Delta\tau_{A,B}^2 + \Delta\tau_{A,B}^3 + \Delta\tau_{A,B}^4 + \Delta\tau_{A,B}^5$$

سیستم کلونی مورچه

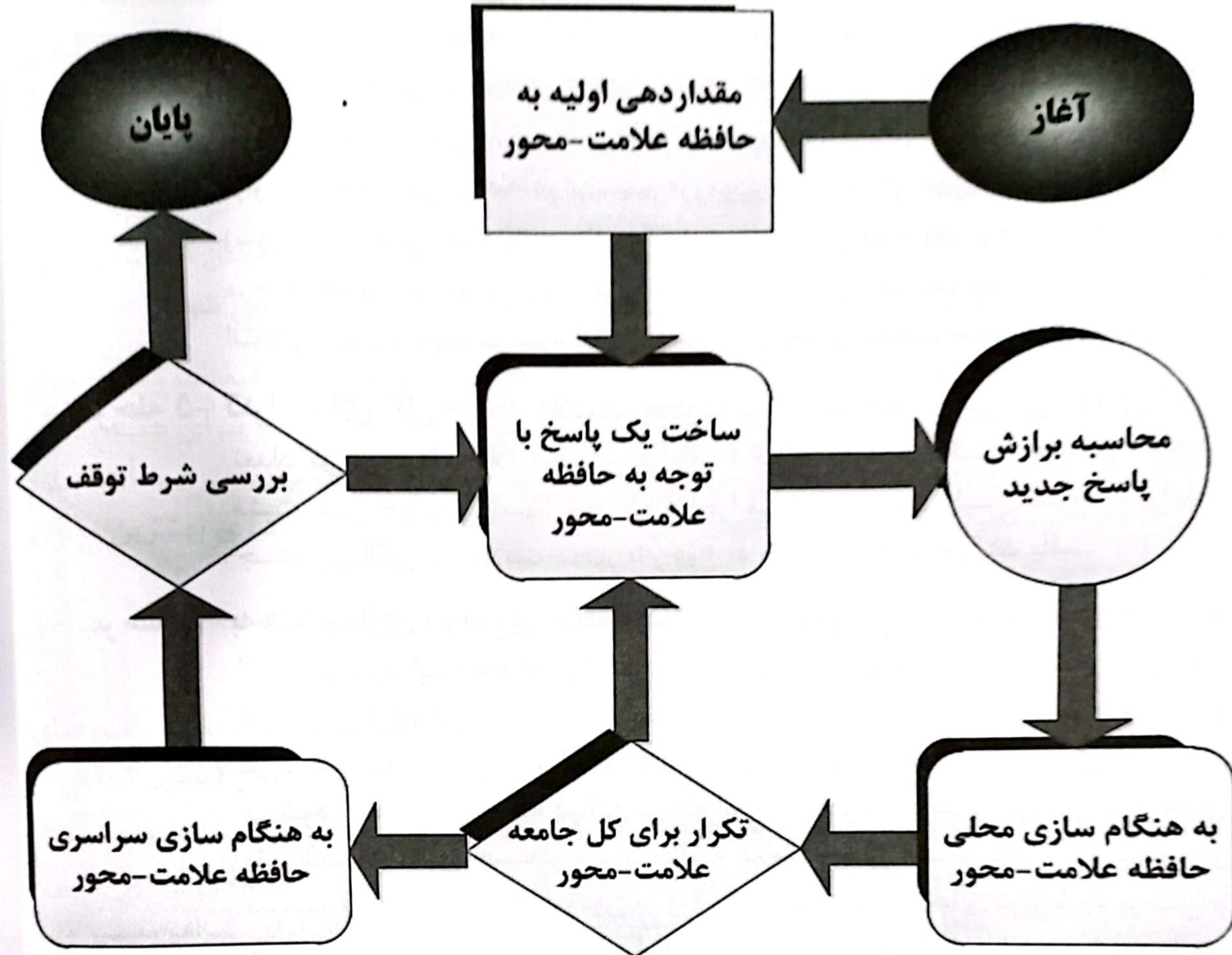
- بیشتر به تجربیات جستجوی خود متکی است.
- تبخیر و فرومون گذاری فقط بر روی یالهایی که متعلق به بهترین تورهای پیدا شده تاکنون هستند انجام می شود.
- هر زمانی که یک مورچه از یک یال برای حرکت استفاده می کند مقداری فرومون را از روی یال مزبور به منظور افزایش قابلیت پویش الگوریتم (با افزایش احتمال پیمایش راه های دیگر) بر میدارد.

سیستم کلونی مورچه

$$j = \begin{cases} \operatorname{argmax}_{l \in N_i^k} \{ \tau_{il} [\eta_{il}]^\beta \} & \text{if } q \leq q_0 \\ J & \text{otherwise} \end{cases}$$

$$\tau_{ij} \leftarrow (1 - \rho) \tau_{ij} + \Delta \tau_{ij}^{bs} \quad \forall (i, j) \in T^{bs}$$

$$\tau_{ij} \leftarrow (1 - \xi) \tau_{ij} + \xi \tau_0$$



حل مسئله فروشنده دوره گرد

- مقادیر اولیه فرومون را در هر مسیر به یک مقدار تصادفی مثبت و کوچک قرار داده می شود.
- یک قانون انتقال ساده برای انتخاب شهر بعدی برای بازدید به صورت زیر است :

$$\Phi_{ij,k}(t) = \begin{cases} \frac{T_{ij}(t)^\alpha}{\sum_{c \in C_{i,k}} T_{ic}(t)^\alpha} & \text{if } j \in C_{i,k} \\ 0 & \text{if } j \notin C_{i,k} \end{cases}$$

که $T_{ij}(t)$ شدت فرومون در لبه (i,j) بین شهرهای i,j را نشان می دهد مورچه k ام با k مشخص شده و α مقدار ثابت و $C_{i,k}$ مجموعه ای از شهرهایی است که مورچه k مجبور به ملاقات آن شهرها هست.

Ant Colony System (ACS) for TSP

- **Dorigo & Gambardella (1997)** introduced four modifications in AS :
 - 1) Different tour construction rule
 - 2) Different pheromone trail updates
 - Global
 - Local

ACS for TSP

- Tour Construction
 - When located at city i , ant k chooses a next city j by the following rule

$$j = \begin{cases} \arg \max_{l \in N_i^k} \{\tau_{ij} [\eta_{il}]^\beta\}, & \text{if } q \leq q_o \\ J, & \text{otherwise} \end{cases}$$

q : random variable uniformly distributed in $[0,1]$

$q_o \in (0,1]$: parameter

J : random variable selected by the following probability ($\alpha = 0$)

$$p_{ij}^k(t) = \begin{cases} \frac{[\tau_{ij}(t)]^\alpha [\eta_{ij}]^\beta}{\sum_{l \in N_i^k} [\tau_{il}(t)]^\alpha [\eta_{il}]^\beta} & \text{if } j \in N_i^k \\ 0 & \text{if } j \notin N_i^k \end{cases}$$

ACS for TSP

- Tour Construction (Example)

$$\tau_{A,B} = 150 \quad \eta_{A,B} = 1/10$$

$$\tau_{A,C} = 35 \quad \eta_{A,C} = 1/7$$

$$\tau_{A,D} = 90 \quad \eta_{A,D} = 1/15$$

If $q \leq q_0$

choose A-B (15)

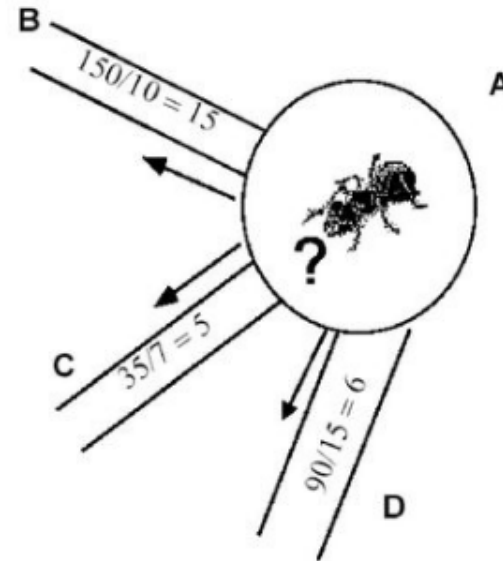
else

choose

A-B with probability 15/26

A-C with probability 5/26

A-D with probability 6/26



ACS for TSP

- Global pheromone trail update
 - Add and evaporate pheromone only to the best tour since trial began

$$\tau_{ij} \leftarrow (1 - \rho)\tau_{ij} + \rho \Delta\tau^{best}_{ij} \quad \forall (i,j) \in T^{best}$$

$$\Delta\tau^{best}_{ij} = 1/c^{best}$$

- Computational complexity
 $O(n^2) \rightarrow O(n)$

ACS for TSP

- Local pheromone update
 - Remove pheromone to the arc (i,j) immediately after having been crossed during tour construction

$$\tau_{ij} \leftarrow (1 - \xi)\tau_{ij} + \xi \tau_o$$

$\xi \in (0,1)$, τ_o : initial pheromone

- Tabu effect
 - The arc recently visited is not desirable for the following ants
 - Increase the exploration of arcs that have not been visited yet $\Rightarrow$ reduce stagnation behavior

ACS for TSP

- Results for a 30 cities instance

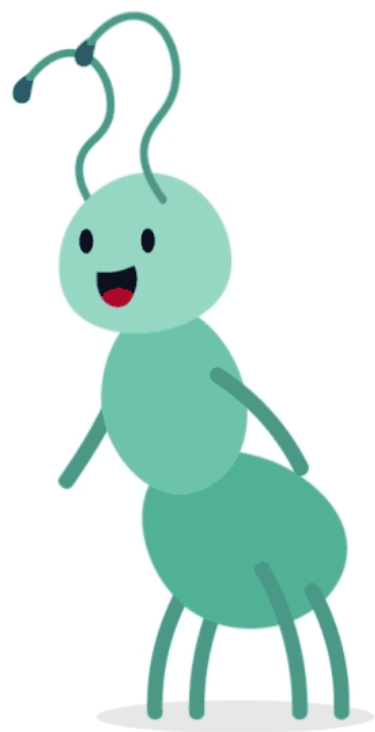
	best	average	std.deviation
ACS	420	420.4	1.3
Tabu-search	420	420.6	1.5
Sim. Annealing	422	459.8	25.1

- Results for larger cities (best tour)

	ACO	Gen.Alg.	Evol.Prog.	Sim.Ann.
50 cities	425	428	426	443
75 cities	535	545	542	580
100 cities	21282	21761		

Conclusions

- ACO is a recently proposed metaheuristic approach for solving hard combinatorial optimization problems.
- Artificial ants implement a randomized construction heuristic which makes probabilistic decisions.
- ACO can find best solutions on smaller problems
- On larger problems converged to good solutions –but not the best
- In ACO Local search is extremely important to obtain good results.



الگوریتم تقلید محور

- موجودات ذره نامیده می شوند. هر ذره یک راه حل است.
- در فضای چند بعدی جستجو حرکت می کنند.
- تغییر در موقعیت ذرات در فضای جستجو مبتنی بر رفتار اجتماعی موجودات در تقلید از موفقیت سایر موجودات است.
- تغییرات یک ذره تحت تاثیر تجربیات خود و یا دانش همسایگانش است.
- برای محیط های پیوسته مناسب است.
- بردار سرعت علاوه بر هدایت فرآیند بهینه سازی دانش تجربی ذره و اطلاعات تغییرات ذرات همسایه را بازتاب می کند.
- دانش تجربی ذره : مولفه شناختی متناسب است با فاصله ذره از بهترین موقعیت خودش از ابتدای جستجو
- اطلاعات تغییرات اجتماعی: مولفه اجتماعی

PSO بهینه سراسری

$$x_i(t+1) = x_i(t) + v_i(t+1) \quad (۴-۲۶)$$

به صورتی که $x_i(0) \sim U(x_{min}, x_{max})$.

$$v_{ij}(t+1) = v_{ij}(t) + c_1 r_{1j}(t)[y_{ij}(t) - x_{ij}(t)] + c_2 r_{2j}(t)[\hat{y}_j(t) - x_{ij}(t)]$$

$$y_i(t+1) = \begin{cases} y_i(t) & \text{if } f(x_i(t+1)) \geq f(y_i(t)) \\ x_i(t+1) & \text{if } f(x_i(t+1)) < f(y_i(t)) \end{cases} \quad (28-4)$$

به صورتی که $f: \mathbb{R}^{n_x} \rightarrow \mathbb{R}$ تابع برازش می‌باشد. همانند الگوریتم‌های تکاملی، تابع برازش معیاری برای اندازه‌گیری میزان نزدیکی راه‌حل مورد نظر به بهینه سراسری می‌باشد، یعنی تابع برازش کارایی و کیفیت یک ذره (یا راه‌حل) را می‌سنجد.

بهترین موقعیت سراسری، $\hat{y}(t)$ ، در گام زمانی t ، به صورت زیر تعریف می‌شود:

$$\hat{y}(t) \in \{y_0(t), \dots, y_{n_s}(t)\} | f(\hat{y}(t)) = \min\{f(y_0(t)), \dots, f(y_{n_s}(t))\} \quad (29-4)$$

که n_s تعداد کل ذرات در ازدحام می‌باشد. لازم به ذکر است که در تعریف رابطه (29-4)، $\hat{y}$ ، بهترین موقعیت کشف شده تا کنون، توسط هر ذره، می‌باشد. همچنین بهترین موقعیت سراسری، می‌تواند از ذرات ازدحام کنونی انتخاب شود [55]. در این صورت خواهیم داشت:

$$\hat{y}(t) = \min\{f(y_0(t)), \dots, f(y_{n_s}(t))\} \quad (30-4)$$

مراحل $gbest$ PSO به صورت ذیل است:

مرحله ۱: مقداردهی اولیه به پارامترهای سرعت و مکان ذرات در ازدحام اولیه.

مرحله ۲: تکرار مرحله ۳ برای همه ذرات.

مرحله ۳: به‌هنگام‌سازی بهترین موقعیت محلی ذره i ام.

مرحله ۴: به‌هنگام‌سازی بهترین موقعیت سراسری همه ذرات.

مرحله ۵: تکرار مراحل ۶ و ۷ برای همه ذرات.

مرحله ۶: محاسبه سرعت جدید ذره i ام با توجه به رابطه ۴-۲۷.

مرحله ۷: محاسبه مکان جدید ذره i ام با توجه به رابطه ۴-۲۶.

مرحله ۸: تکرار مراحل ۲ الی ۷ تا ارضاء شرط خروج.

PSO بهینه محلی

$$v_{ij}(t+1) = v_{ij}(t) + c_1 r_{1j}(t)[y_{ij}(t) - x_{ij}(t)] + c_2 r_{2j}(t)[\hat{y}_{ij}(t) - x_{ij}(t)] \quad (31-4)$$

که $\hat{y}_{ij}(t)$ بهترین موقعیت یافته شده توسط همسایگان ذره i در دامنه j می باشد. بهترین موقعیت محلی ذره، $\hat{y}_i$ ، یعنی بهترین موقعیت پیدا شده در همسایگی N_i ، به صورت زیر تعریف می شود:

$$\hat{y}_i(t+1) \in \{N_i | f(\hat{y}_i(t+1)) = \min \{f(x)\}, \forall x \in N_i\} \quad (32-4)$$

که همسایگان آن به صورت زیر تعریف می شود:

$$N_i = \{y_{i-n_{N_i}}(t), y_{i-n_{N_i}+1}(t), \dots, y_{i-1}(t), y_i(t), y_{i+1}(t), \dots, y_{i+n_{N_i}}(t)\} \quad (33-4)$$

مرحله ۱: مقداردهی اولیه به پارامترهای سرعت و مکان ذرات در ازدحام اولیه.

مرحله ۲: تکرار مراحل ۳ و ۴ برای همه ذرات.

مرحله ۳: به‌هنگام‌سازی بهترین موقعیت محلی ذره i ام.

مرحله ۴: به‌هنگام‌سازی بهترین موقعیت سراسری در همسایگی ذره i ام.

مرحله ۵: تکرار مراحل ۶ و ۷ برای همه ذرات.

مرحله ۶: محاسبه سرعت جدید ذره i ام با توجه به رابطه ۴-۳۱.

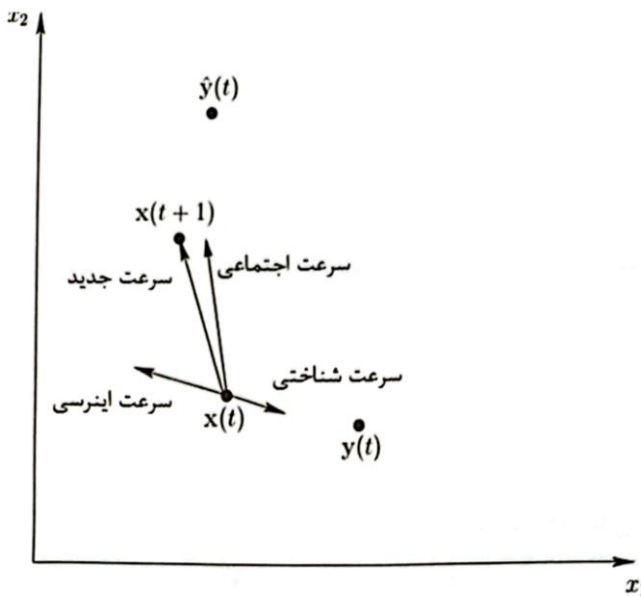
مرحله ۷: محاسبه مکان جدید ذره i ام با توجه به رابطه ۴-۲۶.

مرحله ۸: تکرار مراحل ۲ الی ۷ تا ارضاء شرط خروج.

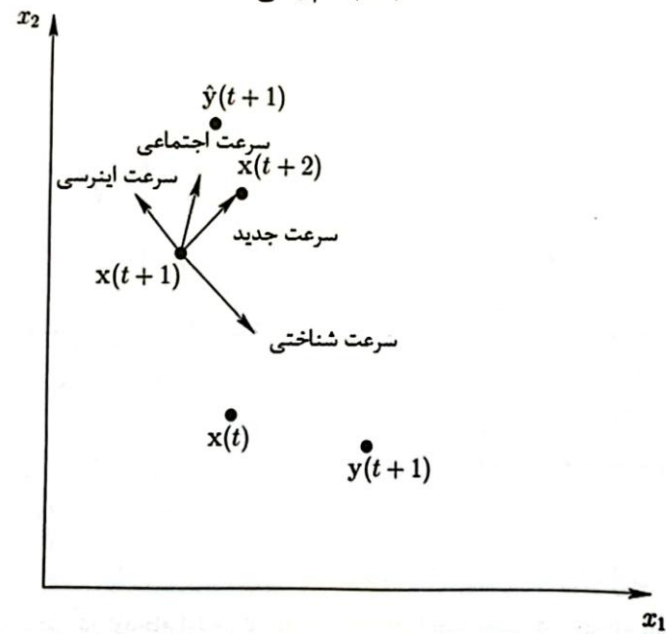
مقایسه دو روش

همان طور که در روابط (۴-۲۷) و (۴-۳۱) نشان داده شده است، محاسبه سرعت شامل سه بخش می باشد:

- **سرعت قبلی**، $v_i(t)$ ، مانند حافظه ای است که جهت حرکت قبلی را در خود نگه می دارد. این مقدار از تغییر مسیر ناگهانی ذره جلوگیری می کند و آن را متمایل به مسیر کنونی آن می نماید. این مؤلفه، مؤلفه اینرسی نیز نامیده می شود.
- **مؤلفه شناختی**، $c_1 r_1 (y_i - x_i)$ ، کارایی مؤلفه i را نسبت به کارایی گذشته اش می سنجد. مؤلفه شناختی همان حافظه موقعیتی موجودات می باشد که برای ذره، بهترین موقعیت یا مکان تجربه شده اش را حفظ می کند. تأثیر این مؤلفه، متمایل شدن حرکت ذره به سمت بهترین موقعیت خودش می باشد. کندی و ابرهارت، مؤلفه شناختی را تمایل حسرت وار^۱ ذره به گذشته اش می نامند [۶۷].
- **مؤلفه اجتماعی**، $c_2 r_2 (\hat{y} - x_i)$ ، برای $gbest$ PSO، یا $c_2 r_2 (\hat{y}_i - x_i)$ برای $lbest$ PSO، کارایی مؤلفه i ، را نسبت به گروهی از ذرات یا همسایگان می سنجد. تأثیر مؤلفه اجتماعی، حرکت هر ذره به سمت بهترین موقعیت پیدا شده تا کنون توسط همسایگان ذره می باشد.

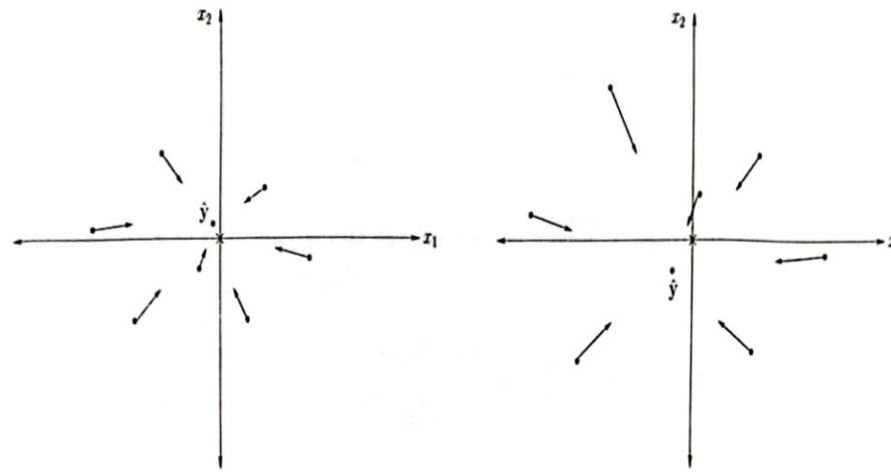


(الف) گام زمانی t



(ب) گام زمانی $t+1$

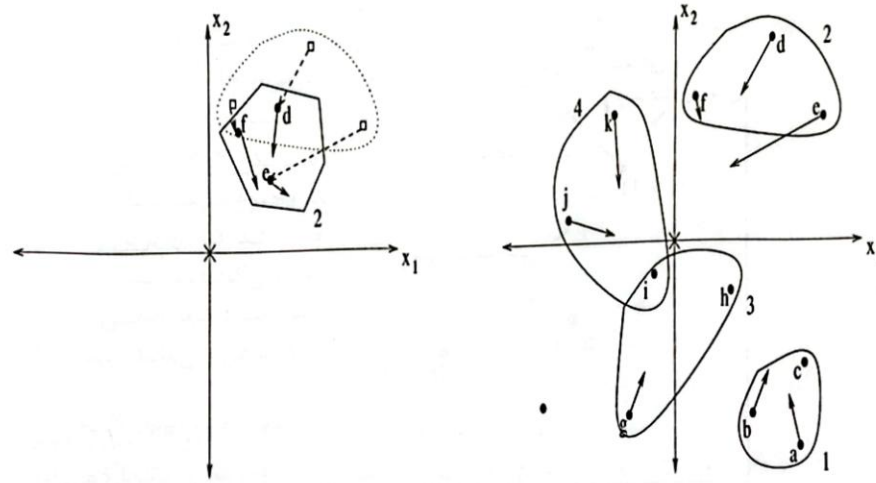
شکل ۳-۴ نمایش هندسی به هنگام سازی سرعت و موقعیت برای یک ذره دوبعدی



(ب) در گام زمانی $t+\Delta t$

(الف) در گام زمانی t

شکل ۴-۴ نمایش چند ذره‌ای $gbest$ PSO



(ب) بهینه محلی در ازدحام ثانویه (گام $t+\Delta t$)

(الف) بهینه محلی در ازدحام اولیه (گام t)

شکل ۵-۴ نمایش چند ذره‌ای $lbest$ PSO