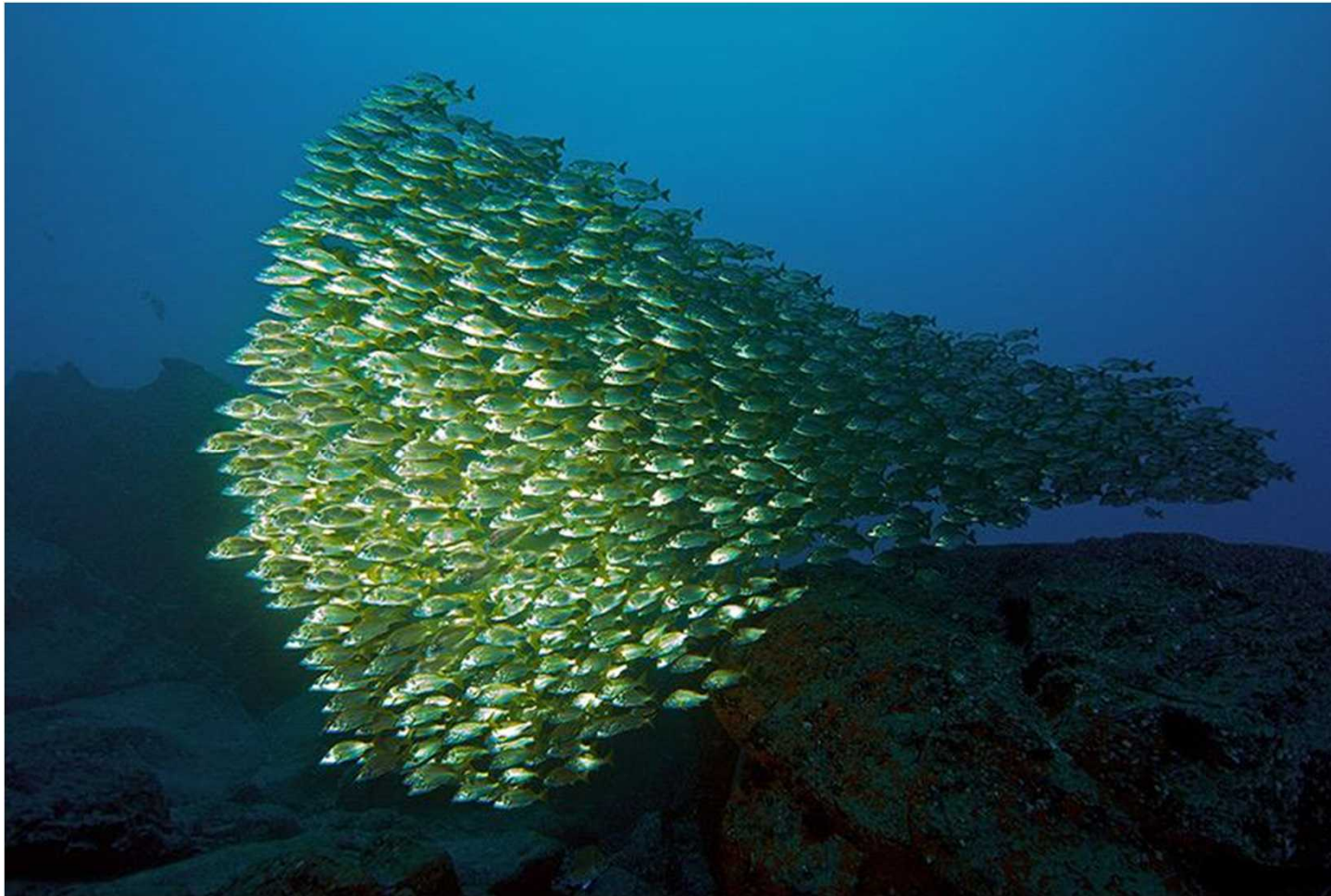


# Swarm Intelligence



<http://pixdaus.com/single.php?id=168307>

# Swarm Intelligence

- „Intelligence hejna“
- algoritmy inspirované chováním skupin ptáků, hmyzu, ryb apod.
  
- Particle Swarm Optimization
- Ant Colony Optimization
- Glowworm Swarm Optimization
- ...
  
- rozdílné chování v předávání informací

# Swarm Intelligence

## Výhody:

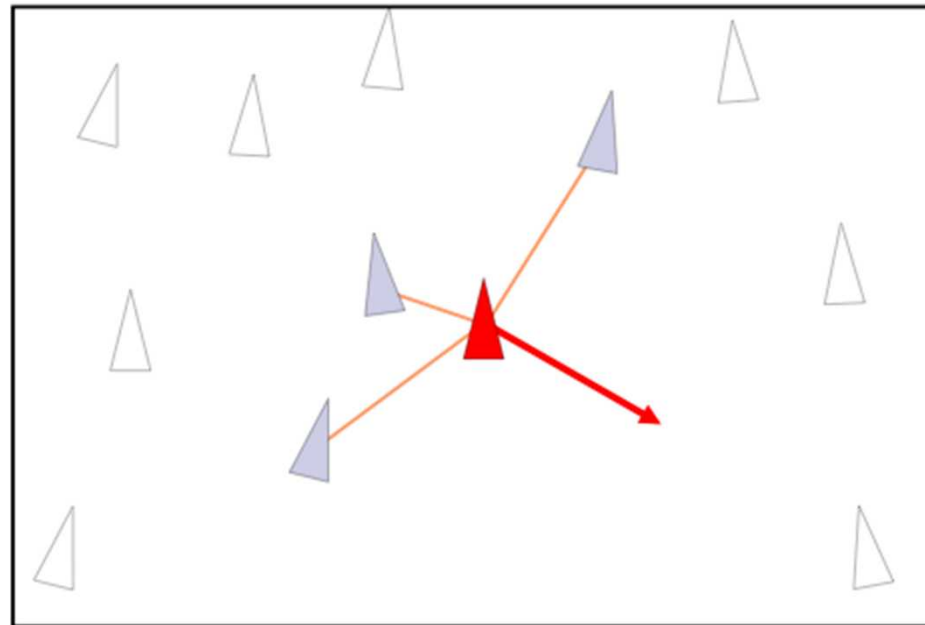
- snadná implementace
- nepotřebuje gradienty ani Hessovu matici

## Nevýhody:

- potřeba správně určit konstanty a parametry pro danou úlohu

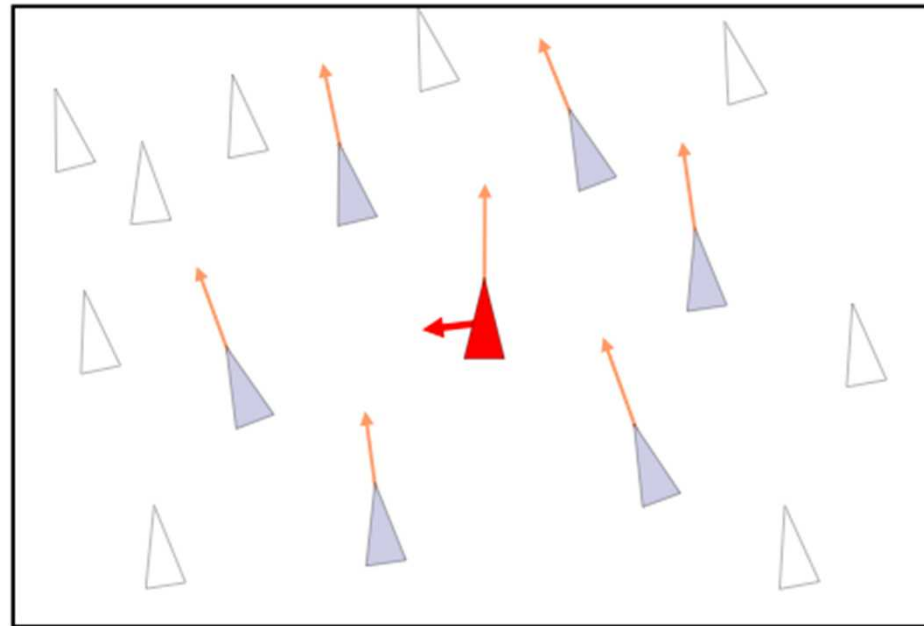
# Pravidla Swarm Intelligence

## Pravidlo 1: Vyhnutí se kolizi



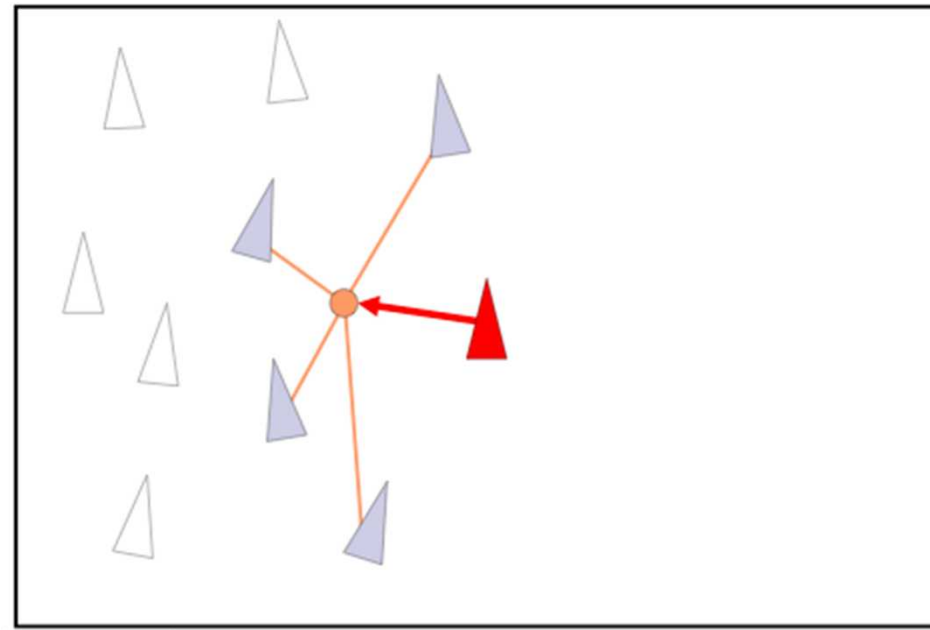
# Pravidla Swarm Intelligence

## Pravidlo 2: Sladění rychlostí okolních ptáků



# Pravidla Swarm Intelligence

Pravidlo 3: Zůstat blízko okolních členů hejna



# Particle Swarm Optimization



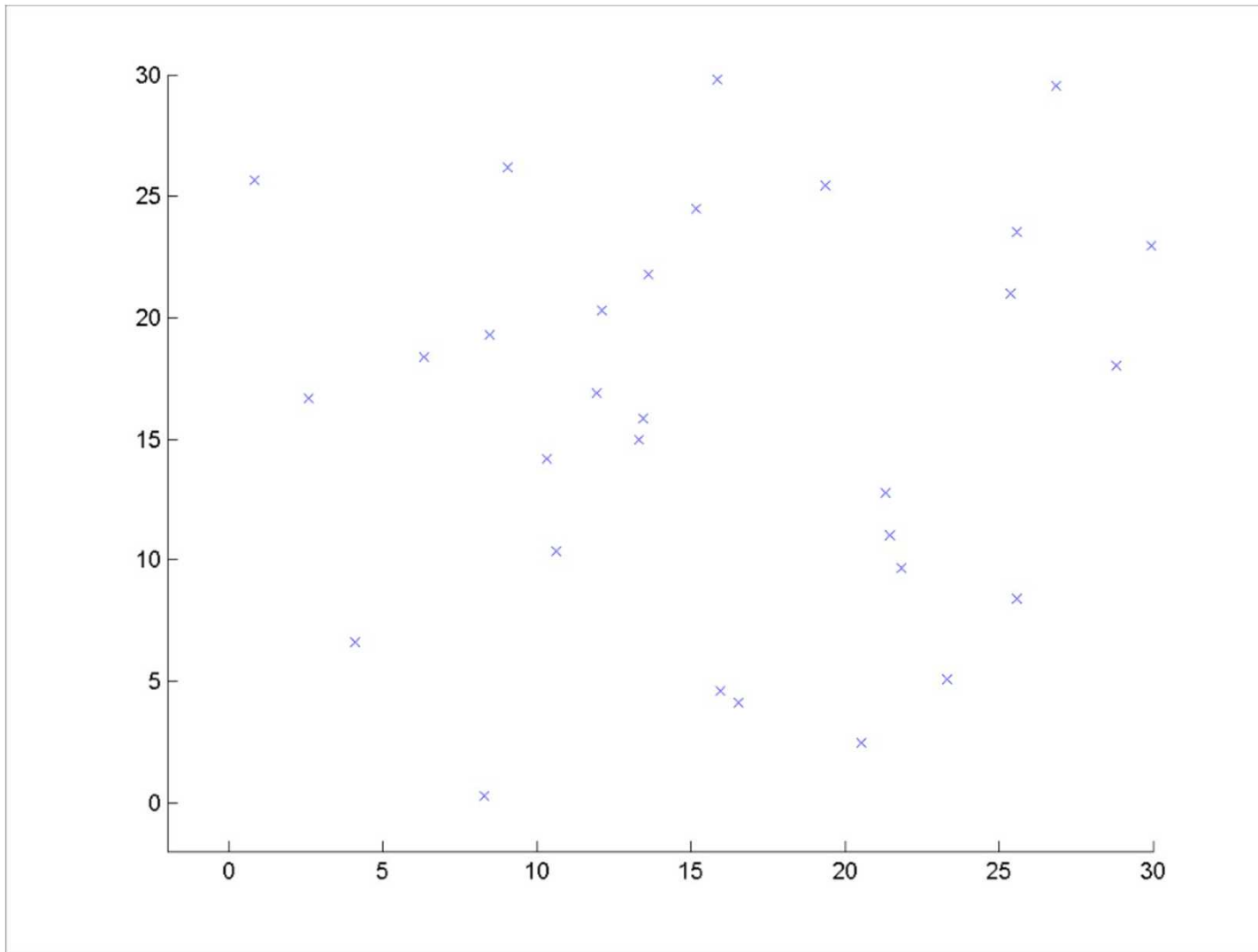
<http://www.crackedcamera.com/flock-of-birds-san-diego-ca/>

# Particle Swarm Optimization

- „Optimalizace hejnem částic“
- poprvé uveřejněna v roce 1995 autory Kennedym a Eberhartem [1]
- stochastická metoda
- není potřeba gradient, Hessova matice
- pro diskrétní, spojité i kombinované úlohy



# Particle Swarm Optimization



# Particle Swarm Optimization

Původní přístup:

$$v_{id}^{k+1} = w \cdot v_{id}^k + c_1 \cdot rand() \cdot (p_{id} - x_{id}) + c_2 \cdot Rand() \cdot (p_{gd} - x_{id})$$

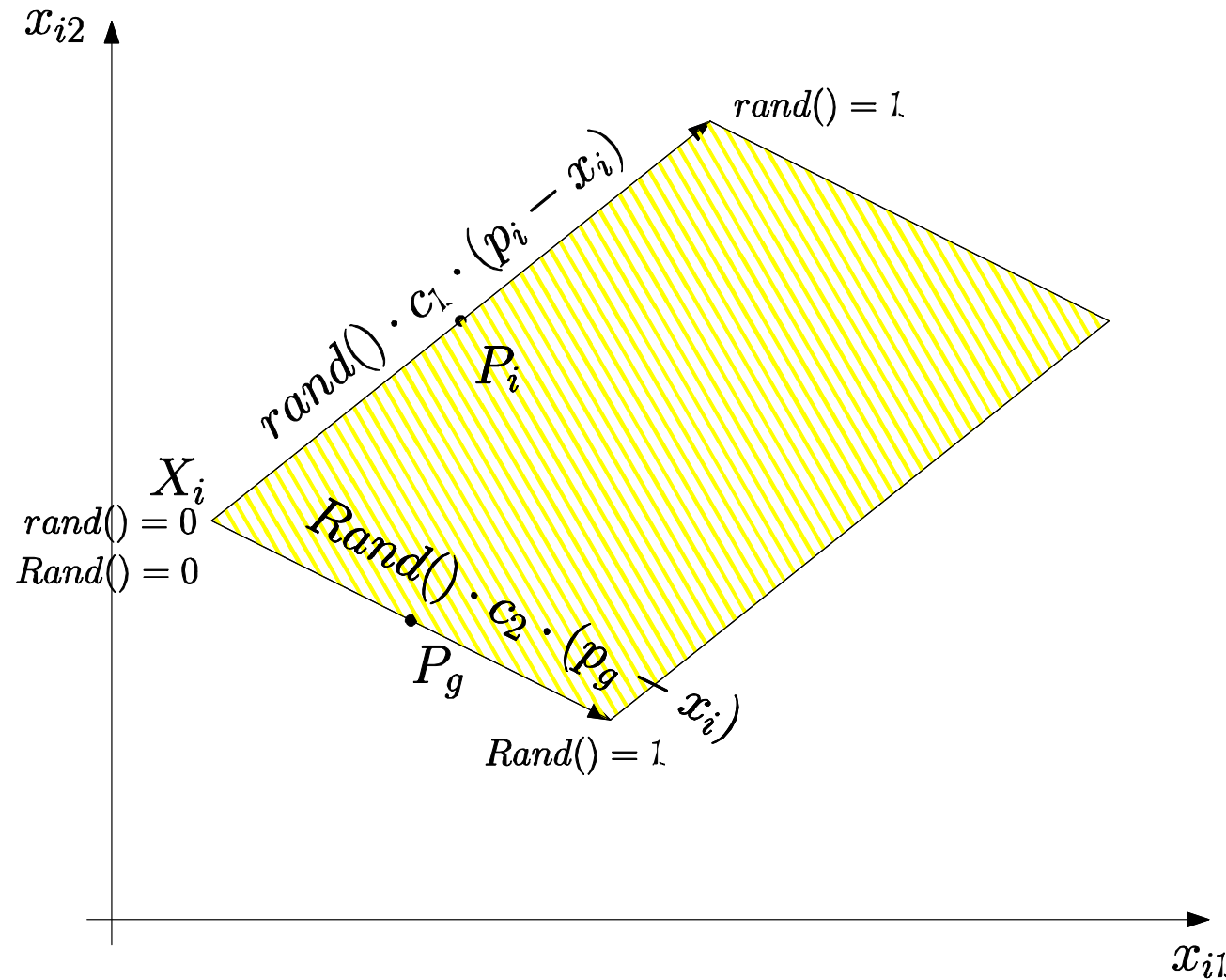
$$x_{id}^{k+1} = x_{id}^k + v_{id}^{k+1}$$

w – váhovací faktor

c<sub>1</sub> – kognitivní (poznávací) faktor

c<sub>2</sub> – sociální faktor

# Stochastický příspěvek rychlosti



# Particle Swarm Optimization

Clercova „Constriction method“ (1999) [2, 3]

$$v_{id}^{k+1} = K \cdot \left[ v_{id}^k + c_1 \cdot rand() \cdot (p_{id} - x_{id}) + c_2 \cdot Rand() \cdot (p_{gd} - x_{id}) \right]$$

$$K = \frac{2}{\left| 2 - \varphi - \sqrt{\varphi^2 - 4\varphi} \right|}, \text{ kde } \varphi = c_1 + c_2, \varphi > 4$$

$$x_{id}^{k+1} = x_{id}^k + v_{id}^{k+1}$$

K – faktor konstrikce (zúžení)

# Orientační hodnoty parametrů

$C_1, C_2$	2
rand(), Rand()	(0, 1)
w	0,4 - 0,9 pro konst.
K	0,729 (pro $\phi = 4,1$ )

pozn.: w - paralela s SA  
- lze konstantní číslo, lineárně se  
měnící hodnoty atp.

# Algoritmus

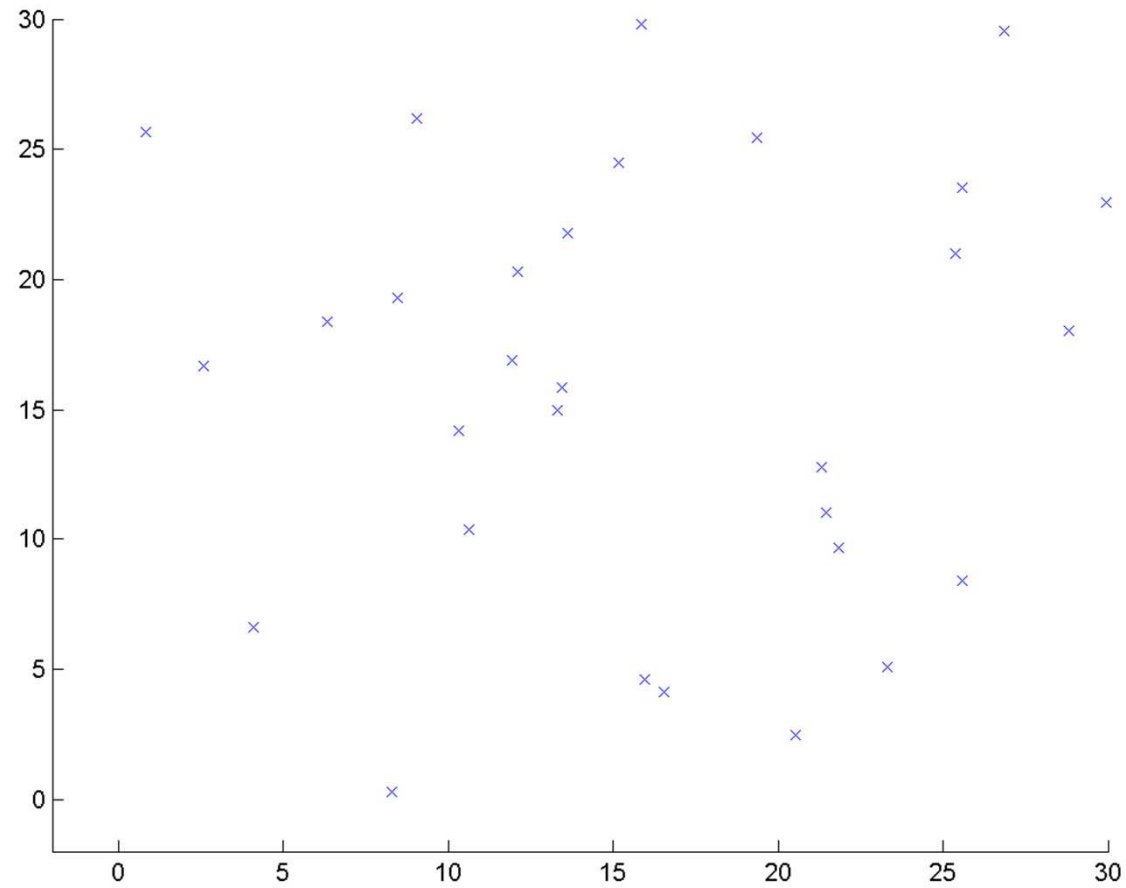
nastavení konstant, faktorů, ...

rozmístění počátečních částic v prostoru

určení počáteční rychlosti

```
while (ukončovací podmínka != true )  
    k++                % iterace cyklu  
    for i = 1 : počet částic  
        výpočet hodnoty účelové funkce  $f_i^k$   
        if ( $f_i^k < p_i$ ) then  $p_i = f_i^k$   
        if ( $f_i^k < p_g$ ) then  $p_g = f_i^k$   
        aktualizace rychlosti částice  $V_i$   
        aktualizace polohy částice  $X_i$   
    end  
end
```

# PSO.m



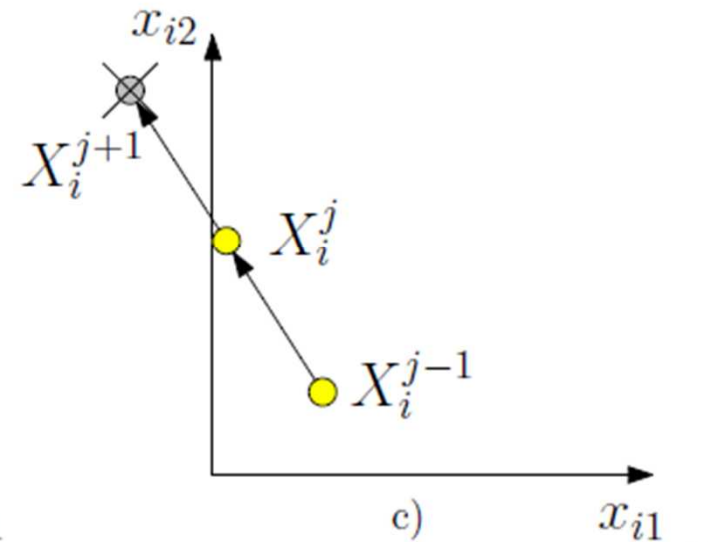
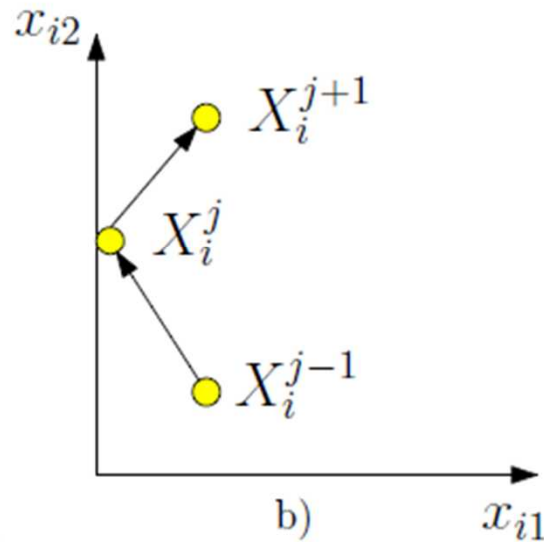
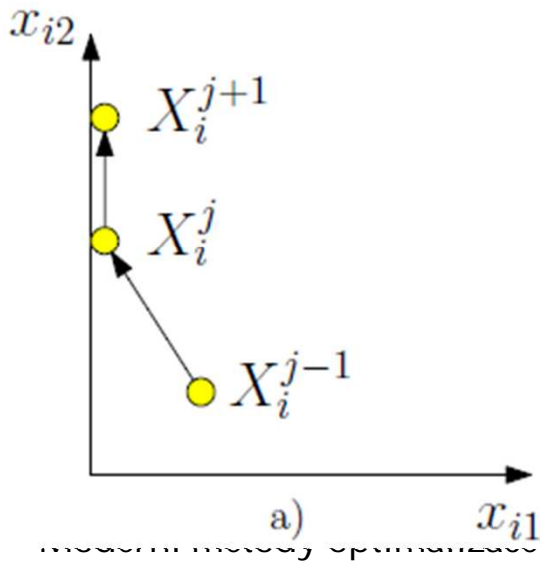
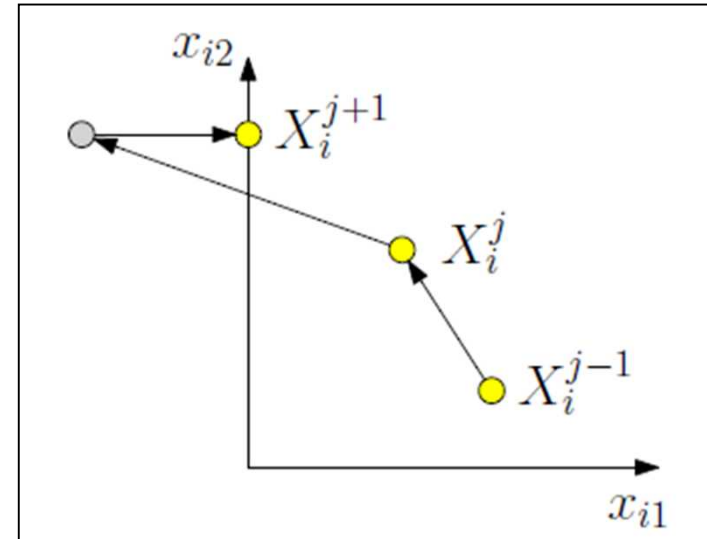
# Varianty PSO [4]

- náhodná reinicializace rychlosti částic
- omezení maximální rychlosti
  - omezení délky vektoru
  - omezení každé složky vektoru
- omezení minimální rychlosti
- potrhlost částic (craziness)
- váhovací faktor (konstantní, lineární, ...)
- faktor konstrikce
- a mnohé další

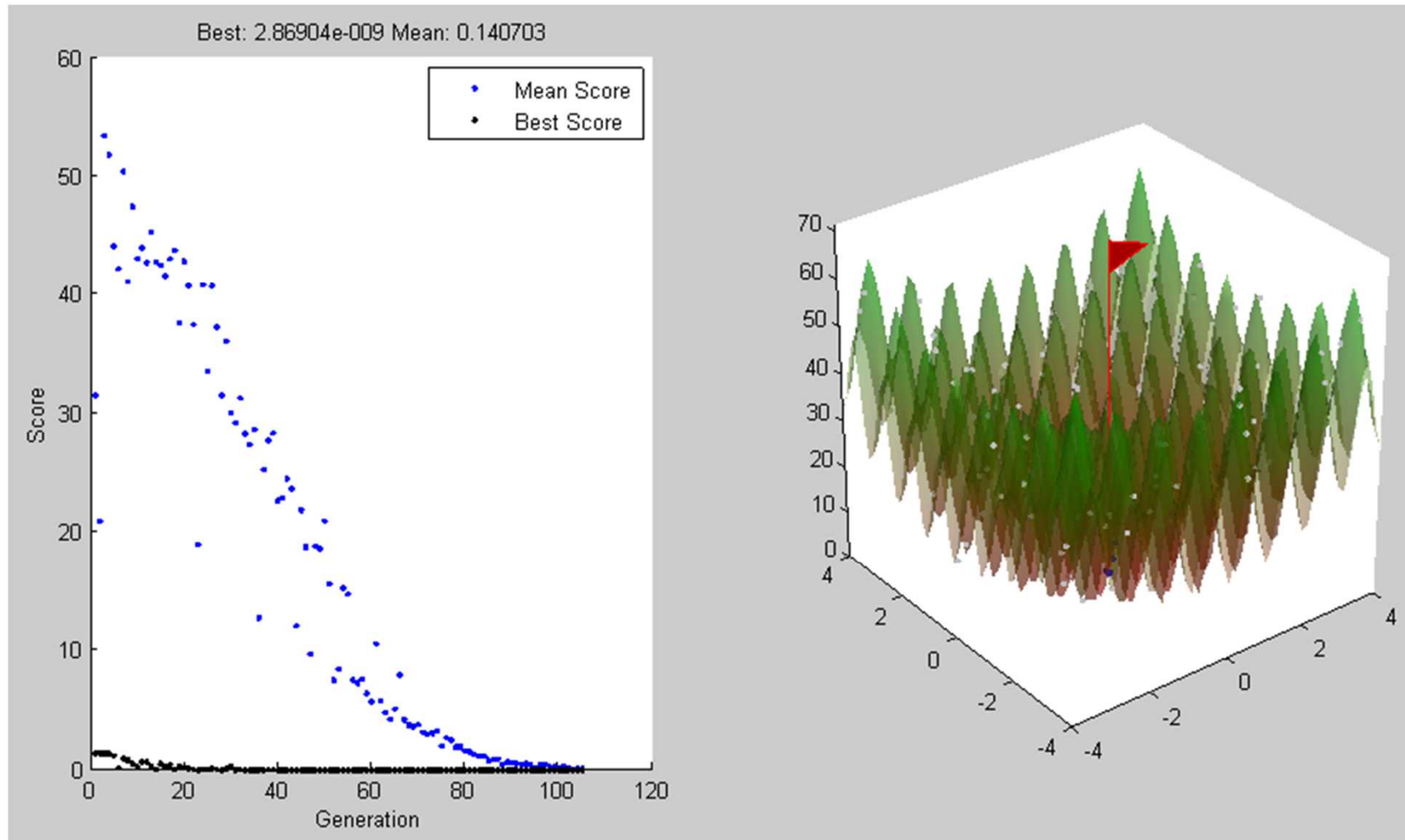


# Hraniční omezení

- Absorbující stěny
- Odráživé stěny
- Neviditelné stěny



# Another Particle Swarm Toolbox



# Reference

- [1] Kennedy, J.; Eberhart, R. C. (1995). Particle Swarm Optimization. IEEE International Conference of Neural Networks, 4: 1942 – 1948.
- [2] Clerc, M. (1999). The Swarm and The Queen: Towards a Deterministic and Adaptive Particle Swarm Optimization. IEEE Congress on Evolutionary Computation CEC'99, 3: 1951 – 1957.
- [3] Eberhart, R. C.; Shi, Y. (2000). Comparing Inertia Weights and Constriction Factors in Particle Swarm Optimization. IEEE Congress on Evolutionary Computation, 1: 84 – 88.
- [4] Wilke, D. N.; Kok, D.; Groenwold, A. A. (2006). Comparison of Linear and Classical Velocity Update Rules in Particle Swarm Optimization : Notes on Diversity. International Journal for Numerical Methods in Engineering, 70: 962 – 984.

# Reference

- [5] Shi, Y.; Eberhart, R. C. (1999). Empirical Study of Particle Swarm Optimization. IEEE Congress on Evolutionary Computation CEC'99, 3: 1945 – 1950.
- [6] Clerc, M. (2006). Particle Swarm Optimization. 1. vyd. Wiley-ISTE. ISBN-13: 978-1-905209-04-0

# Ant Colony Optimization

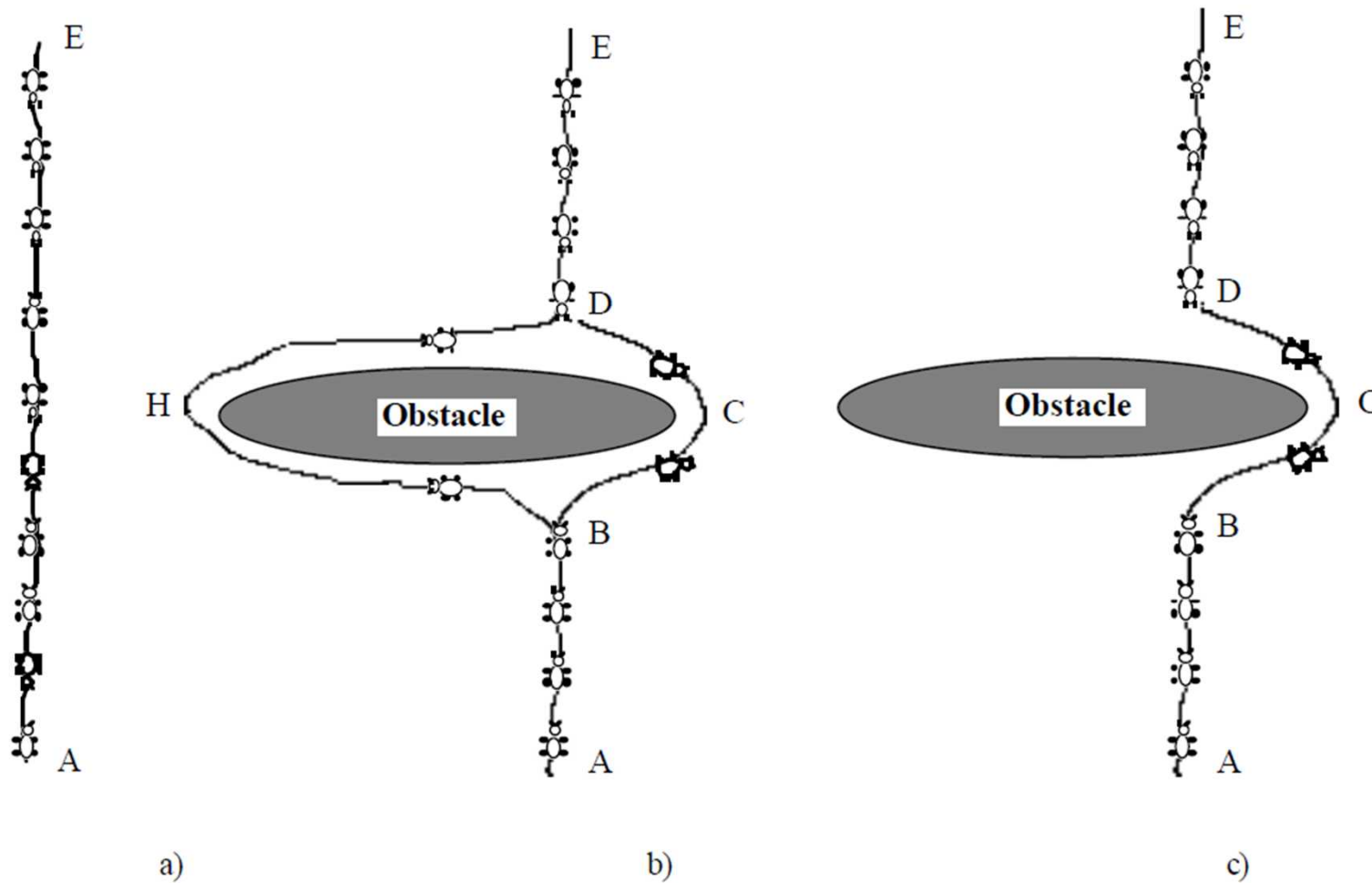


<http://ananthu-howtonameit.blogspot.com/2010/11/secret-of-success-get-mind-set-of-ant.html>

# Ant Colony Optimization

- „Optimalizace mravenčí kolonií“
- v 90. letech uveřejněna M. Dorigem a kol.
- stochastická metoda
- není potřeba gradient, Hessova matice
- algoritmus původně testovaný na TSP
  
- feromon

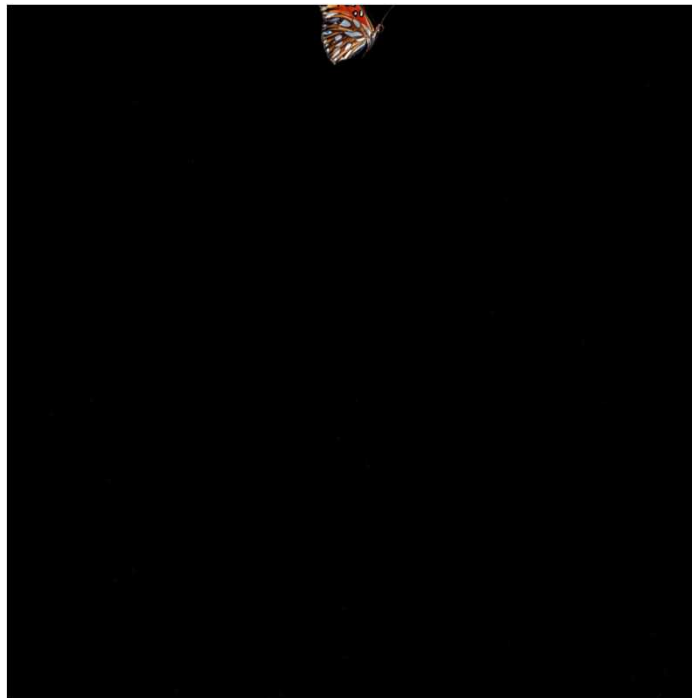
# Ant Colony Optimization



převzato z [7]

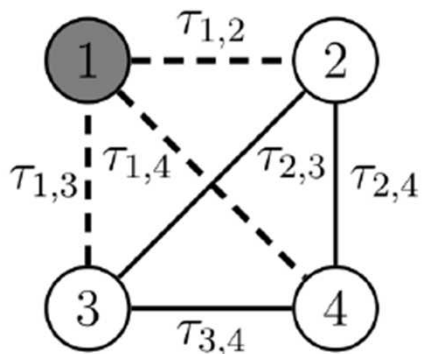
# Ant Colony Optimization

<http://users.sussex.ac.uk/~tn41/antStage2/war/AntAppletFull.html>



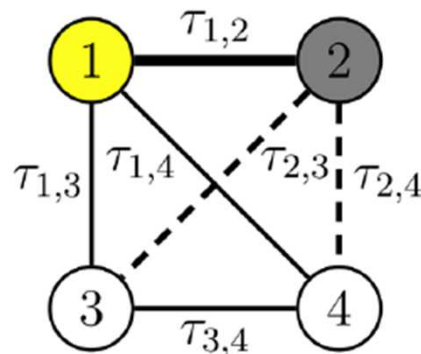


# Travelling Salesman Problem



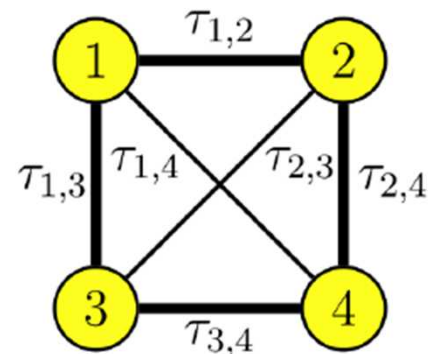
$$\mathbf{p}(e_{1,j}) = \frac{\tau_{1,j}}{\tau_{1,2} + \tau_{1,3} + \tau_{1,4}}$$

(a) First step of the solution construction.



$$\mathbf{p}(e_{2,j}) = \frac{\tau_{2,j}}{\tau_{2,3} + \tau_{2,4}}$$

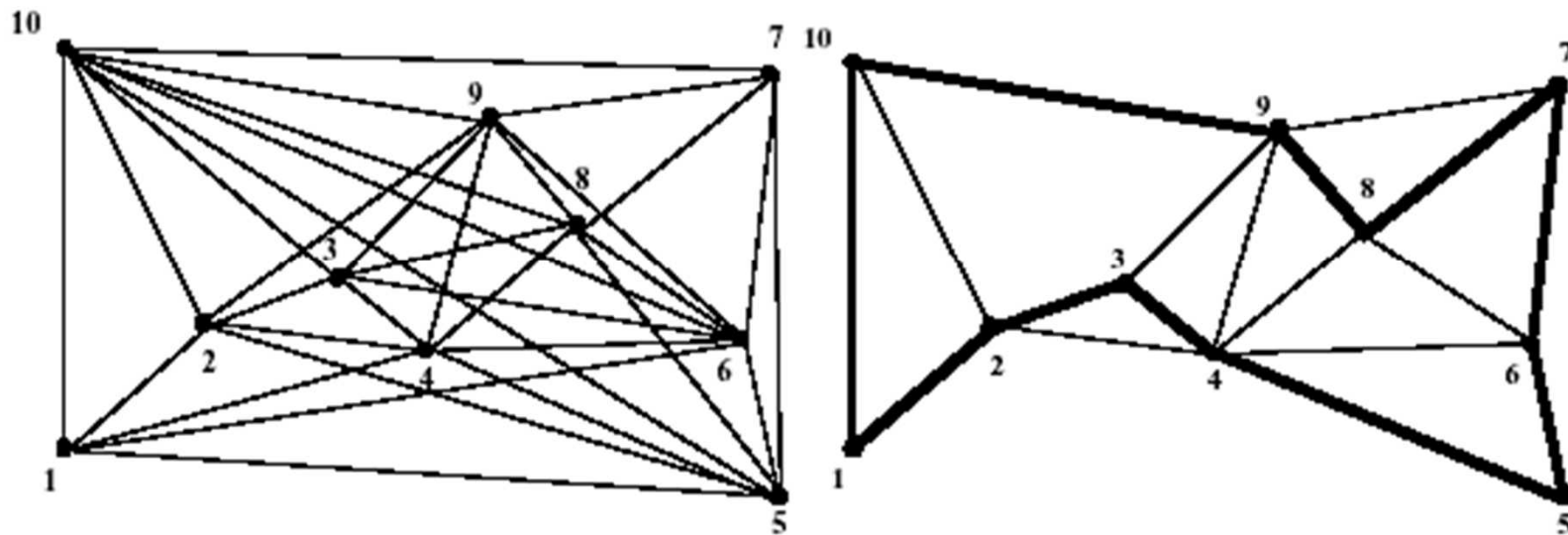
(b) Second step of the solution construction.



(c) The complete solution after the final construction step.

převzato z [8]

# Travelling Salesman Problem



# Algoritmus

nastavení počtu mravenců, jejich rozmístění (náhodné)

nastavení počátečních hodnot feromonů

while (ukončovací podmínka != true )

    t++                                   % iterace cyklu (čas)

    sestavení řešení

    (lokální průzkum místa)

    aktualizace hodnot feromonů

    přemístění mravenců

end

# Ant Colony Optimization

- Ant System (AS)
- *MAX-MIN* Ant System
- Ant Colony System
- Hyper-cube AS

# Způsoby aktualizace feromonu

- Ant System
  - feromon je aktualizován všemi mravenci

$$\tau_{ij}^{t+1} = (1 - \rho) \cdot \tau_{ij}^t + \sum_{k=1}^m \Delta \tau_{ij}^t, \quad \Delta \tau_{ij}^t = \begin{cases} Q / L_k^* \\ 0^{**} \end{cases}$$

\* je-li mravenec k použit na hraně (i,j) při řešení

\*\* v ostatních případech

# Způsoby aktualizace feromonu

- *MAX-MIN* Ant System
  - feromon aktualizují jedinci s nejlepším nalezeným řešením (nejlepší v iteraci, nejlepší od začátku nebo kombinace)

$$\tau_{ij}^{t+1} = \left[ (1 - \rho) \cdot \tau_{ij}^t + \Delta \tau_{ij}^{best} \right]_{\tau_{\min}}^{\tau_{\max}},$$

$$\Delta \tau_{ij}^{best} = \begin{cases} 1 / L_{best}^* \\ 0^{**} \end{cases}$$

\* patří-li (i,j) k nejlepší nalezené cestě

\*\* v ostatních případech

# Způsoby aktualizace feromonu

- Ant Colony System

- lokální aktualizace feromonu po každém vykonaném kroku

$$\tau_{ij} = (1 - \varphi) \cdot \tau_{ij} + \varphi \cdot \tau_0$$

- globální aktualizace feromonu na konci iterace jako u *MAX-MIN AS*

$$\tau_{ij} = \begin{cases} (1 - \rho) \cdot \tau_{ij} + \rho \cdot \Delta\tau_{ij}^* \\ \tau_{ij}^{**} \end{cases}$$

\* patří-li (i,j) k nejlepší nalezené cestě,  $\Delta\tau_{ij} = 1/L_{best}$

\*\* v ostatních případech

# Reference

- [7] Colorni, A.; Dorigo, M.; Maniezzo, V. (1991). Distributed Optimization by Ant Colonies. In European Conference on Artificial Life, 134-142.
- [8] Blum, Ch. (2005). Ant Colony Optimization: Introduction and Recent Trends. Physics of Life Reviews, 2(4): 353 – 373.
- [9] Dorigo, M. (2004). Ant Colony Optimization. 1. vyd. The MIT Press. ISBN: 978-0262042192.
- [10] Dorigo, M.; Birattari, M.; Stutzle, T. (2006). Ant Colony Optimization: Artificial Ants as a Computational Intelligence Technique. Technical Report No. RT/IRIDIA/2006-023, Université Libre de Bruxelles, IRIDIA.



# Swarm Intelligence - shrnutí

- stochastické metody – nemusí nutně najít globální optimum
- rozdíl v „komunikaci“
  - PSO – na úrovni hejna
  - ACO – feromonem lokálně

# Odkazy

<http://www.mathworks.com/matlabcentral/fileexchange/11559-particle-swarm-optimization-simulation>

<http://www.mathworks.com/matlabcentral/fileexchange/25986-another-particle-swarm-toolbox>

<http://www.liacs.nl/~baeck/NC/slides/Applet/ants.html>

<http://www.mathworks.com/matlabcentral/fileexchange/14543>

**Prosba.** V případě, že v textu objevíte nějakou chybu nebo budete mít námět na jeho vylepšení, ozvěte se prosím na **matej.leps@fsv.cvut.cz**.

*Datum poslední revize: 8.11.2011*

*Verze: 001*