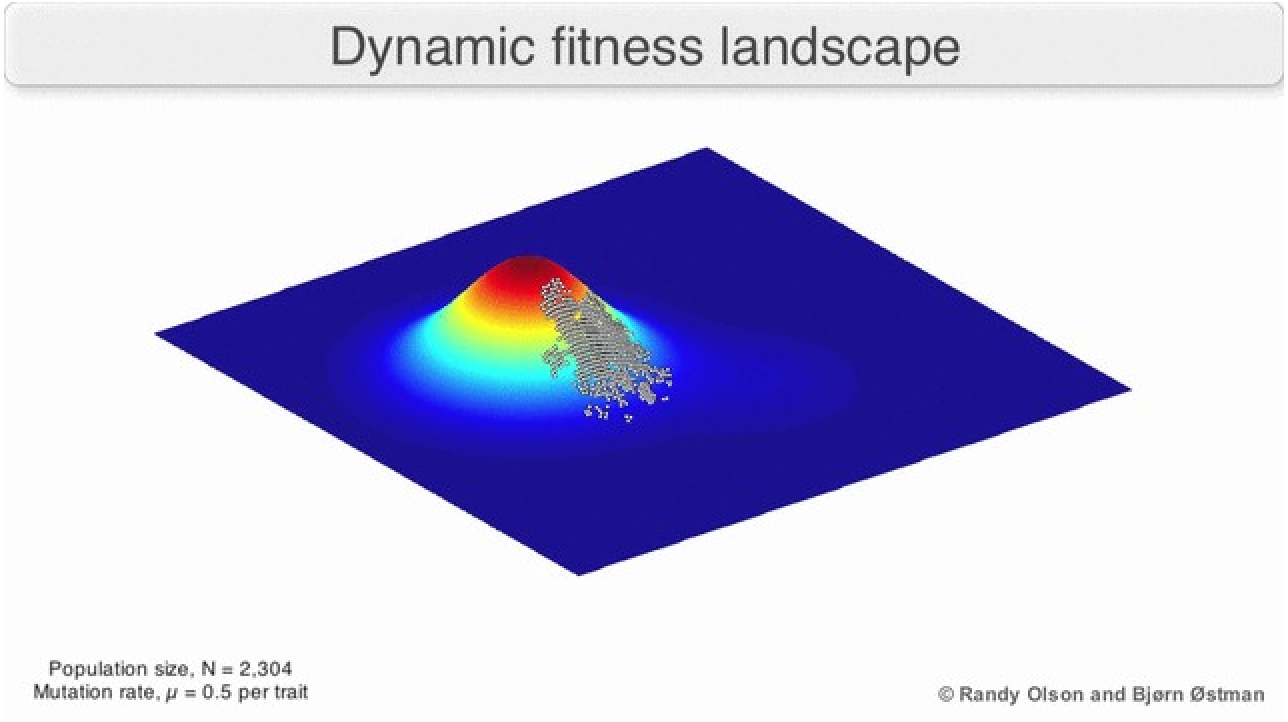
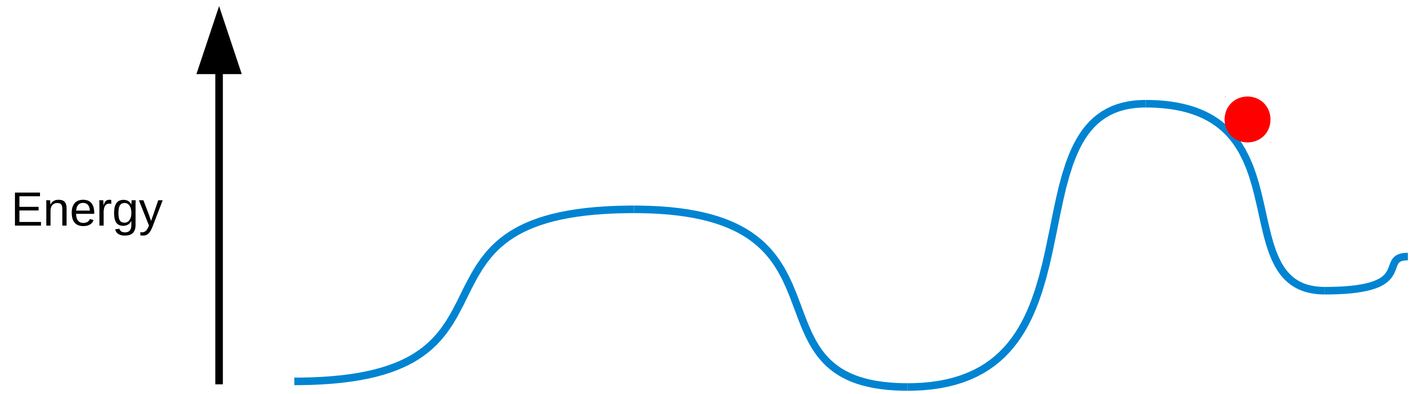
Stochastic Optimization



Simmulated Annealing

Simmulated Annealing

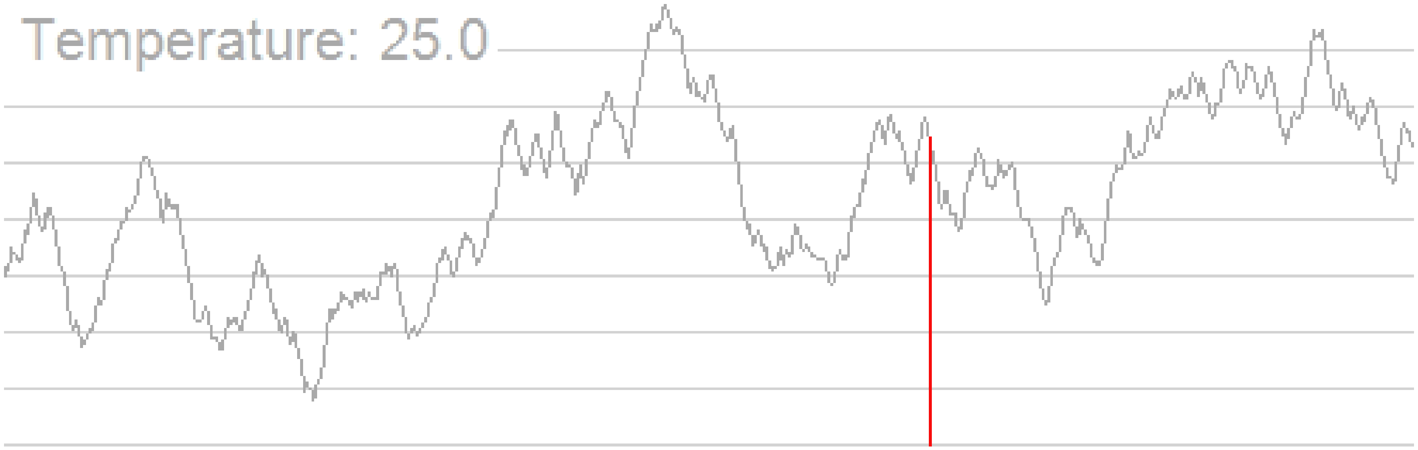


Greedy algorithms can get stuck in local optima, and depend on initial conditions :(

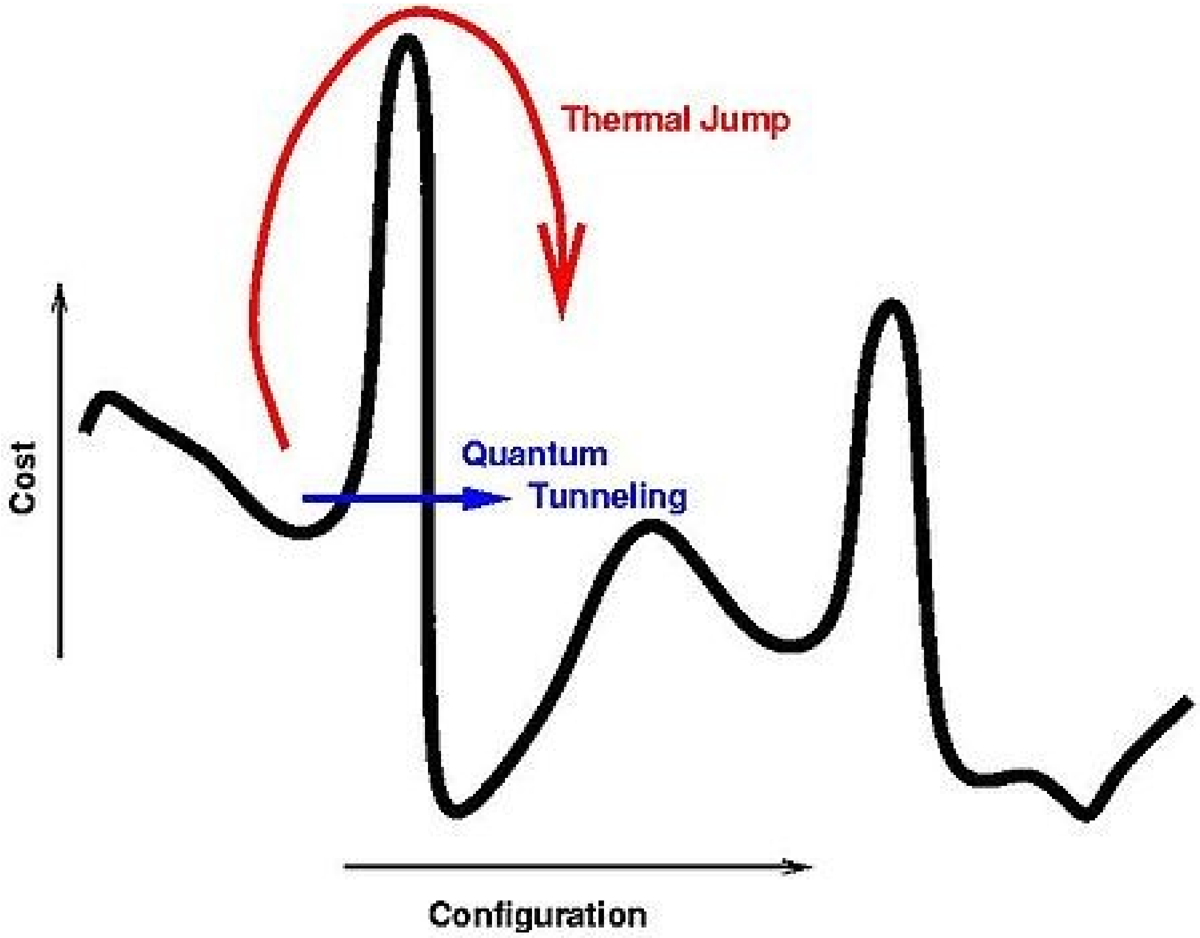
Simmulated Annealing procedure:

- Initial configuration *i* with E_i
- Choose random (nearby) state *j* with E_j
- Accept change $i \rightarrow j$ with probability $P_{i\rightarrow j} = \min\left[1, \exp\left(-\frac{E_j - E_i}{T}\right)\right]$
- Gradually decrease *T* according to a “cooling schedule”

Simmulated Annealing

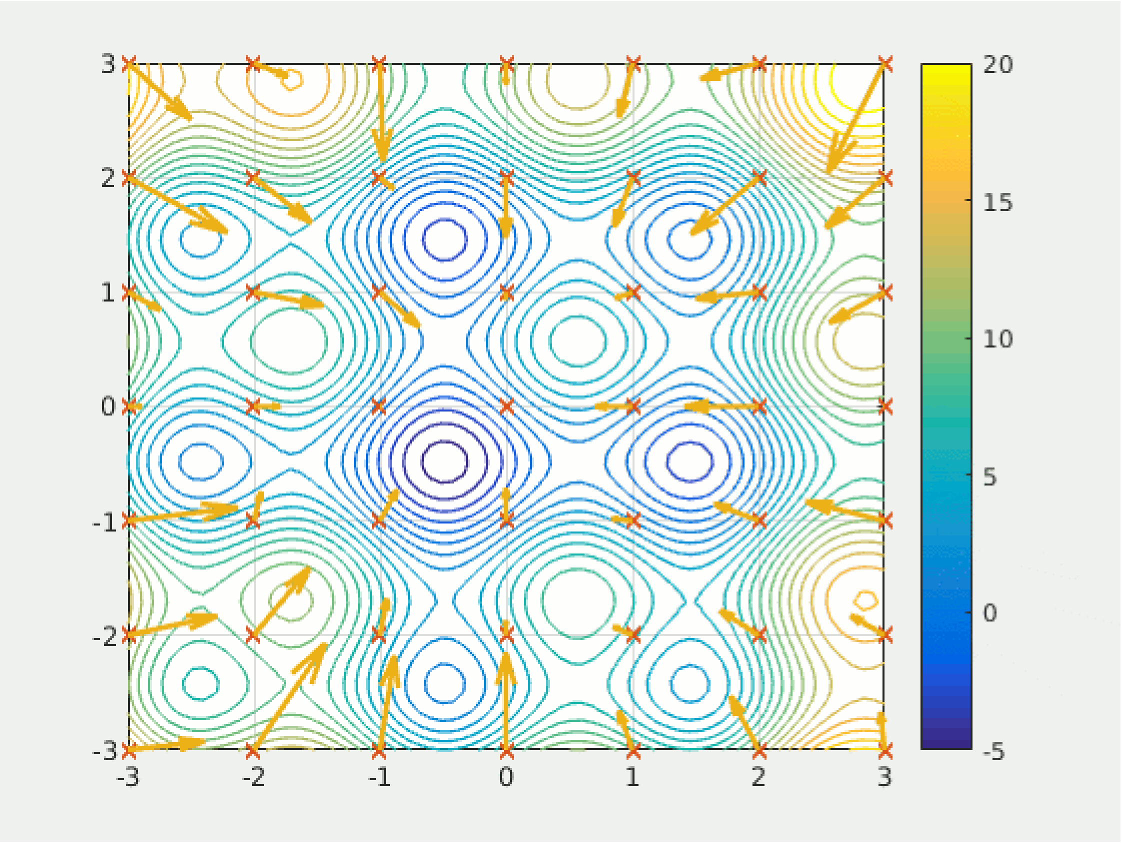


Quantum Annealing



Swarm intelligence

Swarm intelligence



Ant Colony Optimization

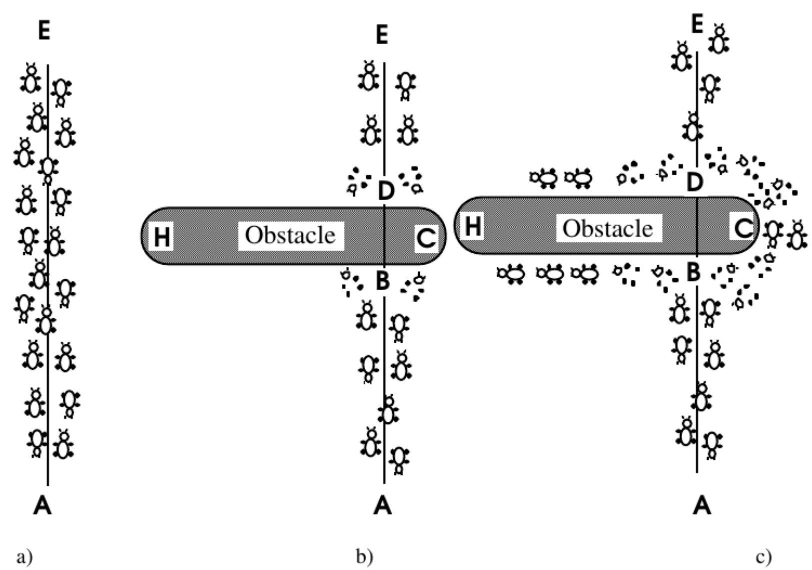
Swarm intelligence

Ant Colony Optimization

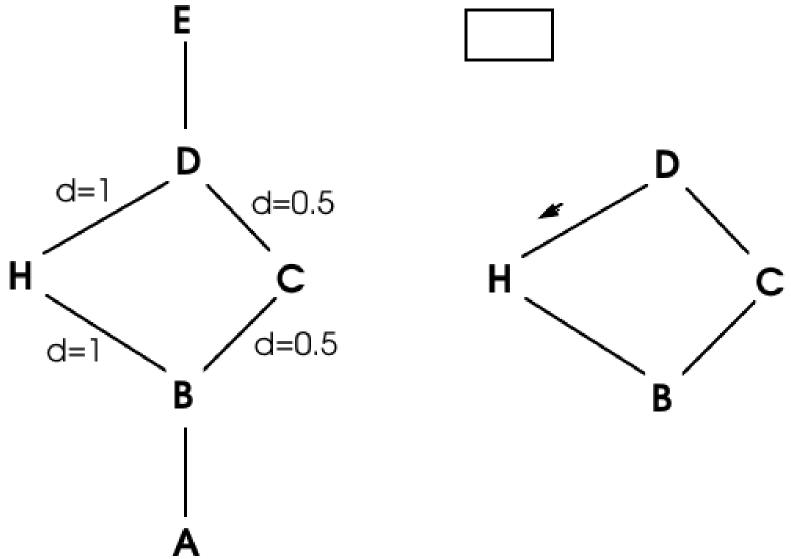
Submitted to IEEE Transactions on Systems, Man, and Cybernetics

The Ant System:
Optimization by a colony of cooperating agents

Marco Dorigo, Vittorio Maniezzo, Alberto Colorni



Ant Colony Optimization



TSP: Which is the shortest path that visits each city exactly once?

Ant Colony Optimization

Let $\tau_{ij}(t)$ be the *intensity of trail* on edge (i,j) at time t. Each ant at time t chooses the next town, where it will be at time t+1. Therefore, if we call an *iteration* of the AS algorithm the m moves carried out by the m ants in the interval (t, t+1), then every n iterations of the algorithm (which we call a cycle) each ant has completed a tour. At this point the trail intensity is updated according to the following formula

$$\tau_{ij}(t+n)=\rho\cdot\tau_{ij}(t)+\Delta\tau_{ij} \tag{1}$$

where

ρ is a coefficient such that (1 - ρ) represents the *evaporation* of trail between time t and t+n,

$$\Delta\tau_{ij} = \sum_{k=1}^{m} \Delta\tau_{ij}^{k} \tag{2}$$

where $\Delta\tau_{ij}^{k}$ is the quantity per unit of length of trail substance (pheromone in real ants) laid on edge (i,j) by the k-th ant between time t and t+n; it is given by

$$\Delta\tau_{ij}^{k} = \begin{cases} \dfrac{Q}{L_k} & \text{if k - th ant uses edge (i, j) in its tour (between time t and t + n)} \\ 0 & \text{otherwise} \end{cases} \tag{3}$$

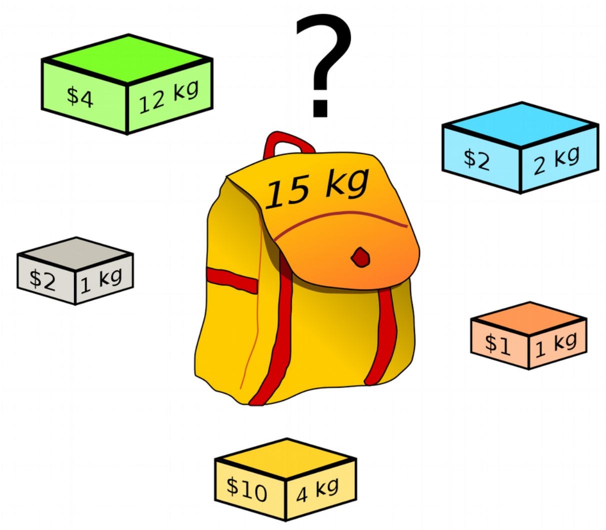
where Q is a constant and L_k is the tour length of the k-th ant.

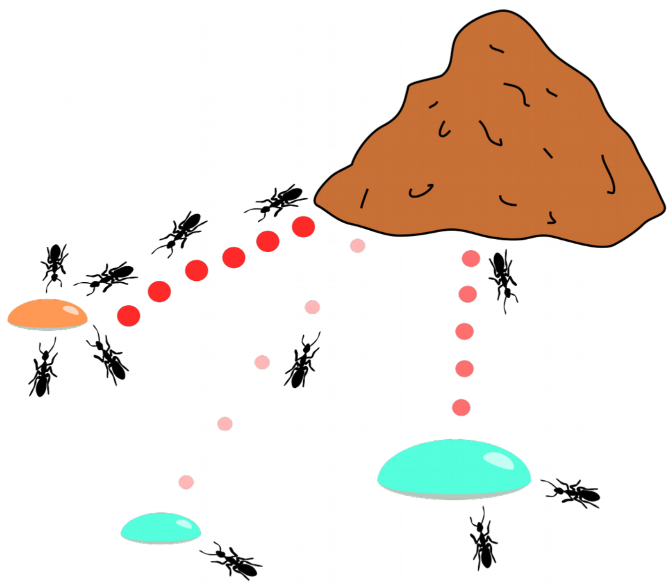
We call *visibility* η_{ij} the quantity $1/d_{ij}$. This quantity is not modified during the run of the AS, as opposed to the trail which instead changes according to the previous formula (1).

We define the transition probability from town i to town j for the k-th ant as

$$p_{ij}^{k}(t) = \begin{cases} \dfrac{[\tau_{ij}(t)]^{\alpha} \cdot [\eta_{ij}]^{\beta}}{\sum\limits_{k \in allowed_k} [\tau_{ik}(t)]^{\alpha} \cdot [\eta_{ik}]^{\beta}} & \text{if } j \in allowed_k \\ 0 & \text{otherwise} \end{cases} \tag{4}$$

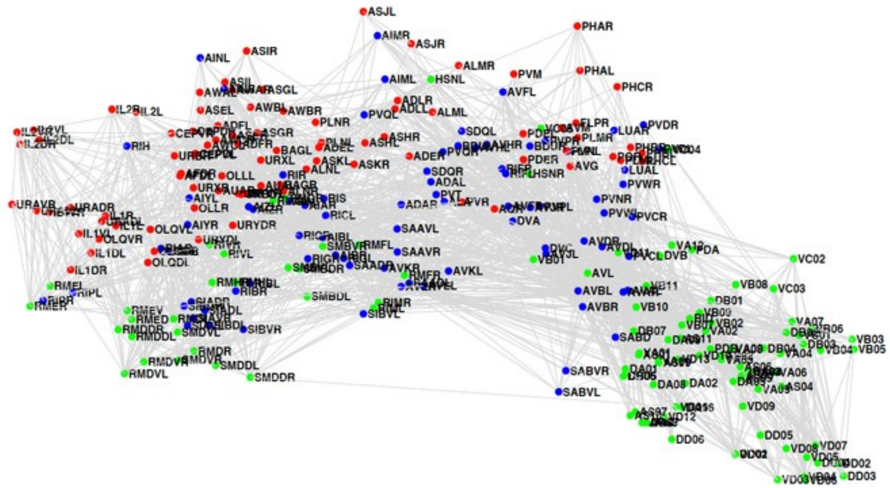
Ant Colony Optimization





Knapsack problem: Which boxes should be chosen to maximize the amount of money while still keeping the overall weight under or equal to 15 kg?

Ant Colony Optimization

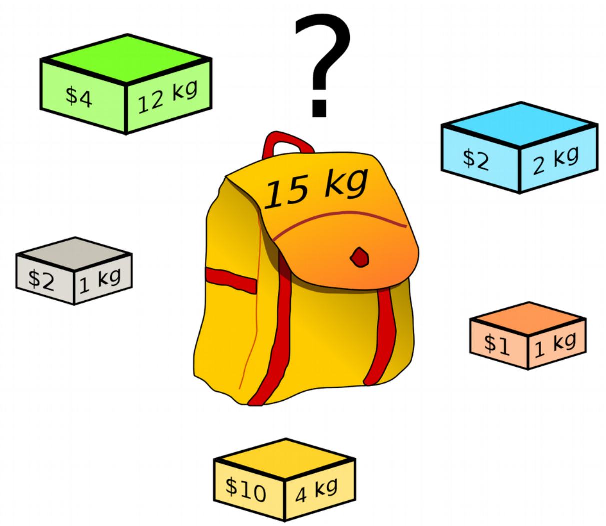


$$Q = \frac{1}{2m} \sum_{vw} \left[A_{vw} - \frac{k_v k_w}{2m} \right] \delta(c_v, c_w)$$

Finding community structure in very large networks

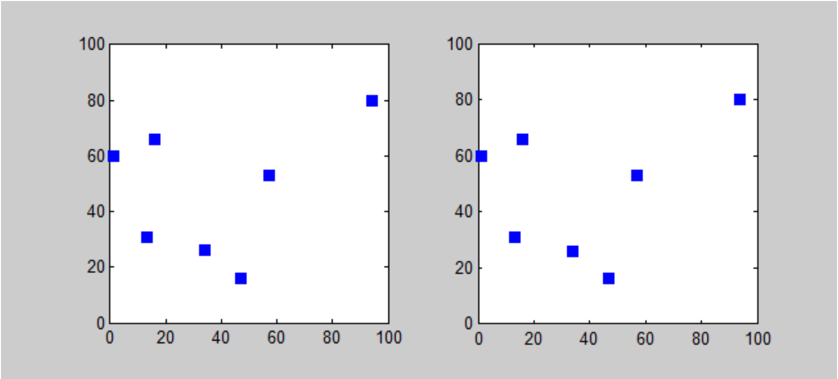
Aaron Clauset,[1] M. E. J. Newman,[2] and Cristopher Moore[1,3]

Knapsack problem:



Which boxes should be chosen to maximize the amount of money while still keeping the overall weight under or equal to 15 kg?

Travelling salesman problem:



Which is the shortest path that visits each city exactly once?