Stochastic Simulation
Simulated annealing

Bo Friis Nielsen

Institute of Mathematical Modelling

Technical University of Denmark

2800 Kgs. Lyngby – Denmark

Email: bfni@dtu.dk

Simulated annealing

- Stochastic algorithm for optimisation
- Large scale problems
- Attempts to find the global optimum in presence of multiple local optima

$$\min_{\boldsymbol{x}} f(\boldsymbol{x})$$

Physical inspiration (with apologies)

Steel and other materials can exist in several crystalline structures.

One - the ground state - has lowest energy.

The material may be "caught" in other states which are only locally stable.

This is likely to happen when welding, machining, etc.

By heating the material and **slowly** cooling, we ensure that the material ends in the ground state.

This process is called **annealing**.

P.d.f. of the state at fixed temperature

Use X to denote the state of the system (e.g., positions of atoms).

Let $U(x)$ to denote the energy of state x.

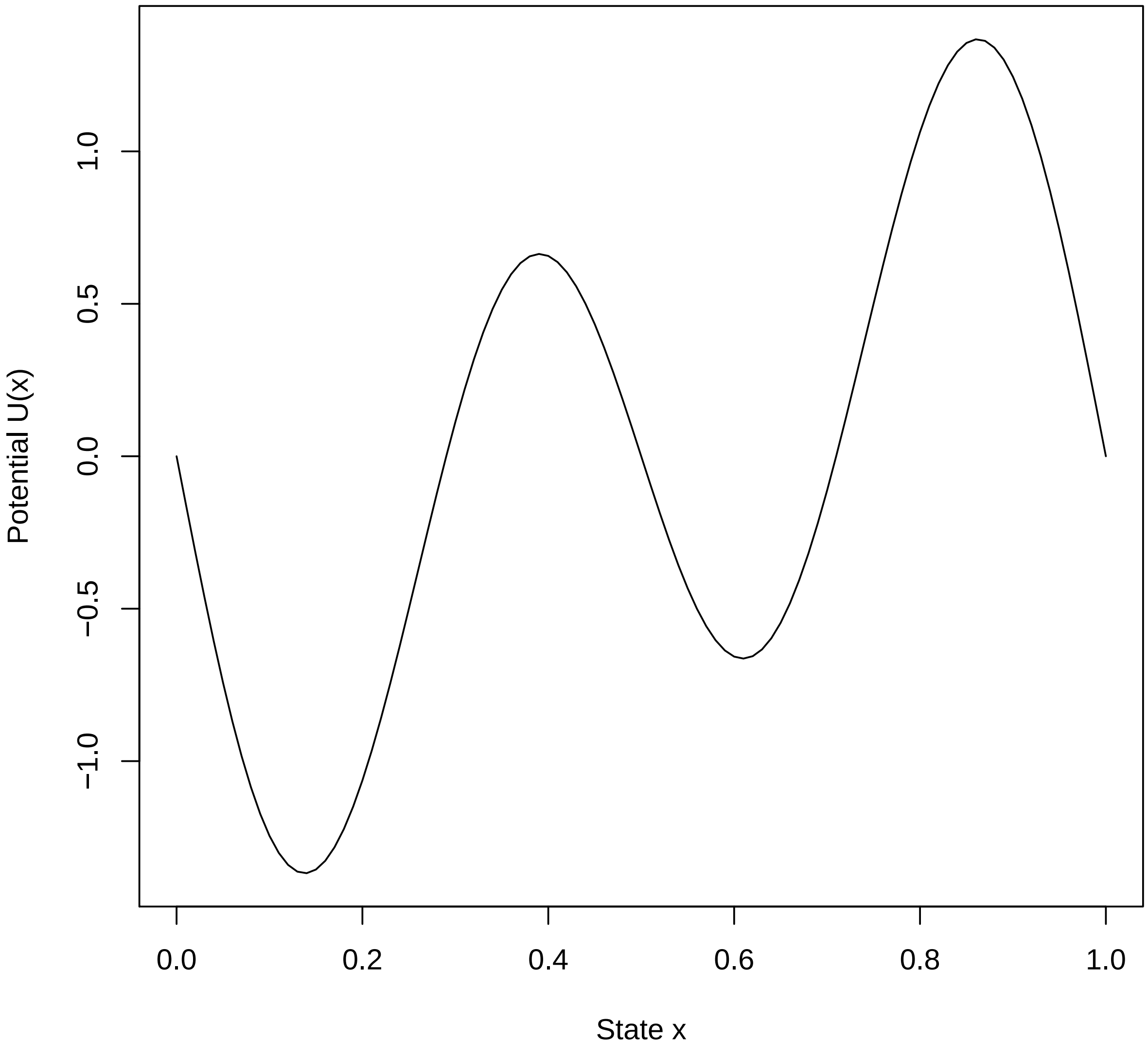
According to statistical physics, if the temperature is T, the p.d.f. of X is *the Canonical Distribution*

$$f(x,T) = c_T \cdot \exp(-\frac{U(x)}{T})$$

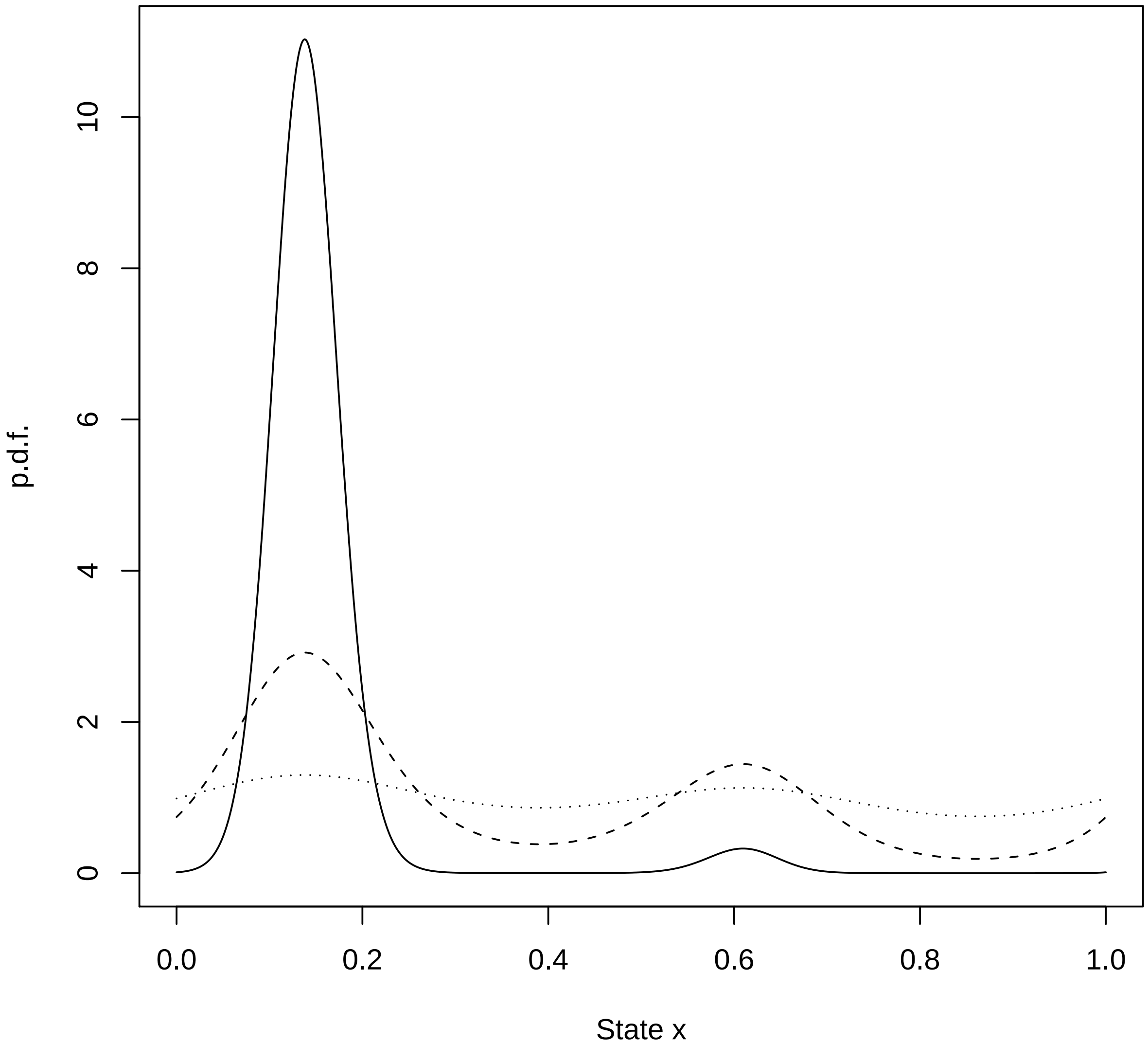
So states with low U are more probable; in particular at low T.

Note the normalization constant c_T is unknown; can be found by integration, but our algorithms will not require it.

Example energy potential



Corresponding p.d.f., for $T = 0.2, 1, 5$



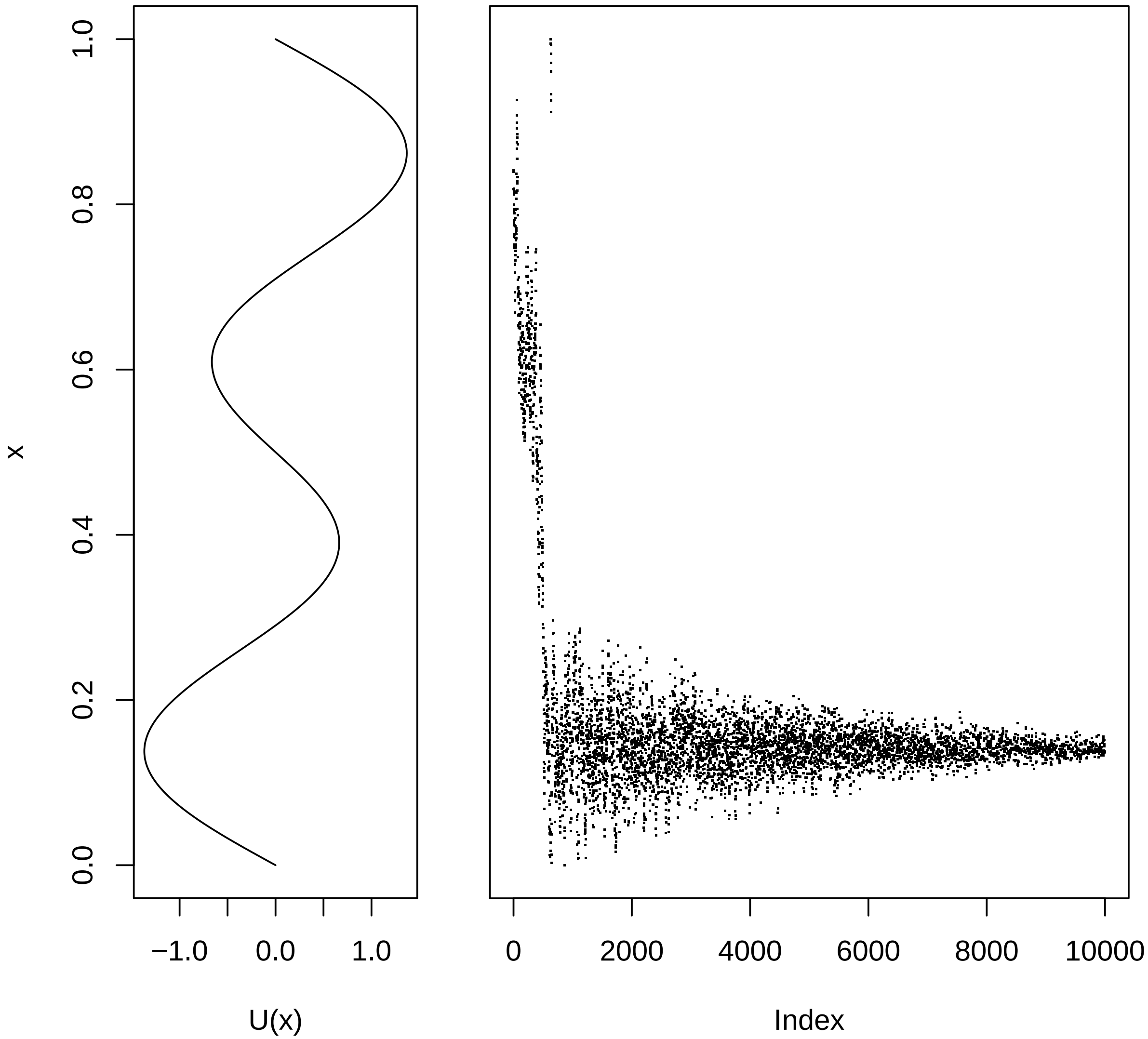
An algorithm for Simulated Annealing

Let the temperature be a decreasing function of time or iteration number - k.

At each time step, update the state according to the random walk Metropolis-Hastings algorithm for MCMC, where the target p.d.f. is $f(x, T_i)$.

I.e., permute the state X_i randomly to generate a candidate Y_i. If the candidate has lower energy than the old state, accept. Otherwise, accept only with probability

$$\exp(-(U(Y_i) - U(X_i))/T_i)$$


x
0.0
0.2
0.4
0.6
0.8
1.0
−1.0
0.0
1.0
U(x)
0
2000
4000
6000
8000
10000
Index

Different issues

- Try with different schemes for lowering the temperature
- Alternative initial solutions
- Different candidate generation algorithms
- Refine with local search

Travelling salesman problem (TSP)

A basic problem in combinatorial optimisation

Given n stations, and an n-by-n matrix A giving the cost of going from station i to j.

Find a route S (a permutation of $1,\ldots,n$) which

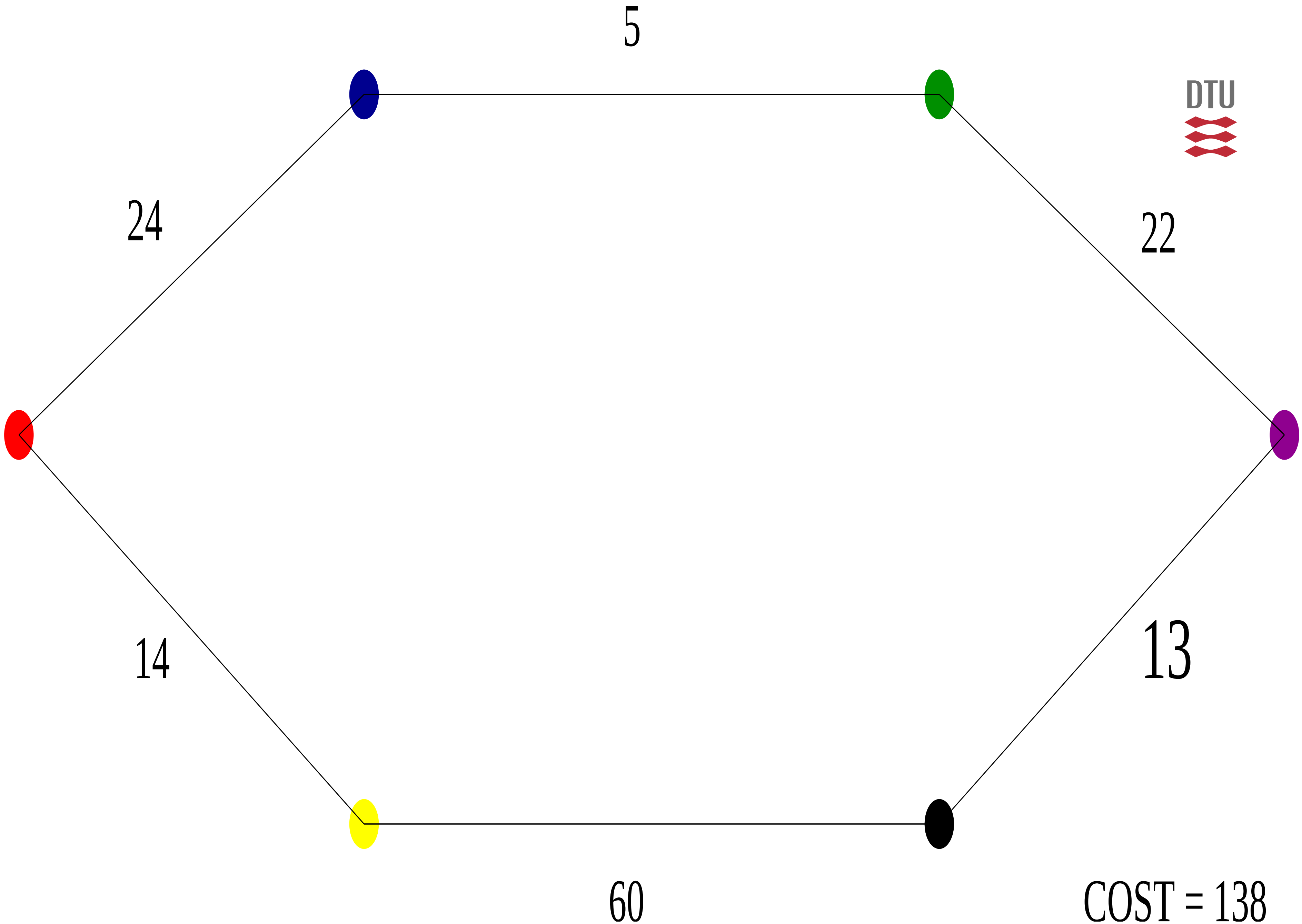
- starts and ends at station 1, $S_1 = 1$
- has minimal total cost

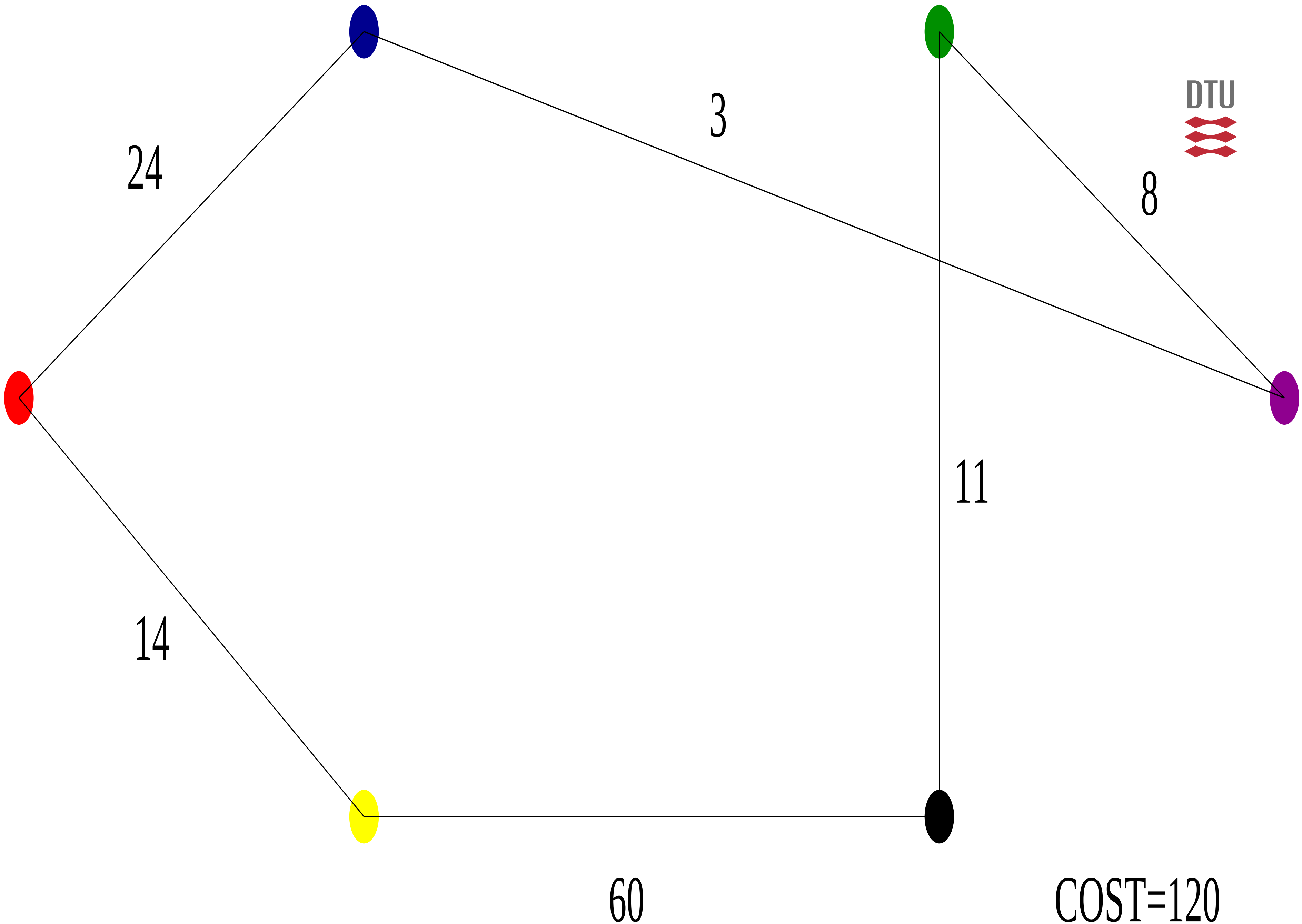
$$\sum_{i=1}^{n-1} A(S_i, S_{i+1})$$

Cost matrix - an example

Town	Town to					
from	1	2	3	4	5	6
1	-	5	3	1	4	12
2	2	-	22	11	13	30
3	6	8	-	13	12	5
4	33	9	5	-	60	17
5	1	15	6	10	-	14
6	24	6	8	9	40	-

- Initial solution: $\{1,2,3,4,5,6,1\}$ initial cost: $5+22+13+60+14+24 = 138$


5
24
22
14
13
60
COST = 138



24
3
8
11
14
60
COST=120

Exercise 7

Implement simulated annealing for the travelling salesman.

Have input be positions in plane of the n stations.

Let the cost of going $i \mapsto j$ be the Euclidian distance between station i and j.

As cooling scheme, use e.g. $T_k = 1/\sqrt{1+k}$.

The route must end where it started.

Initialise with a random permutation of stations.

As proposal, permute two random stations on the route.

Plot the resulting route in the plane.

Debug with stations on a circle. Then modify your progamme to work with costs directly and apply it to the cost matrix from the course homepage.