Optimization Problems and Travelling Salesman Problem

Shan He

School for Computational Science
University of Birmingham

Module 06-27818 and 27819: Advanced Aspects of Nature-Inspired Search and Optimisation (Ext)

Outline of Topics

Motivating example

How to become rich and healthy?

A pathetic example

How to save money but still stay alive:

- Save money: minimise the cost of buying food
- Being alive: For each nutrient, e.g., Vitamin C, at least meet the minimum required level

Food	**Calcium**	**Vitamin C**	**Calories**	Price (GBP/100g)
Broccoli	47	89.2	53	0.381
Milk	276	0	120	0.1
Oranges	40	53.2	87	0.271

Exercise 1: Let's sovle it!

- Let's solve it by LibraOffice Calc Solver
- Download the zip file from Canvas
- Open DietProblem.ods
- Click Menu → Tools → Solver

Diet problem

- Diet problem:
  - Given: a set of available foods with cost and nutrition information
  - Sought: selected foods with minimum cost but also satisfy daily nutritional requirement
- Studied in the 1930s and 1940s.
- Motivated by USA Army's desire to minimize the cost of feeding soldiers while still providing a healthy diet
- Nobel Laureate George Stigler further formulated this as Stigler Diet problem
- Essentially an optimisation problem (linear programming problem).

Code example: Generating solutions for TSP

- Download the source code from Canvas
- `load('cities.mat')`
- Open the matrix 'cities', which is a 2×48 cities TSP problem
  - 48 USA state capital cities
  - The minimal tour has length 10628.
  - Check this page
- Open the optimal solution 'att48_s.txt' file and copy
- `optimalsolution =[` paste the solution `]`
- `inputcities = cities(:,optimalsolution)`
- `distance(inputcities)`
- Read help file about how to generate random permutation: `help randperm`

Exercise 2: Randomised search algoirthms for TSP

- Open 'randomsearch_skeleton.m' file
- Complete the code