

# An introduction for synthetic data

## 1. Overview

---

This file introduces the data of instances with varying scales. All instances are divided into two groups, small-scale instances, and large-scale ones. They are saved in two files named Small-scale instances and Large-scale instances. Each file contains several files to store the instances with specific scales. For example, in the file "scale\_20\_100", instances consist of 20 nodes and 100 edges.

## 2. Introduction of instances

---

Each instance consists of two txt files named 'edta\_X.txt' and 'ndta\_X.txt', where 'X' distinguishes the different instances. The 'edta\_X.txt' file stores a matrix with five rows whose columns represent different edges. The explanations in order are shown as follows.

1. node index (int): the tail of this edge;
2. node index (int): the head of this edge;
3. capacity (int): the maximum allowable flow value of this edge;
4. fixed cost (double): the fixed cost of this edge;
5. unit cost (double): the unit cost of this edge.

The 'ndta\_X.txt' file stores a vector whose item is the supply/demand of the node corresponding to the current index.

## 3. Data load example

---

There is a specific example in Python to illustrate the load of instances.

```
with open("./edta_1.txt", "r") as f:
    tail = list(map(int, f.readline().split("\t")))
    head = list(map(int, f.readline().split("\t")))
    capa = list(map(float, f.readline().split("\t")))
    fixc = list(map(float, f.readline().split("\t")))
    unic = list(map(float, f.readline().split("\t")))
    print(tail[:5])
    print(head[:5])
    print(capa[:5])
    print(fixc[:5])
    print(unic[:5])
```

[7] ✓ 0.0s Python

```
... [0, 0, 0, 0, 1]
     [2, 6, 14, 19, 5]
     [24.0, 12.0, 17.0, 16.0, 18.0]
     [9.732463617723024, 9.903170815180982, 6.832871027351818, 3.314229496908263, 8.491741601431354]
     [10.166310474594273, 8.74788315237505, 4.023415667432704, 9.172069888394654, 9.249257153618888]
```