License:

The license of Gurobi 10.0.0 can be downloaded at the official website of Gurobi: https://www.gurobi.com/

Summary of the functions:

- *function [output_args] = StartfunctionUFL(m,n,option)*

This function is the main function, including three options, each of which calls a different function.

- *function [c,f,c1,mn,data] = GenerateUFL(m,n)*

This function is used to randomly generate UFL instances. Our 100 UFL instances are generated using this function. Since random instances have already been generated, you can directly access them in the data folder.

- *function [v1,u1,obj1,objboundUFL,timeUFL,result] = NetworkGurobi(m,n,c,f)*

This function constructs a network for the UFL problem and calculates its optimal solution.

- *function [AdjustmentCostTotalPer,DiffNumberPer,cfinal,ufinal,ffinal,vfinal,FVAL] = InvCostAdjUFL(m,n,dataread)*

This function is used to solve the inverse UFL game, where [l, u] = [v*, v*].

- *function [AdjustmentCostTotalPer,DiffNumberPer,cfinal,ufinal,ffinal,vfinal,FVAL] = InvCostAdjUFLRange(m,n,dataread)*

This function is used to solve the inverse UFL game, where [l, u] = [0.95v*, 1.05v*].

- *function [output_args] = checkresult*

This function is used to compute statistical metrics for the results of 100 instances.

Code instructions:

To run the code smoothly and correctly, please set the variables in the file "UFL\src\StartfunctionUFL.m" as described below. Matlab 2016 and Gurobi 10.0.0 are recommended.

1. Set parameters m, n

Note that in the instances we generate, m = n: where m represents the number of facilities, and n represents the number of customers.

2. Set parameter option

option = 1 is to generate and save random instances and compute the optimal solution by GUROBI;
option = 2 is to use generated instances to compute CIOP for UFL;
option = 3 is to use generated instances to compute CIOP for UFL with range.