Facultat Informàtica de Barcelona

Algorithmic Methods for Mathematical Modeling
AMMM(MIRI)

AMMM Project

Authors:
Adrian Rodriguez Bazaga,
Pau Rodriguez Esmerats

December 29, 2017

Contents

1 Introduction

Write the context of the Project: mixed integer linear programming (SIMPLEX and branch and cut) and metaheuristics algorithms to model and solve mathematical problems.

The motivation: to model, implement and test each of the models applied to a problem.

Introduce the problem (copy from assignment).

Explain the outcomes of the project: 3 models of the problem, time comparison of MILP versus Metaheuristics, Comparison of the 2 Metaheuristics.

2 Models

2.1 ILP Model

This section explains how the Mixed Integer LP model has been designed and impemented according to the constraints of the problem (see Introduction for more details). First, the mathematical constraints that allows us to build a SIMPLEX problem instance are explained. Second, the code used to implement the model in ILOG script is presented. Finally, we present a brief explanation of the tests that have been performed on the model to determine its correctness.

2.1.1 Mathematical formulation of the model

Decision vars

- $w_{n,h}(\mathbb{B})$: whether the nurse n works at the hour h
- $z_n(\mathbb{B})$: whether the nurse n works during the shift(24h) or not
  - $\star$ $zn = 1 \Rightarrow$ The nurse n works at least 1 hour, $\exists h, w_{n,h} = 1$
  - $\star$ $zn = 0 \Rightarrow \forall h, w_{n,h} = 0$
- $s_n(\mathbb{N})$: hour in which the nurse n starts working,
  such that $w_{n,s_n} = 1$ and $w_{n,s_n-i} = 0$, $\forall i : 1 \leq s_n - i < s_n$
- $e_n(\mathbb{N})$: hour in which the nurse n stops working,
  such that $w_{n,e_n} = 1$ and $w_{n,e_n+i} = 0, \forall i : e_n < e_n + i \leq 24$

Known instance variables

- $demand_h$
- $nNurses$
- $minHours$
- $maxHours$
- $maxConsec$
- $maxPresence$

Objective function Min: $\sum_{n=1}^{nNurses} z_n$

Constraints

- set the zn values correctly: $\forall n : 1 \leq n \leq nNurses$,
  $24 \cdot z_n \geq \sum_{1 \leq h \leq 24} w_{n,h}$
  $z_n \leq \sum_{1 \leq h \leq 24} w_{n,h}$
- At any hour h, at least demandh nurses should be working:
  $\forall h : 1 \leq h \leq 24$,
  $\sum_{1 \leq n \leq nNurses} w_{n,h} \geq demand_h$
- Each nurse that works, should work at least minHours:
  $\forall n : 1 \leq n \leq nNurses$
  $\sum_{1 \leq h \leq 24} w_{n,h} \geq minHours \cdot z_n$
- Each nurse that works, should work at most maxHours:
  $\forall n : 1 \leq n \leq nNurses$
  $\sum_{1 \leq h \leq 24} w_{n,h} \leq maxHours \cdot z_n$

- Each nurse works at most maxConsec consecutive hours:
  $\forall n : 1 \le n \le nNurses,$
  $\forall h_1 : 1 \le h_1 \le 24 - maxConsec,$
  $\sum_{h_1 \le h \le h_1 + maxConsec} w_{n,h} \le maxConsec$

- Each nurse can stay in the hospital at most maxPresence hours:
  $\forall n : 1 \le n \le nNurses, \forall h : 1 \le h \le 24, e_n \ge h \cdot w_{n,h}$
  $\forall n : 1 \le n \le nNurses, s_n \ge 0$
  $\forall n : 1 \le n \le nNurses, \forall h : 1 \le h \le 24, s_n \le (h - 24) \cdot w_{n,h} + 24 \cdot z_n$
  $\forall n : 1 \le n \le nNurses, e_n - s_n + 1 - (2 \cdot 24) \cdot (1 - z_n) \le maxPresence \cdot z_n$

- Each nurse can rest at most one consecutive hour (exam hint version):
  $\forall n : 1 \le n \le nNurses, \forall h : 2 \le h \le 23 : r_{n,h} = 1 - w_{n,h}$
  $\forall n : 1 \le n \le nNurses, \forall h : 2 \le h \le 23 : wa_{n,h} = w_{n,h+1}$
  $\forall n : 1 \le n \le nNurses, \forall h : 2 \le h \le 23 : wb_{n,h} = w_{n,h-1}$
  $\forall n : 1 \le n \le nNurses, \forall h : 2 \le h \le 23, \forall M : M \ge 24$
  $M \cdot (1 - r_{n,h}) + M \cdot wb_{n,h} - 24 \cdot wa_{n,h} + 24 \cdot r_{n,h} \ge \sum_{1 \le h_i \le h} w_{n,h_i}$
  which is equal to :
  $2 \cdot M \cdot (1 - r_{n,h}) + M \cdot wb_{n,h} - M \cdot wa_{n,h} + M \cdot r_{n,h} \ge \sum_{1 \le h_i \le h} w_{n,h_i}$

- Each nurse can rest at most one consecutive hour:
  $\forall n : 1 \le n \le nNurses, \forall h : 2 \le h \le 22, \forall M : M \ge 24$
  $M - M \cdot w_{n,h-1} + M \cdot w_{n,h} + M \cdot w_{n,h+1} \ge \sum_{h+1 \le h_i \le 24} w_{n,h_i}$
  can be rewritten as :
  $\forall n : 1 \le n \le nNurses, \forall h : 2 \le h \le 22, \forall M : M \ge 24$
  $M - M \cdot wb_{n,h} + M \cdot w_{n,h} + M \cdot wa_{n,h} \ge \sum_{h+1 \le h_i \le 24} w_{n,h_i}$
  $M - M \cdot wb_{n,h} + M \cdot (1 - r_{n,h}) + M \cdot wa_{n,h} \ge \sum_{h+1 \le h_i \le 24} w_{n,h_i}$
  $M \cdot (2 - r_{n,h}) - M \cdot wb_{n,h} + M \cdot wa_{n,h} \ge \sum_{h+1 \le h_i \le 24} w_{n,h_i}$
  $2 \cdot M \cdot (1 - r_{n,h}) - M \cdot wb_{n,h} + M \cdot wa_{n,h} + M \cdot r_{n,h} \ge \sum_{h+1 \le h_i \le 24} w_{n,h_i}$
  $2 \cdot M \cdot (1 - r_{n,h}) - M \cdot wb_{n,h} + M \cdot wa_{n,h} + M \cdot r_{n,h} \ge \sum_{h \le h_i \le 24} w_{n,h_i}$

2.1.2 ILOG script implementation of the ILP model

2.1.3 Testing the ILP model

2.2 GRASP model

A brief context of the GRASP:

- acronym
- building blocks / components
- intensification, diversification and strategy applied

2.2.1 GRASP model pseudocode

This paragraph shows the pseudocode of our GRASP model for the problem of interest.

After the pseudocode is shown, brief comments of each line should be added.

2.2.2 GRASP model test

Tests performed to determine if the model is correct. Results obtained (very brief)

2.3 BRKGA model

A brief context of the BRKGA metaheuristic algorithm:

- acronym
- building blocks / components
- intensification, diversification and strategy applied

2.3.1 BRKGA model pseudocode

This paragraph shows the pseudocode of our BRKGA model for the problem of interest.

After the pseudocode is shown, brief comments of each line should be added.

2.3.2 BRKGA model test

Tests performed to determine if the model is correct. Results obtained (very brief)

3 Tests

3.1 Instance generation

This section shows how the instances have been generated. Possibly adding a very schematic pseudocode of our generation algorithm.

This section also explains the sets of instances that are prepared for the project:

- small-medium set: used to compare the solving time of the ILP model versus the Metaheuristics models
- large set: used to test the Metaheuristics models parameters over a large set of instances in order the choose the best performing setup for each model.

3.1.1 Instance generation

pseudocode
comments
how feasibility is assured
tests on how to increment solving time, including some graphs with the Best Integer and Best Bound evolution for different variations of the same instance
Conclusion on how to increase the "size" of the instance without computing the solving time of the ILP model.

3.1.2 Medium Set

Composition of the set, number of instances and solving times.

3.1.3 Large Set

Composition of the set, number of instances, solving times or another measure of the size of the problem.

3.2 Comparison of ILP and Metaheuristics models

Setup of the test:

- instances set
- execution conditions (same computer)
- models parameters
- results and times gathering procedure

Results

Chart of Solving times for all the instance set for the three models

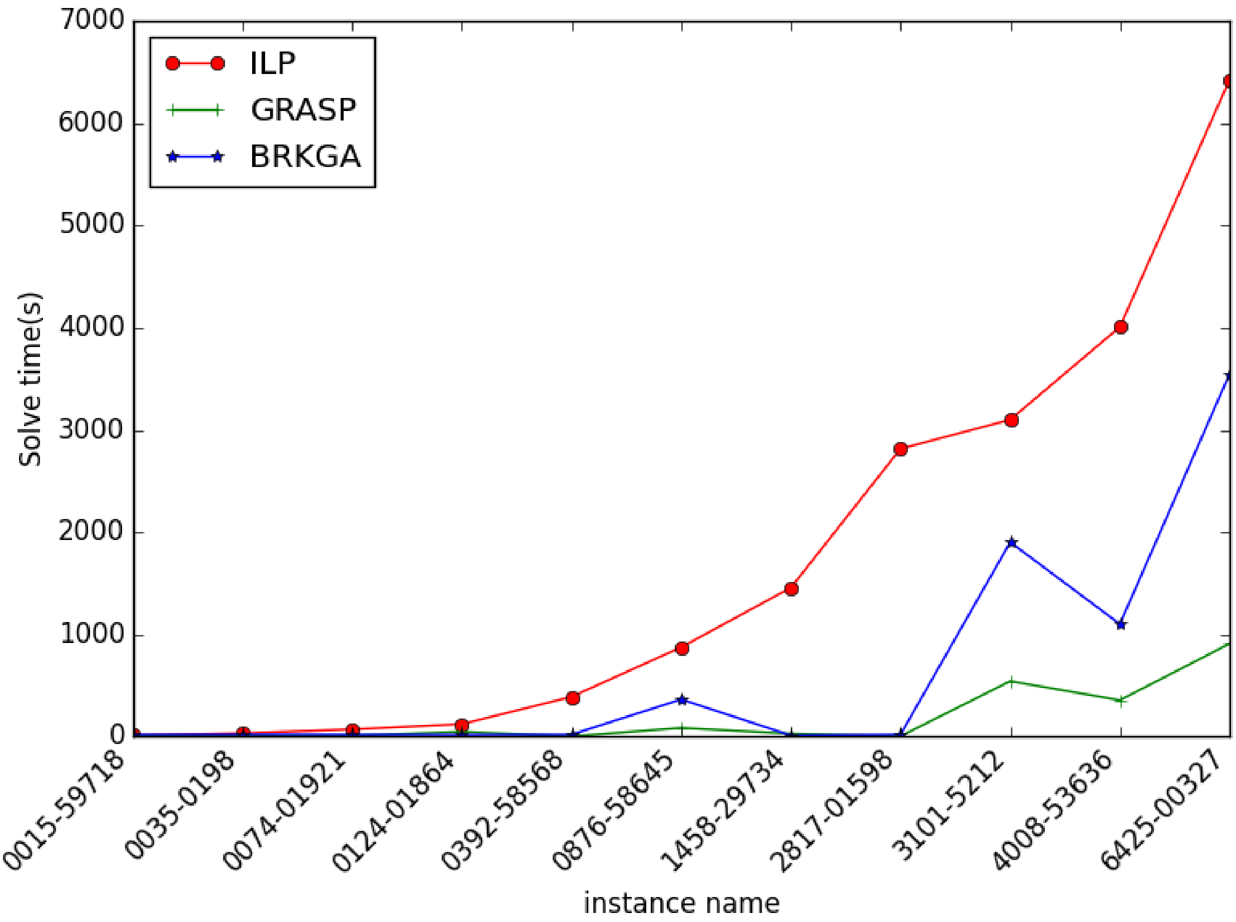
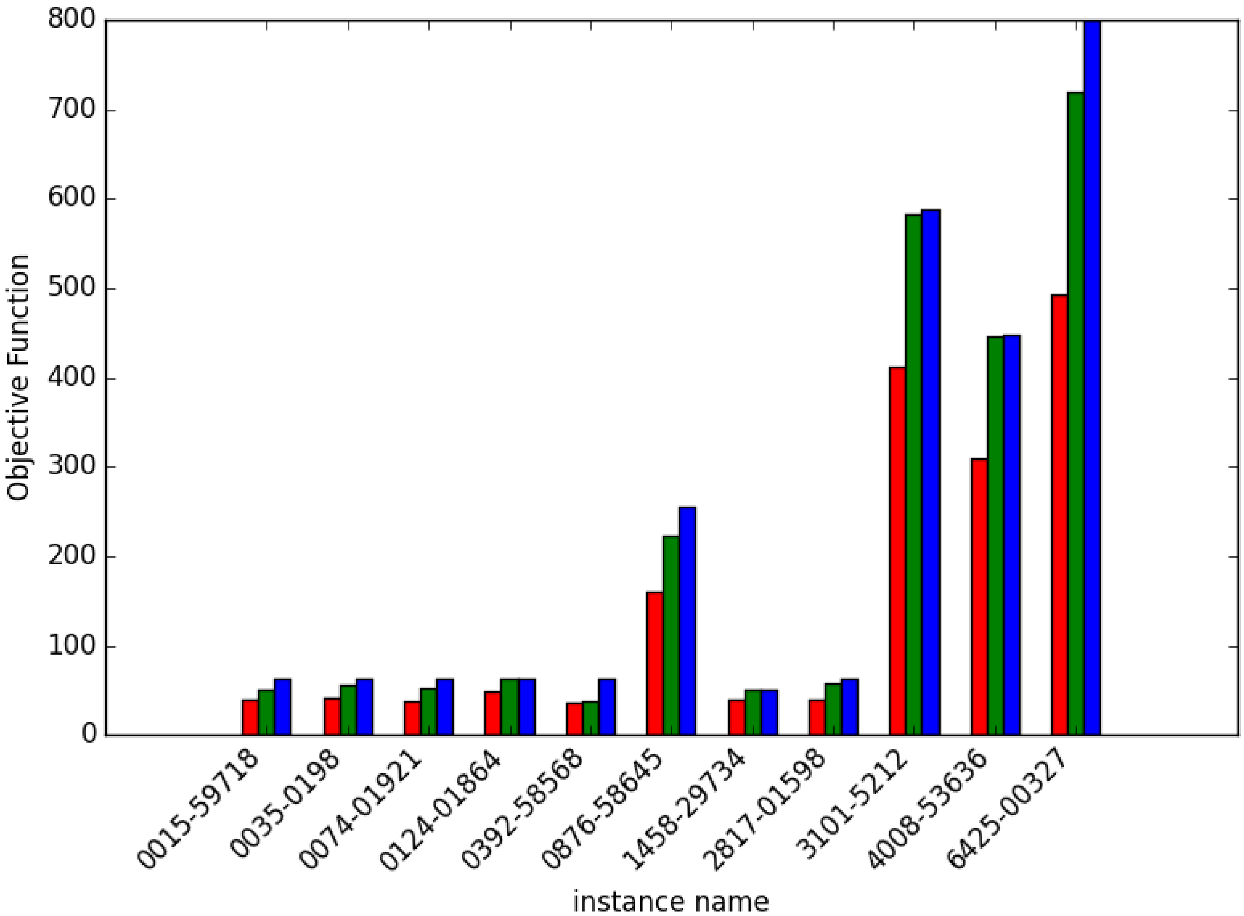


Chart of Objective function value for all the instance set for the three models


Objective Function
0
100
200
300
400
500
600
700
800
0015-59718
0035-0198
0074-01921
0124-01864
0392-58568
0876-58645
1458-29734
2817-01598
3101-5212
4008-53636
6425-00327
instance name

3.3 Comparison of GRASP and BRKGA models

Test of the section: parameter selection for each model, model performance comparision with best parameter setup.

3.3.1 GRASP parameter selection

Setup of the tests:

- instances set
- execution conditions (same computer)
- procedure
- results and times gathering procedure

Graph of alpha
Graph of Iterations
Graph of best improvement vs first improvement?

3.3.2 BRKGA parameter selection

Setup of the tests:

- instances set
- execution conditions (same computer)
- procedure
- results and times gathering procedure

Graph of iheritance probability
Graph of elite proportion
Graph of mutant proportion
Graph of population size
Graph of number of generations

3.3.3 Metaheuristic models comparision

Choosing the best performing parameter setup of the two models, perform a comparison of how objective function evolves in relation to time.

Draw conclusions about, why one is better than the other, why one stops sooner than the other. (think in terms of diversification and intensification and solution space exploration)

4 Conclusion

The conclusion may contain facts discovered throught the comparision of solving times. Thoughs about intensification and diversification of the algorithms. Parametrizations that work and why the do, etc..