Inspira Crea Transforma
The International Spring School on Integrated Operational Problems: What a PhD student can learn

Juan David Palacio Domínguez
PhD student on Mathematical Engineering - EAFIT

Doctoral Seminar on Mathematical Engineering - August 3, 2018
Outline

• What is the International Spring School on Integrated Operational Problems – ISSIOP?
• About Troyes in France
• Research and lectures at the ISSIOP
• My schedule at the ISSIOP
• Personal experience and acknowledgements
What is the ISSIOP?

It is the fourth edition of the GdR RO Young Researchers School, organized in collaboration with 4 working groups (GT2L, GT Bermuda, GT Meta, GT OSI)

In 2018 the school takes an international dimension: all the courses are made by scientists from several countries (Spain, Belgium, France).
What is the ISSIOP?
About Troyes in France
About Troyes in France
About Troyes in France
Research and lectures at ISSIOP

Ch. Prins
UTT

J. Billaut
Université F. Rabelais (Tours)

A. Corberán
Universidad de Valencia

D. Feillet
E. Mines Saint-Etienne

Marc Sevaux
Université de Bretagne-Sud

Kenneth Sörensen
Univ. Antwerpen
Research and lectures at ISSIOP

1. Split algorithms - Ch. Prins
2. Column generation – D. Feillet
3. Integrated scheduling and routing – J. Billaut
4. Introduction to web services – M. Sevaux
5. Optimization of Smart grids – E.G. Talbi
6. Internet of things - L. Merghem-Boulahia
7. The trustfull promises of block chain – Ph. Entzman
8. Constraint programming – E. Hebrard
9. Linear programming for routing - A. Corberán
10. Last advances in metaheuristics – K. Sörensen
My schedule at ISSIOP

Lectures

- Split algorithms
- Column generation
- Integrated scheduling and routing
- Constraint programming
- Linear programming for routing
- Last advances in metaheuristics

Projects

- Column generation
- Integrated scheduling and routing
Column generation lecture

Vehicle Routing Problem with Time Windows (VRPTW)

Column generation algorithm

Implementation remarks

Image taken from: http://fc.isima.fr/~lacomme/Spring_School/conf/slides/
Column generation lecture

minimize $\sum_{1 \leq u \leq U} \sum_{(v_i, v_j) \in A} c_{ij} x_{ij}^u$

subject to

$\sum_{v_j \in V \backslash \{v_0\}} x_{ij}^u = 1 \quad (v_i \in V \backslash \{v_0\})$,

$\sum_{v_j \in V \backslash \{v_i, v_0\}} x_{ij}^u = 0 \quad (v_i \in V, 1 \leq u \leq U)$,

$\sum_{v_i \in V \backslash \{v_0, v_0\}} x_{0i}^u \leq 1 \quad (1 \leq u \leq U)$,

$\sum_{(v_i, v_j) \in A} d_i x_{ij}^u \leq Q \quad (1 \leq u \leq U)$,

$s_i^u + st_i + c_{ij} - s_j^u + M x_{ij}^u \leq M \quad ((v_i, v_j) \in A, v_j \neq v_0, 1 \leq u \leq U)$,

$s_i^u + st_i + c_{i0} - b_0 + M x_{i0}^u \leq M \quad ((v_i, v_0) \in A, 1 \leq u \leq U)$,

$a_i \leq s_i^u \leq b_i \quad (v_i \in V, 1 \leq u \leq U)$,

$x_{ij}^u \in \{0, 1\} \quad ((v_i, v_j) \in A, 1 \leq u \leq U)$,

The set of all feasible routes

minimize $\sum_{r_k \in \Omega} c_k \theta_k$

subject to

$\sum_{r_k \in \Omega} a_{ik} \theta_k \geq 1 \quad (v_i \in V \backslash \{v_0\})$,

$\sum_{r_k \in \Omega} \theta_k \leq U$,

$\theta_k \in \mathbb{N}$

(r_k \in \Omega).

Image taken from: http://fc.isima.fr/~lacomme/Spring_School/conf/slides/
Column generation lecture

Linear relaxation of the extended formulation

Initial set $\Omega_1$
(t=1) → Solve $\text{MP}(\Omega_t)$
(e.g., by simplex)

A new route $r_k \in \Omega \setminus \Omega_t$ needs to be added?

YES
$\Omega_{t+1} \leftarrow \Omega_t \cup \{\theta_k\}$
t $\leftarrow$ t+1

NO → Stop

Image taken from: http://fc.isima.fr/~lacomme/Spring_School/conf/slides/
Integrated scheduling and routing lecture

Basics on scheduling and routing

Integrated problem

GLPK (for a MILP implementation)

Python (for a metaheuristic implementation)
Integrated scheduling and routing lecture

1- Scheduling problem

2- Batching problem

3- Routing problem for each batch

Image taken from: http://fc.isima.fr/~lacomme/Spring_School/conf/slides/
Integrated scheduling and routing lecture

```plaintext
# scheduling

s.t. contr_Ck_1 {i in 0..m-1, j1 in 0..n-1, j2 in 0..n-1; j1!=j2}:
    # if j1 precedes j2
    C[i,j2] >= C[i,j1] + p[i,j2] - BigM*(1-y[j1,j2]);

s.t. contr_Ck_2 {j1 in 0..n-1, j2 in 0..n-1; j1!=j2}:
    # j1 before j2 or j2 before j1
    y[j1,j2]+y[j2,j1]=1;

s.t. contr_Ck_2b {i in 1..m-1, j in 0..n-1}:
    # "routing" constraint
    C[i,j] >= C[i-1,j] + p[i,j];

s.t. contr_Ck_2a {j in 0..n-1}:
    # first job
    C[0,j] >= p[0,j];

s.t. def_TjM {j in 0..n-1}:
    # tardiness expression
    Tj[j] >= C[m-1,j] - d[j];

end;

MILP implementation for the scheduling part using GLKP

```
Integrated scheduling and routing

```python
# Parameters
# n jobs, m machines, k=nbjobs
nbjobs=len(pp)
INFINI = 999999999
TabuList=[]
TabuSize = 7
TIME_LIMIT = nbjobs * m / 4
DELTA = nbjobs/2 # is used to limit the swaps
FLAG_SWAP_BOTH = 1
FLAG_SWAP_SEQ = 1
FLAG_SWAP_BATCH = 1
FLAG_2OPT = 0

# ==

def InsertTabu(voisinage,indi,indj):
    ElemTabu=[voisinage,indi,indj]
    if len(TabuList) == TabuSize:
        del TabuList[0]
    TabuList.append(ElemTabu)

def NotTabu(voisinage,indi,indj):
    ElemTabu=[voisinage,indi,indj]
    notTabu = True
    if ElemTabu in TabuList:
        notTabu = False
    return(notTabu)

# ==

# Neighborhood
# ==

def swap_both(i,j,sol):
    # the sequence sol is modified by a swap both in the sequence and in the batches
    #print('before swap_seq:',i,j,sol)
    job_i = sol[0][i]
    job_1 = [job_i]
    job_j = sol[0][j]
    job_2 = [job_j]
    sol[0]=sol[0][0:i]+job_2+sol[0][i+1:j]+job_1+sol[0][j+1:nbjobs]
    u=0
```

---

**Inpira Crea Transforma**

---

**Universidad EAFIT**

**Vigilada Mineducación**
Split algorithms lecture

Route-first cluster-second methods

Basic splitting procedure

Applications to heuristics and metaheuristics.
Split algorithms lecture

1. Giant tour $T = (1, 2, 3, 4, 5)$ with demands

2. Auxiliary graph $H$ of possible trips for $Q = 10$ – Shortest path in bold

3. Optimal splitting, cost 205

Image taken from: http://fc.isima.fr/~lacomme/Spring_School/conf/slides/
Lp-based methods for routing problems

lecture

Polyhedral combinatorics applied to:

- TSP
- OARP
- Close-enough ARP

Lp-based methods for solving routing problems

Ángel Corberán

Universitat de València, Spain

Spring School on Integrated Operational Problems
May 14-16, Troyes, France

Image taken from: http://fc.isima.fr/~lacomme/Spring_School/conf/slides/
Lp-based methods for routing problems lecture
Last advances in metaheuristics lecture

Complementary local search operators

Properties of good solutions

Solution metrics
Last advances in metaheuristics lecture

State of the art

- Use as many local search (constructive) operators as possible
- Either VNS or LNS
- Fit in a metaheuristic framework
  - This is your Unique Selling Point
  - But it really does not matter all that much
- Beware of “Frankenstein” algorithms

Image taken from: http://fc.isima.fr/~lacomme/Spring_School/conf/slides/
Last advances in metaheuristics lecture

Image taken from: http://fc.isima.fr/~lacomme/Spring_School/conf/slides/
Personal experience and acknowledgements
The International Spring School on Integrated Operational Problems: What a PhD student can learn

Juan David Palacio Domínguez
PhD student on Mathematical Engineering - EAFIT

Doctoral Seminar on Mathematical Engineering - August 3, 2018