Modeling Patterns for LocalSolver

T. Benoist, J. Darlay, B. Estellon, F. Gardi, R. Megel

LocalSolver

Who we are

BOUYGUES

Large industrial group with businesses in construction, telecom, media

www.bouygues.com

Innovation24

Operation Research subsidiary of the Bouygues group

www.innovation24.fr

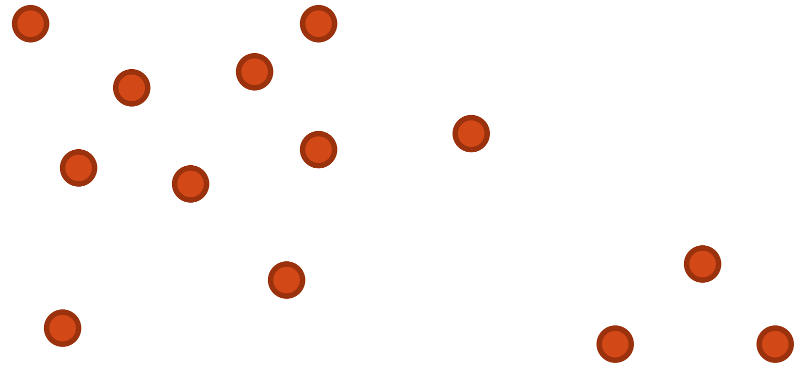
LocalSolver

Flagship product of Innovation 24

www.localsolver.com

LocalSolver in one slide

Select a set S of P cities among N
Minimizing the sum of distances
from each city to the closest city
in S



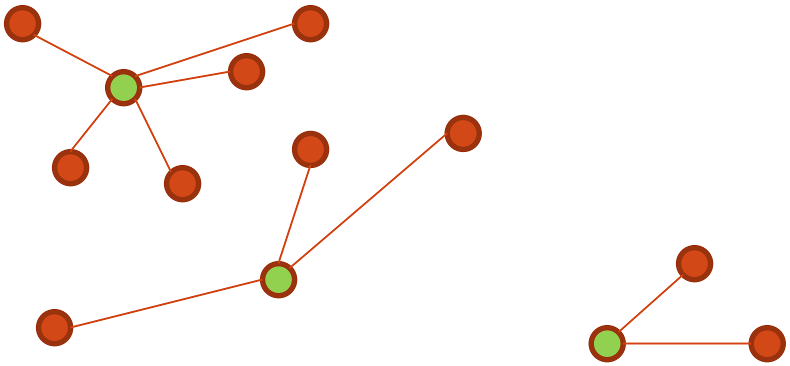
```
function model() {
    x[1..N] <- bool();
    constraint sum[i in 1..N] (x[i]) == P;

    minDistance[i in 1..N] <- min[j in 1..N]  (x[j] ? distance[i][j] : +inf);
    minimize sum[i in 1..N] (minDistance[i]);
}
```

Results on the OR Library

- 28 optimal solutions on the 40 instances of the OR Lib
- an average gap of 0.6%
- with 1 minute per instance

LocalSolver in one slide

Select a set S of P cities among N
Minimizing the sum of distances
from each city to the closest city
in S



```
function model() {
    x[1..N] <- bool();
    constraint sum[i in 1..N] (x[i]) == P;

    minDistance[i in 1..N] <- min[j in 1..N]  (x[j] ? distance[i][j] : +inf);
    minimize sum[i in 1..N] (minDistance[i]);
}
```

Results on the OR Library

- 28 optimal solutions on the 40 instances of the OR Lib
- an average gap of 0.6%
- with 1 minute per instance

LocalSolver in one slide

An hybrid math
programming solver

For large-scale mixed-variable
non-convex optimization problems

Providing high-quality solutions
in short running time
without any tuning

Outline

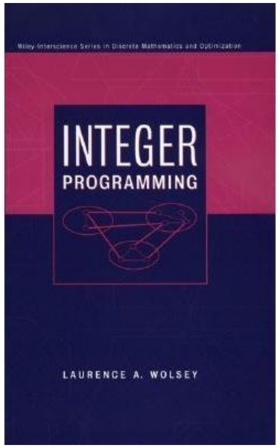
- LocalSolver
- Modeling Patterns for Local Search ?
- Six Modeling Patterns

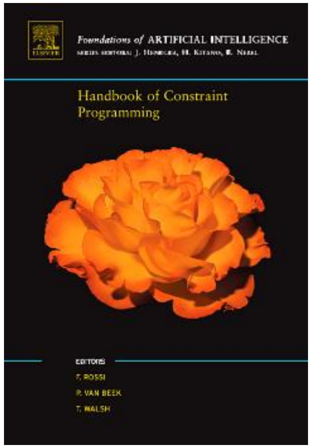
Modeling Patterns for LocalSolver

Why ?

Modeling patterns ?

A classic topic in MIP or CP





Very little literature on modeling for Local Search...

...**because of** the absence of model-and-run solver

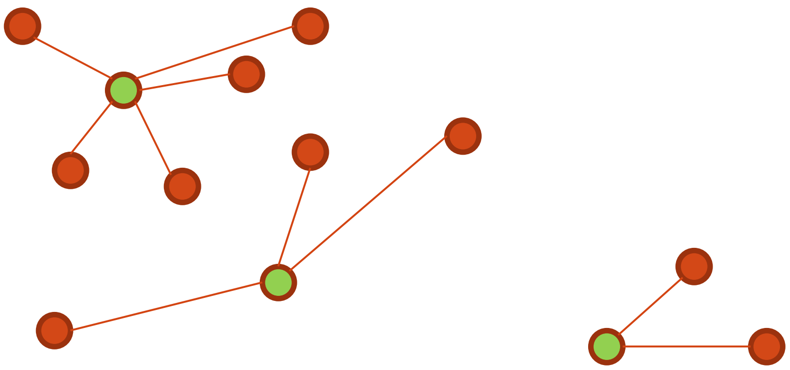
→ models and algorithms were designed together and not always clearly separated

Modeling Pattern #1

Choose the right set of decision variables

Choose the right set of decision variables

Select a set S of P cities among N
Minimizing the sum of distances
from each city to the closest city
in S



```
function model() {
    x[1..N] <- bool();
    constraint sum[i in 1..N] (x[i]) == P;

    minDistance[i in 1..N] <- min[j in 1..N]  (x[j] ? distance[i][j] : +inf);
    minimize sum[i in 1..N] (minDistance[i]);
}
```

Results on the OR Library

- 28 optimal solutions on the 40 instances of the OR Lib
- an average gap of 0.6%
- with 1 minute per instance

Modeling Pattern #2

Precompute what can be precomputed

Precompute what can be precomputed

Document processing : dans un tableau une case de texte a plusieurs configurations *hauteur x largeur* possibles.

LocalSolver : mathematical programming by local search

29 x 82

LocalSolver : mathematical programming by local search

34x 61

LocalSolver : mathematical programming by local search

45x 43

Comment choisir la configurations de chaque case de façon à minimiser la hauteur du tableau (sa largeur étant limitée) ?

UNIVERSITY of LIMERICK

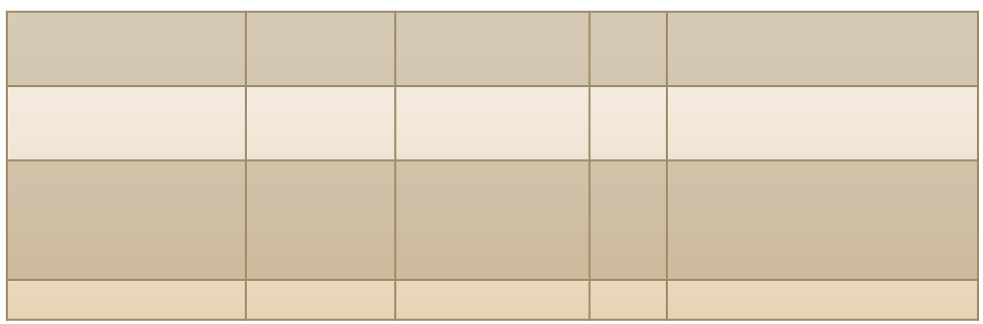
OLLSCOIL LUIMNIGH

Precompute what can be precomputed

Première modélisation : 1 variable de décision par configuration (largeur, hauteur) possible pour chaque cellule

Formulation étendue :

- On remarque qu'à partir de la largeur d'une colonne on peut déterminer la hauteur minimum de chacune de ses cellules.
- 1 variable de décision par largeur possible pour chaque colonne
- Conséquence : en changeant une variable de décision, LocalSolver va changer la hauteur et la largeur de toutes les cellules dans la colonne



-> R. Megel (Roadef 2013).
Modélisations LocalSolver de type « génération de colonnes » .

LocalSolver

Modeling Pattern #3

Do not limit yourself to linear operators

Do not limit yourself to linear operators

TRAVELING SALESMAN PROBLEM

MIP approach: X_{ij}=1 if city j is after city i in the tour

- Matching constraints $\sum_j X_{ij} = 1$ and $\sum_i X_{ij} = 1$
- Plus an exponential number of subtour elimination constraints
- Minimize $\sum_{ij} c_{ij} X_{ij}$

Polynomial non-linear model: : X_{ik}=1 if city *i* is in position *k* i in the tour

- Matching constraints $\sum_k X_{ik} = 1$ and $\sum_i X_{ik} = 1$
- $Y_k \leftarrow \sum_i i X_{ik}$ the index of the k^{th} city of the tour
- Minimize $\sum_k c_{[Y_k, Y_{k+1}]}$

"at" operator

TSP Lib: average gap after 10mn = 2.6%

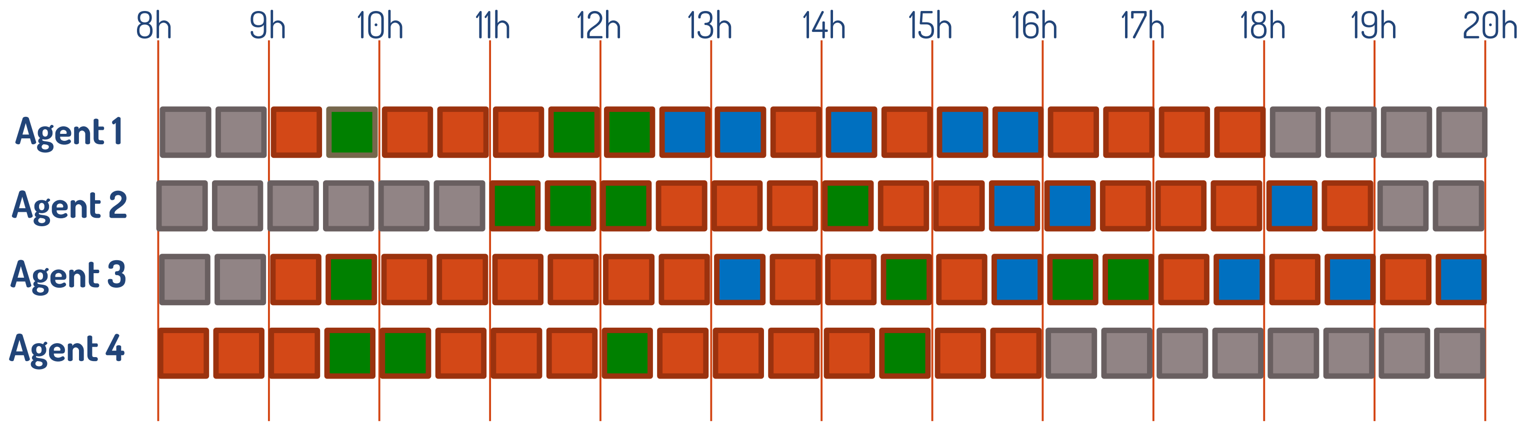
LocalSolver

Modeling Pattern #4

Separate commands and effects

Separate commands and effects

Multi-skill workforce scheduling



Candidate model

$\text{Skill}_{atk} = 1 \Leftrightarrow$ agent a works on skill k at timestep t

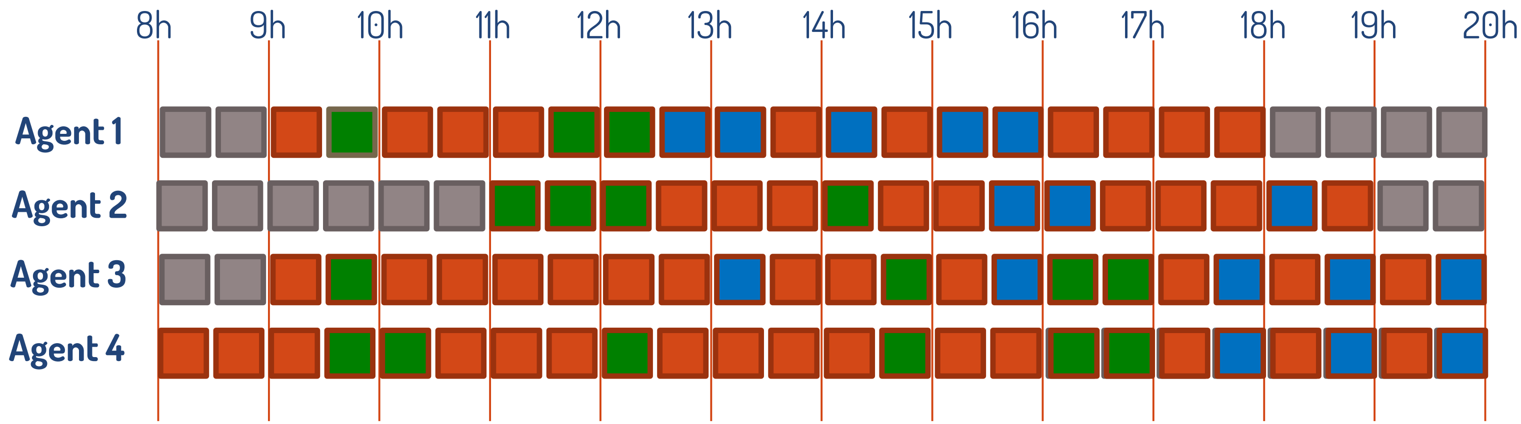
Constraint $SUM_k\,(\text{Skill}_{atk}) <= 1$

Constraint $OR_k\,(\text{Skill}_{atk}) == \ (t \in [\text{Start}_a, \text{End}_a[\ )$

Problem: any change of $Start_a$ will be rejected unless skills are updated for all impacted timesteps

Separate commands and effects

Multi-skill workforce scheduling



Alternative model

$SkillReady_{atk} = 1$ ⇔ agent *a* will works on skill *k* at timestep *t* **if present**

Constraint SUM_k $(SkillReady_{atk}) == 1$

$Skill_{atk} \leftarrow AND(SkillReady_{atk}\ ,\ t \in [Start_a, End_a[)$

Now we have no constraint between skills and worked hours

-> for any change of $Start_a$ skills are automatically updated

Separate commands and effects

Similar case: Unit Commitment Problems

- A generator is active or not, but when active the production is in $[P_{min}, P_{max}]$
- Better modeled without any constraint

$$\begin{aligned}
&ProdReady_{gt} \leftarrow float(P_{min}, P_{max}) \\
&Active_{gt} \leftarrow bool() \\
&Prod_{gt} \leftarrow Active_{gt} \times ProdReady_{gt}
\end{aligned}$$

Modeling Pattern #5

Invert constraints and objectives ?

Invert constraints and objectives

Clément Pajean	Vendredi 28
Modèle LocalSolver d'ordonnancement d'une machine unique sous contraintes de Bin Packing	14h Bât B TD 35

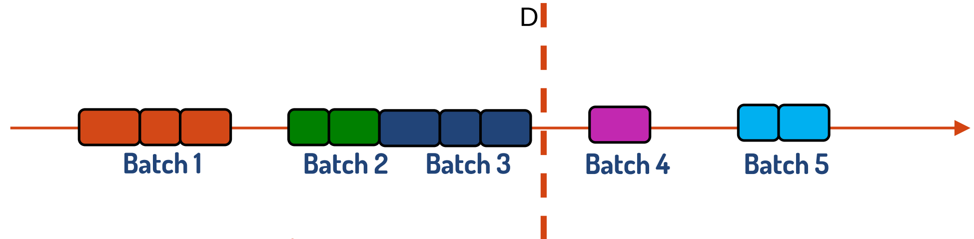
Modeling Pattern #6

Use dominance properties

Use dominance properties

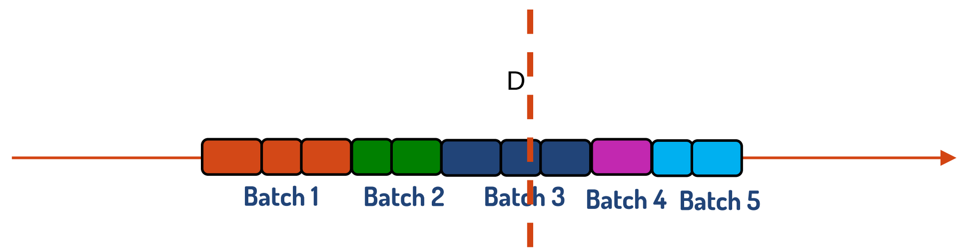
Batch scheduling for *N* jobs having the same due date D.

- Completion time of each job will be that of the batch selected for this job
- Linear late or early cost $(\alpha_k\ \beta_k)$



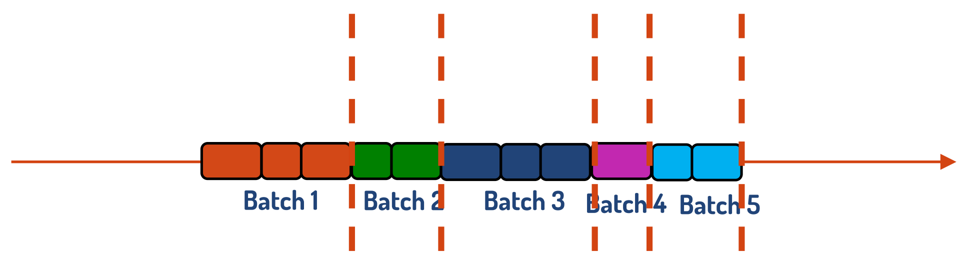
Basic Model

Date variable for each batch
+ assignment of jobs to batches
+ Precedence constraints



No idle time

Only one starting date variable
+ assignment of jobs to batches



Optimal start will position due date at the end of a batch

No start date variable
+ assignment of jobs to batches
+ penalty[k] if due date at the end of batch k
Minimize min[k in 1..5](penalty[k])

We can **minimize a minimum**

As if starting date was automatically adjusted after each move

Summary

1. Choose the right set of decision variables
2. Precompute what can be precomputed
3. Do not limit yourself to linear operators
4. Separate commands and effects
5. Invert constraints and objectives ?
6. Use dominance properties

Modeling Pattern #7

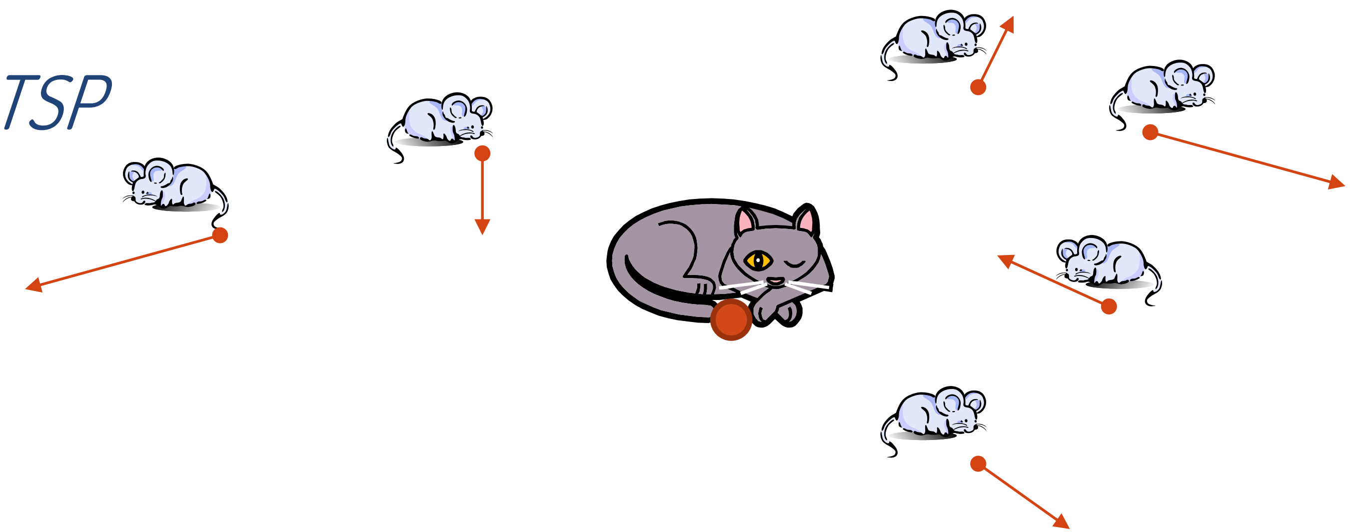
Your turn!

Why solving a TSP with LocalSolver ?

$$\mathrm{Time}[A,B,T] = \frac{2({\alpha_x}^2+{\alpha_y}^2)}{-2(\alpha_x\beta_x+\alpha_y\beta_y)+\sqrt{4(\alpha_x\beta_x+\alpha_y\beta_y)^2-4({\alpha_x}^2+{\alpha_y}^2)({\beta_x}^2+{\beta_y}^2-V^2)}} + T$$

With $\alpha_x, \beta_x, \alpha_y, \beta_y$ function of A,B and T

Kinetic TSP

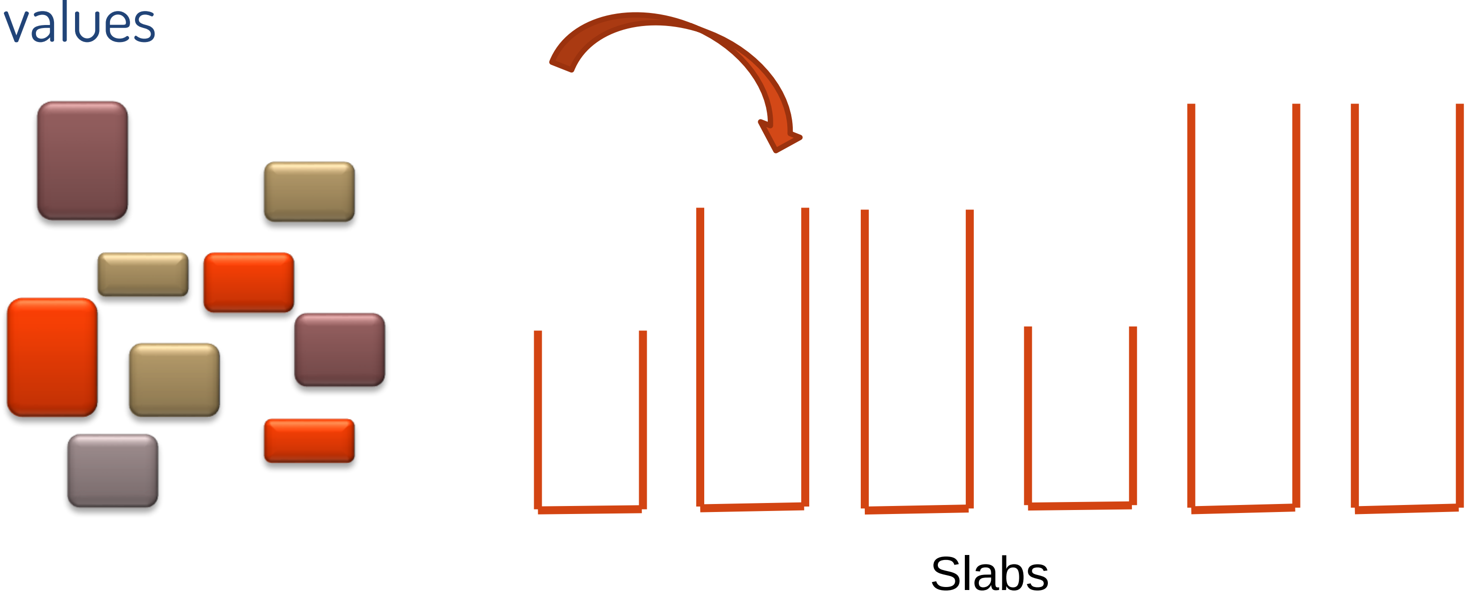


LocalSolver

Choose the right set of decision variables

Industrial « Bin-packing »

Assignment of steel orders to « slabs » whose capacity can take only 5 different values



Model

$X_{ij} = 1 \Leftrightarrow$ Order i is assigned to slab j

$Y_{jk} = 1 \Leftrightarrow$ Slab j takes capacity k

Minimize total size of slabs

Choose the right set of decision variables

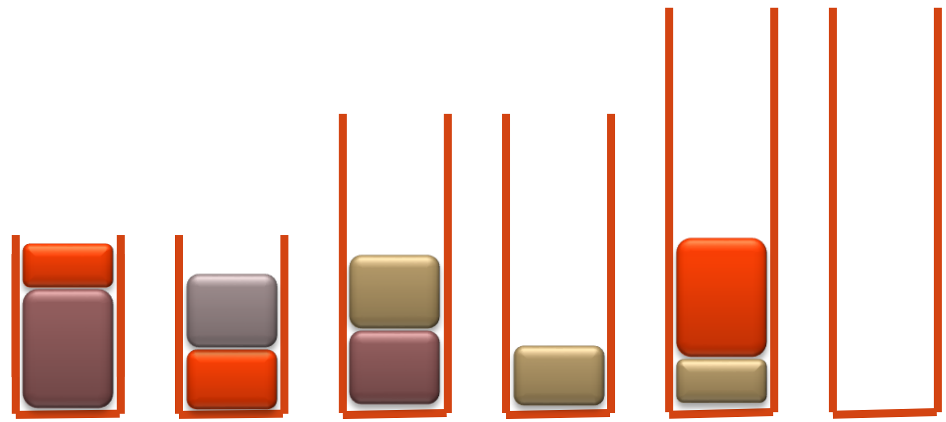
Industrial « Bin-packing »

Assignment of steel orders to « slabs » whose capacity can take only 5 different values

Change

Exchange

Ejection chain



Slabs

In a good model:

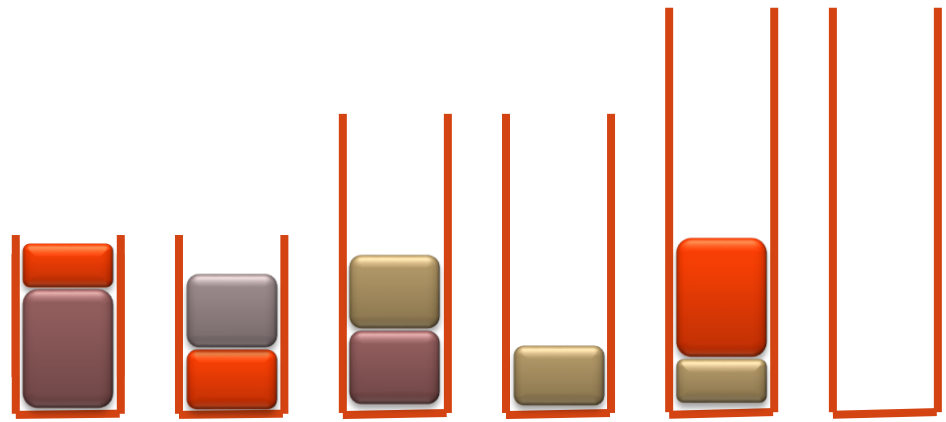
- when a value can be computed from others it is defined with operator <- (it is an **intermediate** variable)
- moving from a feasible solution to another feasible solution only requires modifying a small number of **decision** variables.

Choose the right set of decision variables

Industrial « Bin-packing »

Assignment of steel orders to « slabs » whose capacity can take only 5 different values

- Change
- Exchange
- Ejection chain



Slabs

Model

$X_{ij} = 1 \Leftrightarrow$ Order i is assigned to slab j

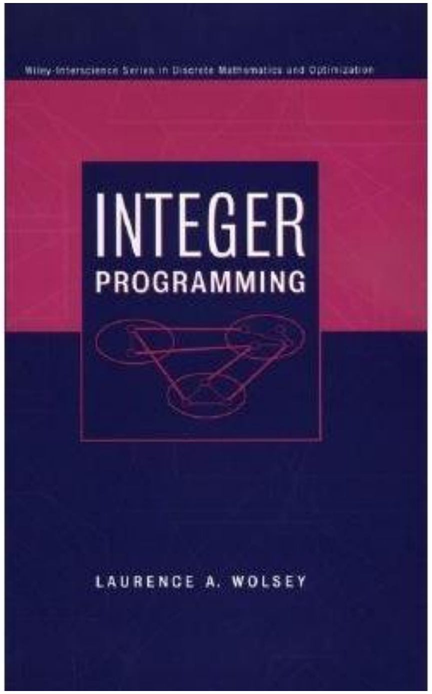
$Capa_k \leftarrow MinCapa[content_k]$

"at" operator

LocalSolver

Modeling patterns for Mixed-Integer Programming

- MIP requires linearizing non linear structures of the problem
- The polyhedron should be kept as close as possible to the convex hull -> valid inequalities, cuts, and so on
- Symmetries should be avoided (or not…)



Modeling patterns for constraint programming

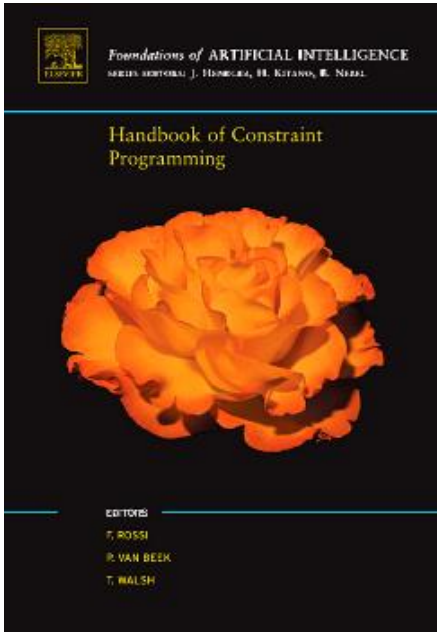
Choice of variables (integer, set variables, continuous...)

Choice of (global) constraints

Redundant constraints

Double point of view (with channeling constraints)

And so on



Modeling patterns for Local Search ?

Very little literature on this topic...

...because of the absence of model-and-run solver

→ models and algorithms were designed together and not always clearly separated

LocalSolver

mathematical programming by local search

www.localsolver.com