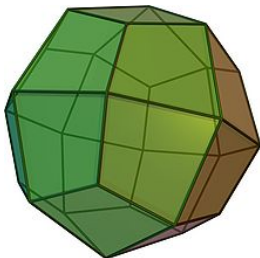


# Optimisation mathématique

Nicolas Bonifas (nicolas.bonifas@polytechnique.edu)

Prologuin 2015



- Ancien candidat Prologin et IOI

- Ancien candidat Prologin et IOI
- Thèse en optimisation mathématique, en collaboration avec ILOG (startup, rachetée par IBM en 2009).

- Ancien candidat Prologis et IOI
- Thèse en optimisation mathématique, en collaboration avec ILOG (startup, rachetée par IBM en 2009).
- Qu'est-ce que l'optimisation mathématique ?

- Ancien candidat Prologin et IOI
- Thèse en optimisation mathématique, en collaboration avec ILOG (startup, rachetée par IBM en 2009).
- Qu'est-ce que l'optimisation mathématique ?
- Ou : comment tout a commencé par de la programmation dynamique.

# Bloqué dans un train



- 
- Pourquoi la SNCF n'emploie-t-elle pas des algorithmiciens pour optimiser les trajets ?

# Bloqué dans un train



- 
- Pourquoi la SNCF n'emploie-t-elle pas des algorithmiciens pour optimiser les trajets ?
- Ça existe déjà !

# Bloqué dans un train



- 
- Pourquoi la SNCF n'emploie-t-elle pas des algorithmiciens pour optimiser les trajets ?
- Ça existe déjà !
- Aide à la décision / Recherche opérationnelle



# Bloqué dans un train



- 
- Pourquoi la SNCF n'emploie-t-elle pas des algorithmiciens pour optimiser les trajets ?
- Ça existe déjà !
- Aide à la décision / Recherche opérationnelle
- Rationaliser la prise de décisions opérationnelles et industrielles

# Bloqué dans un train



- 
- Pourquoi la SNCF n'emploie-t-elle pas des algorithmiciens pour optimiser les trajets ?
- Ça existe déjà !
- Aide à la décision / Recherche opérationnelle
- Rationaliser la prise de décisions opérationnelles et industrielles
- Méthodes : statistiques, théorie des jeux, simulation, **optimisation mathématique, ...**

- Description d'un problème:
  - Variables de décision
  - Contraintes entre les variables
  - Objectif

# Optimisation mathématique

- Description d'un problème:
  - Variables de décision
  - Contraintes entre les variables
  - Objectif
- Résolution automatique (sans écrire de code !)

# Optimisation mathématique

- Description d'un problème:
  - Variables de décision
  - Contraintes entre les variables
  - Objectif
- Résolution automatique (sans écrire de code !)
- Méthodes :
  - Programmation linéaire
  - Programmation par contraintes
  - Heuristiques contrôlées

# Optimisation mathématique

- Description d'un problème:
  - Variables de décision
  - Contraintes entre les variables
  - Objectif
- Résolution automatique (sans écrire de code !)
- Méthodes :
  - Programmation linéaire
  - Programmation par contraintes
  - Heuristiques contrôlées
- Résolution pratique de problèmes NP-difficiles

# Exemple

```
1 sched_production.mod
2
3 tuple JobData {
4   JobIndex ind;
5   JobInfo info;
6 };
7 (JobData) jobs = ...;
8
9 (JobIndex) joblist = { i | <i,j> in jobs };
10 assert ( card(joblist) == card(jobs) );
11
12 JobInfo datarray[joblist];
13
14 execute {
15   for(var j in jobs)
16     datarray[j.ind] = j.info;
17 };
18
19 dvar interval a[j in joblist]
20   in datarray[j].arrival..datarray[j].weekdue
21   size datarray[j].duration;
22
23 cumulFunction reactors = sum (j in joblist) pulse(a[j], datarray[j].reactors);
24 cumulFunction columns = sum (j in joblist) pulse(a[j], datarray[j].columns);
25 cumulFunction centrifuges = sum (j in joblist) pulse(a[j], datarray[j].centrifuges);
26
27 minimize max(j in joblist) endOf(a[j]);
28 subject to {
29   forall (j in joblist) {
30     if (datarray[j].jobprec > 0) {
31       endBeforeStart(a[<j.ordernumber, datarray[j].jobprec>], a[j]);
32     }
33   }
34   reactors <= NbReactors;
35   columns <= NbColumns;
36   centrifuges <= NbCentrifuges;
37 };
38
39 execute {
40   for(var j in joblist) {
41     writeln(j + " on [" + a[j].start + ", " + a[j].end + ")");
42   }
43 }
```

Objective Function

Interval Decision Variable

Cumulative Functions

Scheduling Constraints

**Bon code !**

Pour toute question : [nicolas.bonifas@polytechnique.edu](mailto:nicolas.bonifas@polytechnique.edu)