LocalSolver

A new kind of mathematical optimization solver

Thierry Benoist, Bertrand Estellon, Julien Darlay,
Frédéric Gardi, Romain Megel, Clément Pajean

Innovation 24 & LocalSolver, Paris
www.localsolver.com

MIM 2016, Troyes

Who we are

Bouygues, one of the French largest corporation, €33 bn in revenues
http://www.bouygues.com

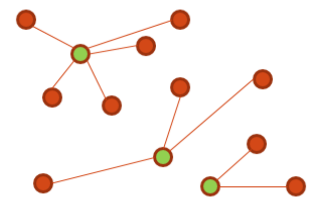
Operations Research subsidiary of Bouygues
20 years of practice and research
http://www.innovation24.fr

Mathematical optimization solver developed by Innovation 24
http://www.localsolver.com

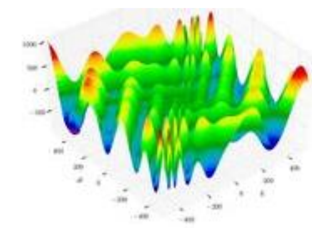
LocalSolver

All-terrain optimization solver

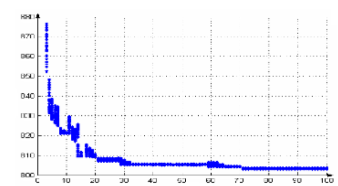
The « Swiss Army Knife » of mathematical optimization



For combinatorial, numerical, or mixed-variable optimization



Suited for tackling large-scale problems



Quality solutions in minutes without tuning

free trial with support – free for academics – rental licenses from 690 €/month – perpetual licenses from 9,900 €

www.localsolver.com

Features

Innovative heuristic search techniques

- Fast: provides high-quality solutions in minutes
- Scalable: able to tackle problems with millions of decisions

Easy to use

- « Model & Run »
  - Rich but simple mathematical modeling formalism
  - Direct resolution: no need of complex tuning
- Innovative modeling language for fast prototyping
- Object-oriented C++, Java, .NET, Python APIs for tight integration
- Fully portable: Windows, Linux, Mac OS (x86, x64)

P-median

Select a subset P among N points minimizing the sum of distances from each point in N to the nearest point in P

```
function model() {
  x[1..N] <- bool() ; // decisions: point i belongs to P if x[i] = 1
  constraint sum[i in 1..N]( x[i] ) == P ; // constraint: P points selected among N
  minDist[i in 1..N] <- min[j in 1..N]( x[j] ? Dist[i][j] : InfiniteDist ) ; // expressions: distance to the nearest point in P
  minimize sum[i in 1..N]( minDist[i] ) ; // objective: to minimize the sum of distances
}
```

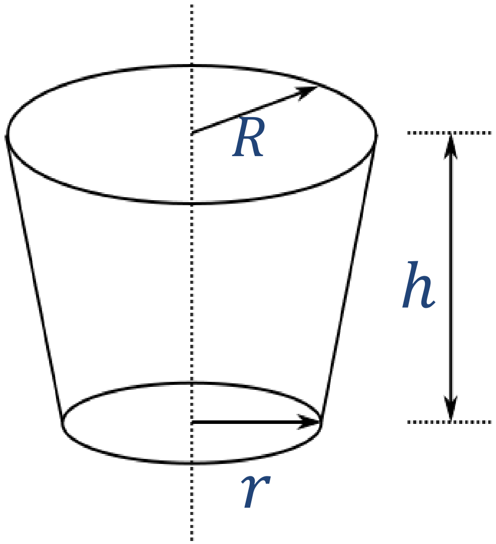
Nothing else to write: “model & run” approach

- Straightforward, natural mathematical model
- Direct resolution: no tuning

LocalSolver

Parametric optimization

Maximize the volume of a bucket with a given surface of metal




```
function model() {

  R <- float(0,1);
  r <- float(0,1);
  h <- float(0,1);

  V <- PI * h / 3.0 * (R*R + R*r + r*r);
  S <- PI * r * r + PI*(R+r) * sqrt(pow(R-r,2) + h*h);

  constraint S <= PI;
  maximize V;
}
```

$$V = \frac{\pi h}{3}(R^2 + Rr + r^2)$$

$$S = \pi r^2 + \pi(R+r)\sqrt{(R-r)^2 + h^2}$$

LocalSolver

Mathematical operators

Decisional	Arithmetical			Logical	Relational	Set-related
bool	sum	sub	prod	not	eq	count
float	min	max	abs	and	neq	indexof
int	div	mod	sqrt	or	geq	partition
list	log	exp	pow	xor	leq	disjoint
	cos	sin	tan	iif	gt	
	floor	ceil	round	array+at	lt	
	dist	scalar		piecewise		

+ operator **call** : to call an external native function which can be used to implement your own (black-box) operator

→ practical to couple simulation & optimization

LocalSolver

Traveling salesman

Find the shortest tour that visits N cities exactly once

```
function model() {
  x <- list(N) ; // order n cities {0, ..., n-1} to visit
  constraint count(x) == N; // exactly n cities to visit
  minimize sum[i in 1..N-1]( Dist[ x[i-1] ][ x[i] ] ) + Dist[ x[N-1] ][ x[0] ] ; // minimize sum of traveled distances
}
```

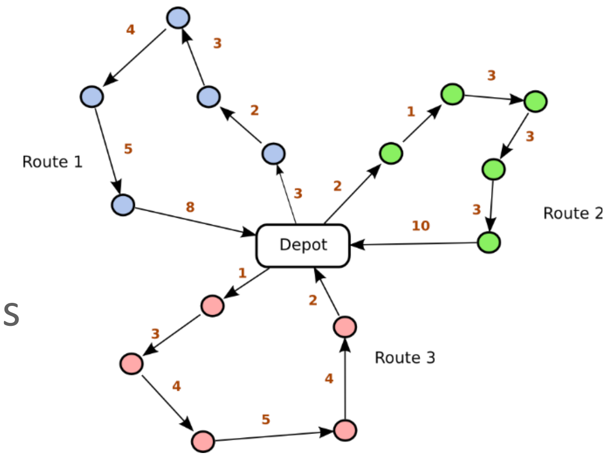
Could you imagine a simpler model

- Natural declarative model → straightforward to understand
- Compact: linear in the size of input → highly scalable (1 million nodes)

Vehicle routing, and more

On Wednesday 11:00 AM
WE-B-8 Industrial Session
Room C002

Solving routing and scheduling problem with real applications



Solving routing and scheduling problems using LocalSolver
Clément Pajean, Innovation 24 & LocalSolver

Planning LNG supply chain

Uberising private jet business

Smart APIs

C++ ISO

Java 5.0

.NET C# 2.0

Python 2.7, 3.2, 3.4

```
########## optimal_bucket.py ##########

import localsolver
import sys

with localsolver.LocalSolver() as ls:

    PI = 3.14159265359

    #
    # Declares the optimization model
    #
    m = ls.model

    R = m.float(0,1)
    r = m.float(0,1)
    h = m.float(0,1)

    # Surface constraint
    # surface = PI * r^2 + PI*(R+r) * sqrt ((R-r)^2 + h^2)
    surface = PI*r*r + PI * m.sqrt((R-r)**2 + h**2) * (R+r)
    m.constraint(surface <= PI)

    # Maximize volume
    # volume = PI * h/3 * (R^2 + R*r + r^2)
    volume = PI * h/3 * (R**2+ R*r + r**2)
    m.maximize(volume)

    m.close()

    #
    # Param
    #
    ls.param.nb_threads = 2
    if len(sys.argv) >= 3: ls.create_phase().time_limit = int(sys.argv[2])
    else: ls.create_phase().time_limit = 6

    ls.solve()
```

Clients

- Construction
- Medias & Leisure
- Telco & Retail
- Large Industry
- Energy
- Banking & Finance
- Transportation
- Logistics
- Food & Agribusiness
- Aerospace & Defense
- IT Services

Car sequencing

Smoothing car production loads along the assembly line

2005 ROADEF Challenge http://challenge.roadef.org/2005/en

RENAULT

PSA PEUGEOT CITROËN

Large-scale instances

- 1,300 vehicles to sequence
- 400,000 binary decisions

Instance with 540 vehicles

- Small instance: 80,000 variables including 44,000 binary decisions
- State of the art: 3,109 (winner of the Challenge)
- Lower bound: 3,103

Minimization
Lower is better

Benchmarks

- Gurobi 5.5: 3.027e+06 in 10 min | 194,161 in 1 hour
- LocalSolver 3.1: 3,476 in 10 sec | 3,114 in 10 min

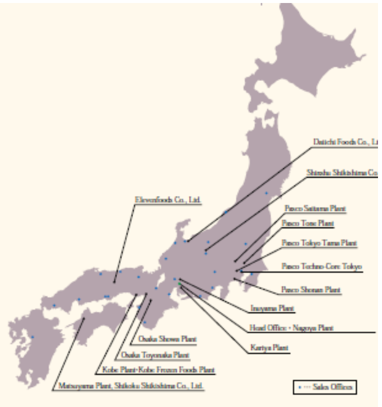
Supply chain optimization

Global supply chain optimization

- Both production and logistics optimization
- 15 factories, each with several production lines
- More than 100 distribution centers to deliver

A challenging context for LocalSolver

- 32,000,000 variables including 8 million binaries
- 1,000,000 constraints, 16 lexicographic-ordered objectives
- Vain attempts to solve the problem with MIP solvers
- LocalSolver finds high-quality solutions in 3 minutes

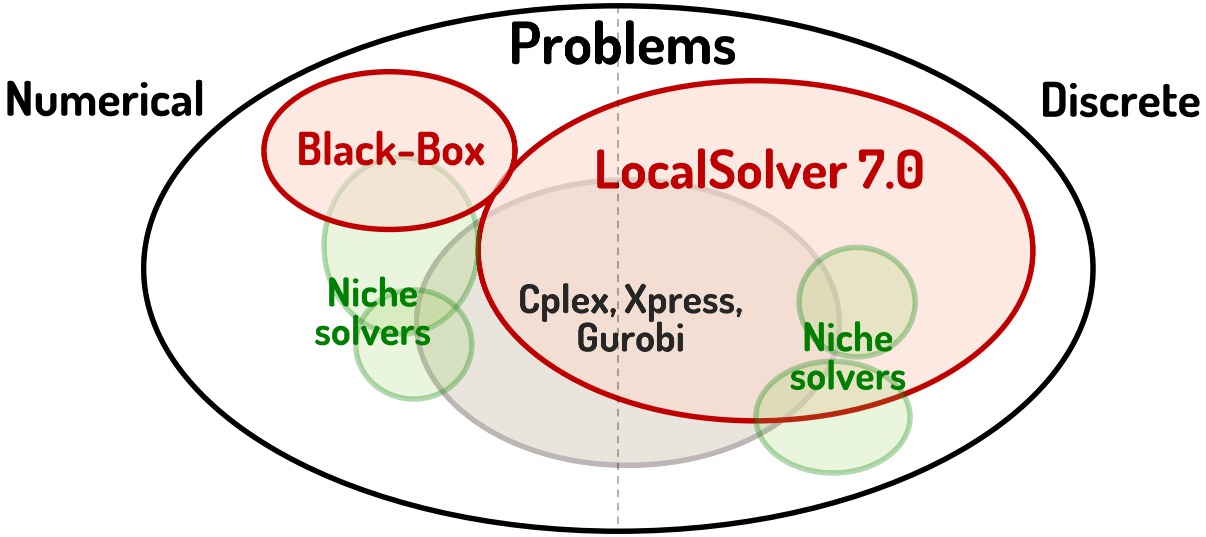


Roadmap

Major features to come

- Integration of the power of LP/MIP techniques into LocalSolver
- Development of set-based modeling features and solving performance

→ First ingredients in 6.5 version planned for July 2016



A new kind of mathematical optimization solver

Thierry Benoist, Bertrand Estellon, Julien Darlay,
Frédéric Gardi, Romain Megel, Clément Pajean

Innovation 24 & LocalSolver, Paris
www.localsolver.com

MIM 2016, Troyes