LocalSolver

www.localsolver.com

Who we are

BOUYGUES

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

Innovation24

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

LocalSolver

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

Innovation 24

Business Analytics & Optimization subsidiary of Bouygues

PhD-engineers in computer science and applied maths

20 years of experience in operations research

- Optimization
- Planning
- Forecasting
- Revenue Management
- Data Science
- Simulation
- Business Rules

\+ an innovative mathematical optimization solver

LocalSolver

www.localsolver.com

Consulting
Software solutions
LocalSolver

Management

Thierry Benoist
Partner, VP Services

- Ecole Polytechnique (X95), PhD & HDR in computer science
- 15 years of practice and research of OR
- Author of a software to optimize €1bn investment for Bouygues Telecom
- ASTI Prize for applied research (2005), 3rd Prize Robert Faure ROADEF (2006)
- Finalist of the EURO Excellence in Practice Award (2012)

Frédéric Gardi
Partner, VP LocalSolver

- PhD & HDR in computer science (UPMC, Paris 6)
- 15 years of practice and research of OR
- Author of an optimization software used by 10,000 users at Société Générale
- Winner of ROADEF Challenge (2005, 2007), 1st Prize Industrial Applications ROADEF (2011)
- 1st Prize Robert Faure ROADEF (2012), President of ROADEF (2014-2015)

ROADEF = French Operations Research and Decision Support Society (http://roadef.org)

LocalSolver

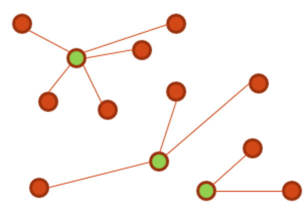
All-terrain optimization solver

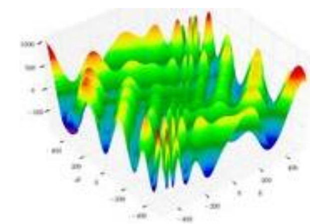
For combinatorial, numerical,
or mixed-variable optimization

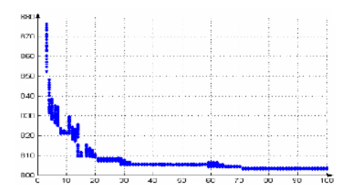
Suited for tackling
large-scale problems

Quality solutions in minutes
without tuning

The « Swiss Army Knife » of
mathematical optimization







free trial with support – free for academics – rental licenses from 590 €/month – perpetual licenses from 9,900 €

www.localsolver.com

LocalSolver

Features

Features

Better solutions, faster

- Provides high-quality solutions quickly (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

- LSP: innovative algebraic modeling language for fast prototyping
- Object-oriented C++, Java, .NET, Python APIs for tight integration

Features

Easy to deploy

- Easy and light installation, licensing, deployment
- Fully portable: Windows, Linux, Mac OS (x86, x64)
- Transparent multithreading on many-core architectures

Flexible licensing & no-surprises pricing

- Free 30-day trial licenses
- Large range of licenses to meet your needs
- OEM/ISV: open to find a licensing agreement in line with your business
- Unique service on the market: we help you to model & solve your problem

LocalSolver

Quick tour

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

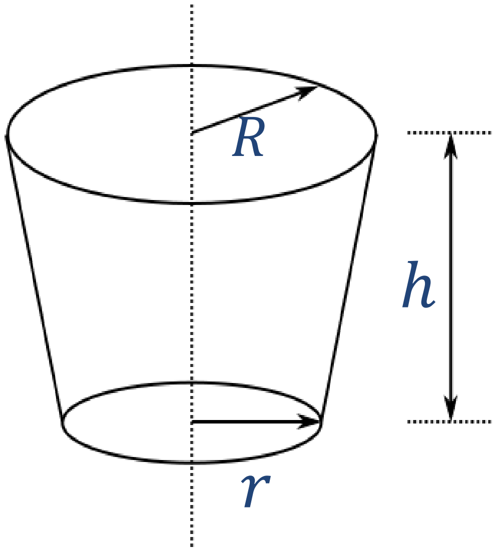
```
function model() {
  x[1..N] <- bool() ; // decisions: point i belongs to P if x[i] = 1
  constraint sum[i in 1..N]( x[i] ) == P ; // constraint: P points selected among N
  minDist[i in 1..N] <- min[j in 1..N]( x[j] ? Dist[i][j] : InfiniteDist ) ; // expressions: distance to the nearest point in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

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 <= 1;
  maximize V;
}
```

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

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

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

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
}
```

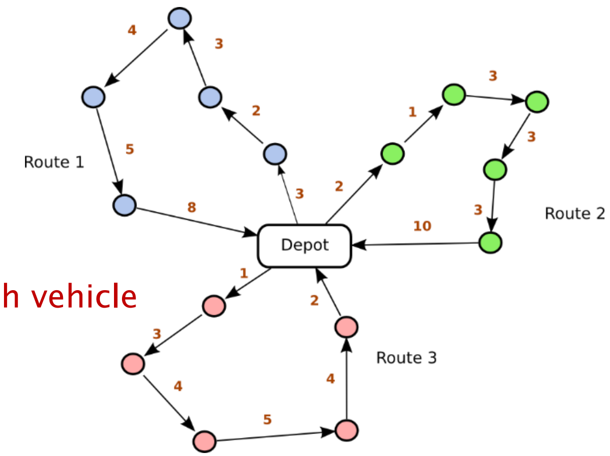
Could you imagine a simpler model

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

Vehicle routing

Find the shortest set of routes for a fleet of K vehicles in order to deliver to a given set of N customers

```
function model() {
  x[1..K] <- list(N) ; // for each vehicle, order the clients to visit
  constraint partition( x[1..K] ); // each client is visited once
  distances[k in 1..K] <- sum[i in 1..N-1]( dist( x[k][i-1], x[k][i]) )
                                    + dist( x[k][N-1], x[k][0] ); // traveled distance for each vehicle
  minimize sum[k in 1..K]( distances[k] ); // minimize total traveled distance
}
```



Black-box optimization

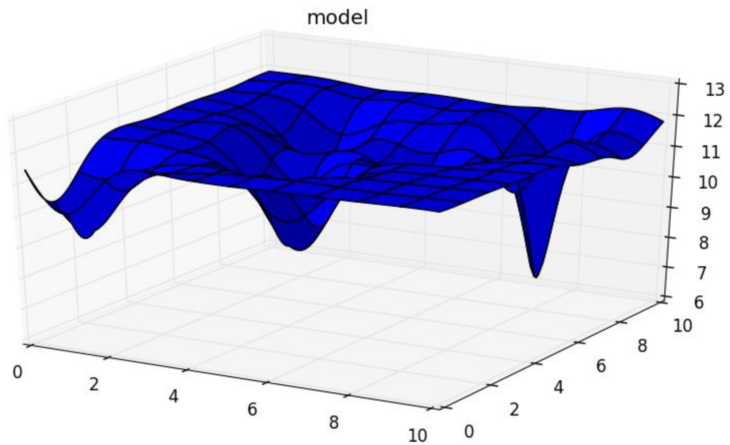
Context

- Function f without analytical form (external code or library)
- Maybe be costly to evaluate (minutes or even hours)
- f may be non-smooth, noisy, or even non-deterministic
- f defined over continuous, integer, boolean variables
- With box constraints (bounds on variables)

$$\min f(x)$$
$$x \in [x^L, x^U]$$

Engineering applications

- Product or system design -> parametric optimization
- Simulation optimization



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()
```

LocalSolver

LocalSolver

Case studies

Clients

- Construction
- Medias & Advertising
- 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

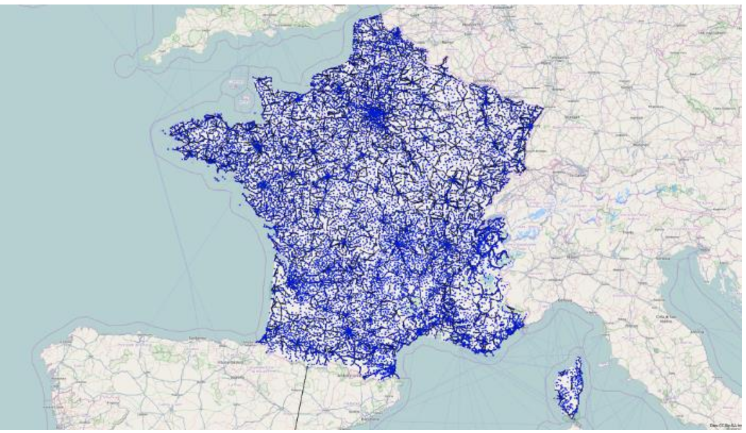
Benchmarks

- 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

Network optimization

ADSL network expansion planning

bouygues TELECOM



Context

- Choose remote concentrator units to unbundle
- Local and global constraints. Ex: forbid paths with too much clients to limit impacts of an incident

Prize collecting Steiner forest problem

- Network with 14,000 nodes and 180,000 edges
- Resulting model: 1.4 million variables
- Required resolution time: minutes

Media planning

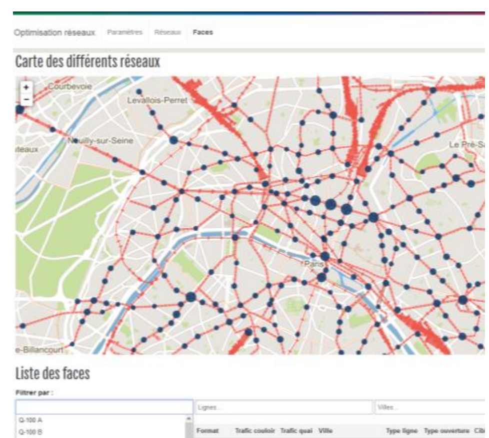
Selling advertising spaces in Paris Underground

60,000 ad points (= faces) to partition into products from 100 to 500 faces

- Covering a maximum of stations
- Balanced according to the traffic
- + a dozen of quality criteria

Huge partitioning problem solved in 1 minute

LocalSolver



Asset management

Street lighting maintenance planning

BOUYGUES ENERGIES & SERVICES

« Illuminate better with less energy »

- Plan the replacement and maintenance of street lighting fixtures over 25 years
- Taking into account costs, resources, energy consumptions, failure rates, etc.

Large-scale combinatorial optimization

- Thousands of streets
- Tens of thousands of lighting fixtures
- Nonlinear model with 1,000,000 variables

LocalSolver

LocalSolver based software used interactively in each bidding process

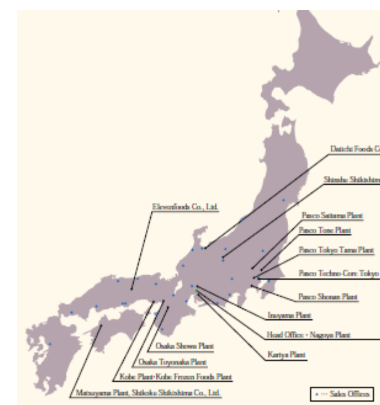
Supply chain optimization

Global supply chain optimization

- Both production and logistics optimization
- 10 factories, each with several production lines
- Large number of stores and distribution centers

A challenging context for LocalSolver

- 20,000,000 variables including 3 million binaries
- Rich model involving setup costs, delivery times, packaging, etc.
- Vain attempts to solve the problem with MIP solvers
- LocalSolver finds high-quality solutions in 5 minutes



LocalSolver

Roadmap

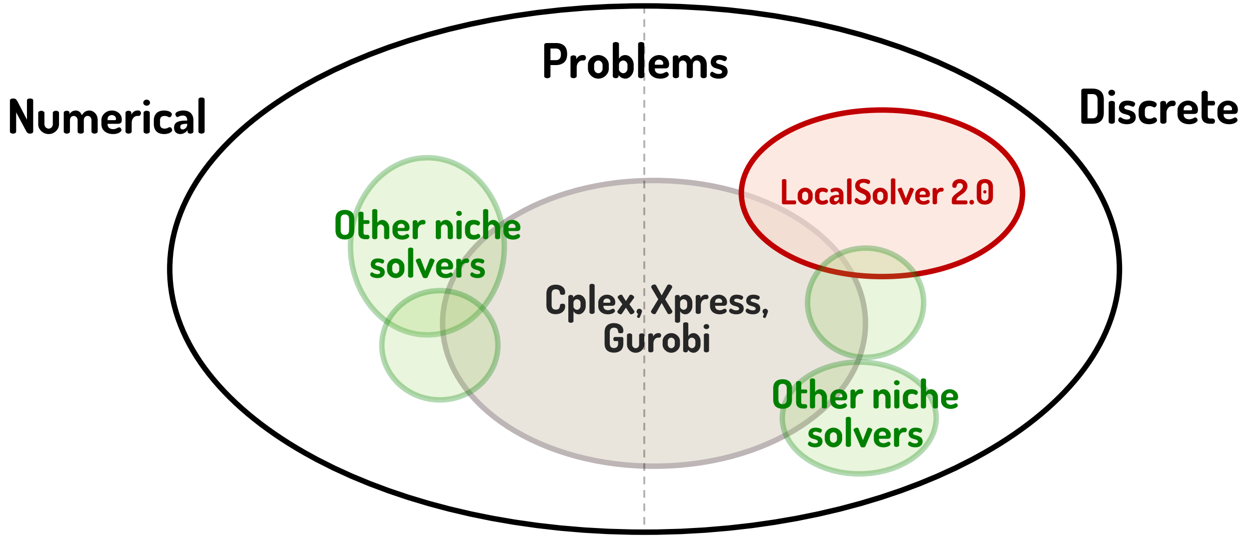
Initially a niche product

Strengths

- Out with the mainstream in optimization
- Enriched modeling formalism

Weaknesses

- Significant cultural change: barrier to entry
- Limited scope to 0/1-decision problems: niche applications



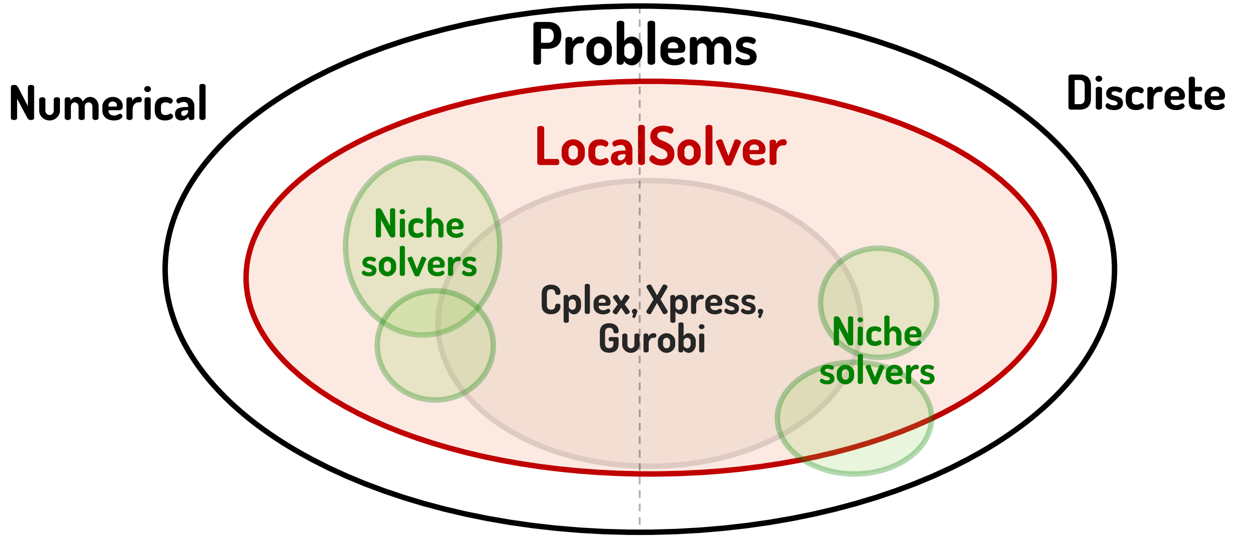
LocalSolver

But an ambitious roadmap

Two missions, one grail

- To produce a technological breakthrough, functionally and technically, addressing the entire mathematical optimization market
- To "convert" the field and the market to LocalSolver

→ To build and offer the best mathematical optimization solver



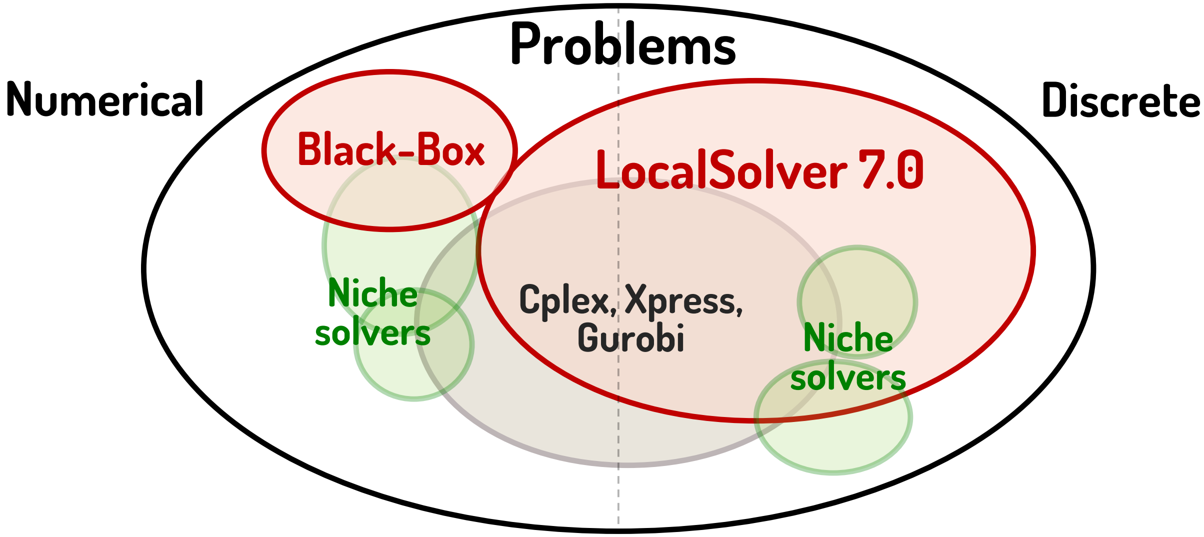
LocalSolver

LocalSolver 7.0: end of 2016

Major features

- 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 June 2016



LocalSolver

Long-term roadmap

John N. Hooker (2007)

"Good and Bad Futures for Constraint Programming (and Operations Research)"
Constraint Programming Letters 1, pp. 21-32

"Since modeling is the master and computation the servant, no computational method should presume to have its own solver.

This means there should be no CP solvers, no MIP solvers, and no SAT solvers. All of these techniques should be available in a single system to solve the model at hand.

They should seamlessly combine to exploit problem structure. Exact methods should evolve gracefully into inexact and heuristic methods as the problem scales up."

LocalSolver

www.localsolver.com