LocalSolver

# A new kind of mathematical programming solver

Thierry Benoist Julien Darlay Bertrand Estellon
Frédéric Gardi Romain Megel

**fgardi@localsolver.com**

**www.localsolver.com**

# Who we are

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

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

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

# Why LocalSolver?

## To answer to unmet needs in optimization

- Simple and generic (nonlinear) mathematical modeling formalism
- Provide high-quality solutions quickly
- Scalable to tackle million-variable problems
- Focus your work on modeling: no need of complex tuning

## Easy to install, use, license, deploy

- For fast prototyping: LocalSolver modeling & scripting language (LSP)
- For tight integration: object-oriented C++, Java, .NET callable libraries
- Fully portable: Windows, Linux, Mac OS (x86, x64)
- Free trial with dedicated & reactive support (even for modeling)
- Transparent licensing & pricing

Free for academics

LocalSolver

# LocalSolver

Quick tour

# Combinatorial optimization

P-median: select a subset P among N points minimizing the sum of distances to each point from N to the nearest point in P

Binary decisions

```
function model() {
  x[1..N] <- bool() ; // decision : point i is in P iff x[i] = 1

  constraint sum[i in 1..N](x[i]) == P ;

  minDist[i in 1..N] <- min[j in 1..N](x[j] ? Dist[i][j] : +inf);

  minimize sum[i in 1..N]( minDist[i] ) ; // minimize sum of distances from P to N

  maximize sum[i in 1..N][j in i..N]( Dist[i][j] * x[i] * x[j] ); // maximize dispersion in P
}
```

Nonlinear expressions

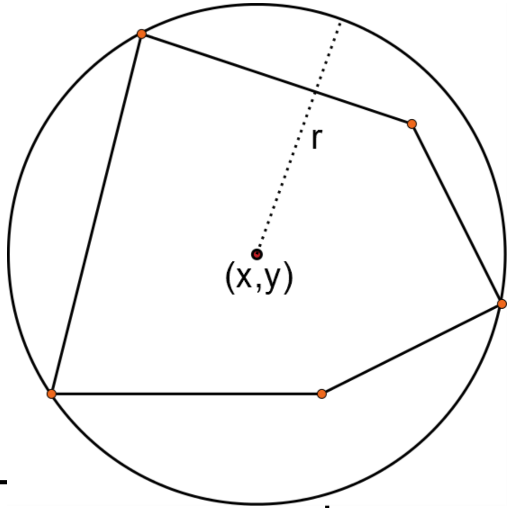
## Nothing else to write: “model & run” approach

- Straightforward mathematical model
- Direct resolution: no tuning

LocalSolver

# Numerical optimization

**Smallest circle:** find a circle with minimal radius which contains a set of points in the plane




Continuous (floating-point) decisions

Nonlinear expressions

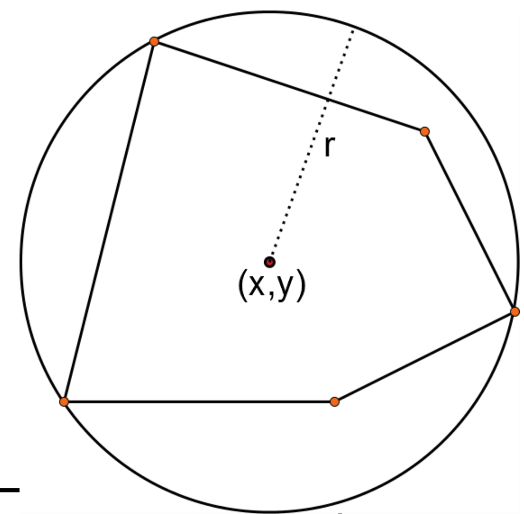
```
x <- float(minx, maxX);
y <- float(minY, maxY);

r2 <- max[i in 1..n]( pow(x-coordX[i],2) + pow(y-coordY[i],2) );

minimize sqrt(r2);
```

# Mixed-variable optimization

Smallest circle: find a circle with integer abscissa and minimal radius which contains a set of points in the plane




Integer decision

Floating-point decision

```
x <- int(minx, maxX);
y <- float(minY, maxY);

r2 <- max[i in 1..n]( pow(x-coordX[i],2) + pow(y-coordY[i],2) );

minimize sqrt(r2);
```

# Mathematical operators

| Decisional | Arithmetic | | | Logical | Relational |
|---|---|---|---|---|---|
| bool | sum | sub | prod | not | == |
| float | min | max | abs | and | != |
| int | div | mod | sqrt | or | <= |
| | log | exp | pow | xor | >= |
| | cos | sin | tan | if | < |
| | floor | ceil | round | array + at | > |

# Benchmarks
# & Case studies

# Combinatorial optimization

## Car Sequencing : schedule cars along an assembly line

| 10 sec | 100 cars | 200 cars | 300 cars | 400 cars | 500 cars |
|---|---|---|---|---|---|
| Gurobi 5.6 | 140 | 274 | X | 429 | 513 |
| LocalSolver 4.5 | 8 | 5 | 8 | 10 | 19 |
| **60 sec** | **100 cars** | **200 cars** | **300 cars** | **400 cars** | **500 cars** |
| Gurobi 5.6 | 3 | 66 | 1 | 356 | 513 |
| LocalSolver 4.5 | 6 | 4 | 3 | 5 | 6 |
| **600 sec** | **100 cars** | **200 cars** | **300 cars** | **400 cars** | **500 cars** |
| Gurobi 5.6 | 3 | 2 | *0 | 1 | 20 |
| LocalSolver 4.5 | 4 | *0 | *0 | 2 | *0 |

**Lower is better**

# Combinatorial optimization

Instances are public: problem submitted as ROADEF-EURO Challenge in 2005
http://challenge.roadef.org/2005/en

## Real-life car sequencing

- Until 1300 cars to sequence → 400,000 binary decisions

## Instance 022_EP_ENP_RAF_S22_J1

- Small instance: 80,000 variables including 44,000 binary decisions
- State of the art: 3,109 obtained by a specific local search algorithm
- Best lower bound: 3,103

**Lower is better**

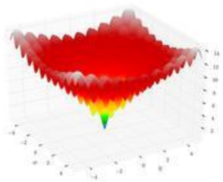
## Results

- Gurobi 5.6: 3.116647e+07 in 10 min | 25,197 in 1 hour
- LocalSolver 4.5: 3,478 in 10 sec | 3,118 in 10 min

# Non-convex numerical optimization

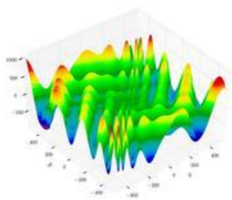
## Near-optimal solutions in a few seconds on artificial landscapes

Oldenhuis (2009). Test functions for global optimization algorithms. Matlab



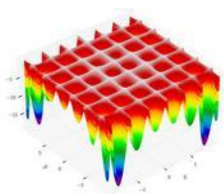
$$f(x,y) = -20\exp\left(-0.2\sqrt{0.5\left(x^2+y^2\right)}\right) - \exp\left(0.5\left(\cos\left(2\pi x\right)+\cos\left(2\pi y\right)\right)\right) + 20 + e.$$

gap (%) < $10^{-6}$



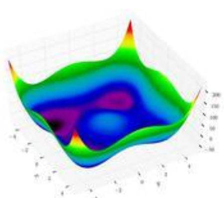
$$f(x,y) = -\left(y+47\right)\sin\left(\sqrt{\left|y+\frac{x}{2}+47\right|}\right) - x\sin\left(\sqrt{\left|x-\left(y+47\right)\right|}\right).$$

gap (%) < $10^{-6}$



$$f(x,y) = -\left|\sin\left(x\right)\cos\left(y\right)\exp\left(\left|1-\frac{\sqrt{x^2+y^2}}{\pi}\right|\right)\right|.$$

gap (%) < $10^{-6}$



$$f(\boldsymbol{x}) = \frac{\sum_{i=1}^{n} x_i^4 - 16x_i^2 + 5x_i}{2}.$$

n = 1,000,000

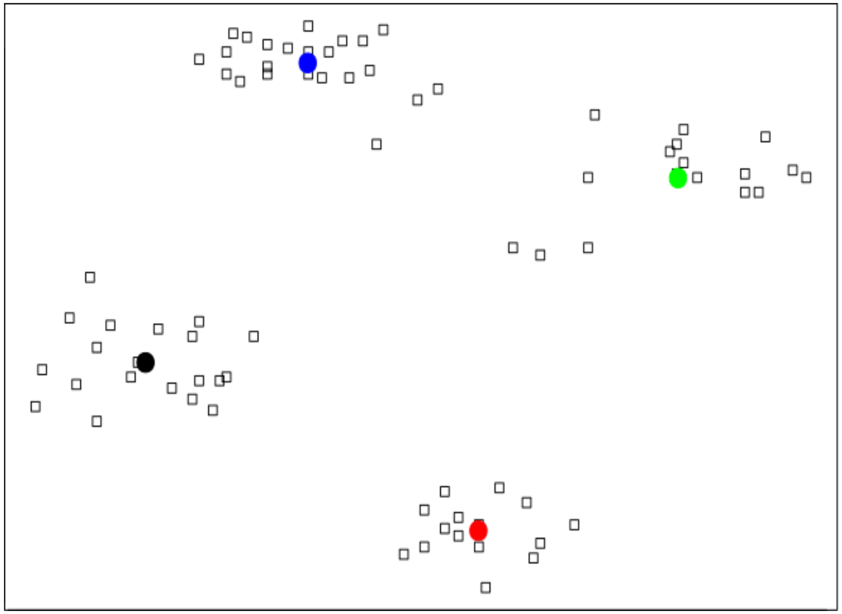
gap (%) < $10^{-6}$

# Machine learning application

## K-means

- Partition observations into k classes to minimize within-clusters sum of squares
- Non-convex quadratic continuous problem
- Until 10,000 observations, 20 dimensions, k=2..50 clusters
- Direct LocalSolver model (60 sec) versus state-of-the-art CG approach (1 day)

| Instance | k | OPT* | LS 4.5 | GAP |
|---|---|---|---|---|
| iris | 2 | 152.348 | 152.369 | 0.01% |
| | 3 | 78.8514 | 78.9412 | 0.11% |
| | 4 | 57.2285 | 57.3556 | 0.22% |
| | 5 | 46.4462 | 46.5363 | 0.19% |
| | 6 | 39.04 | 41.7964 | 1.06% |
| | 7 | 34.2982 | 34.6489 | 1.02% |
| | 8 | 29.9889 | 30.3029 | 1.05% |
| | 9 | 27.7861 | 28.0667 | 1.01% |
| | 10 | 25.834 | 26.0521 | 0.84% |



**Lower is better**

LocalSolver

# Client application panorama

- FUJITSU — Supply chain optimization
- Pasco — Production scheduling + distribution planning
- TF1 PUBLICITE — TV media planning
- AIR LIQUIDE — Field service routing & scheduling
- BOUYGUES ENERGIES & SERVICES — Street lighting maintenance planning
- Bouygues Telecom — Network deployment planning
- SIEMENS — Energy optimization for tramway lines
- eDF — Hydro valley optimization
- PUBLICIS — Advertising display optimization in Paris subway
- ARMEE DE TERRE — Packing and transportation of military equipment

# LocalSolver summary

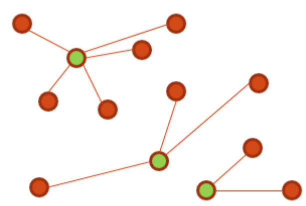
Hybrid math programming solver

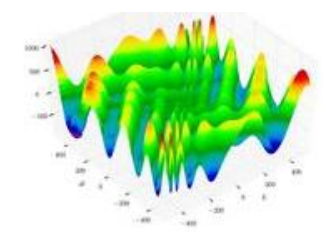
For combinatorial, numerical,
or mixed-variable optimization

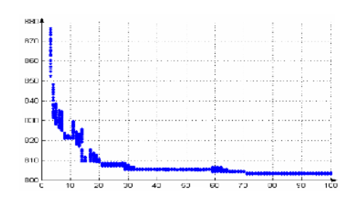
Particularly suited for large-scale
non-convex optimization

High-quality solutions in seconds
without tuning

LocalSolver
=
LS + CP/SAT + LP/MIP + NLP







**free trial with support – free for academics – renting licenses from 590 €/month – perpetual licenses from 9900 €**

**www.localsolver.com**