LocalSolver 4.0 : nouveautés et benchmarks

Thierry Benoist   Julien Darlay   Bertrand Estellon
Frédéric Gardi   Romain Megel

www.localsolver.com

LocalSolver 3.1

Solveur pour l'optimisation combinatoire

- Formalisme de modélisation mathématique simple
- Permet d'attaquer des problèmes combinatoires de grande taille
- Fournit de bonnes solutions en des temps courts

Solveur basé sur la recherche locale

- Mouvements basés sur l'hypergraphe décisions/contraintes
- Evaluation incrémentale : millions de mouvements par minute
- Recuit simulé adaptatif, randomisé, parallélisé, avec restart

Licences académiques gratuites

Licences commerciales à partir de 990 €

LocalSolver

P-médian

Sélectionner un sous-ensemble P parmi N points minimisant la somme des distances de chaque point dans N au point le plus proche dans P.

```
function model() {
  x[1..N] <- bool() ; // décisions : le point i appartient à P si 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] : InfiniteDist);

  minimize sum[i in 1..N]( minDist[i] ) ; // objectif : minimiser la somme des distances
}
```

Approche “model & run”

- Modèle mathématique naturel
- Résolution directe : aucun paramétrage

LocalSolver

LocalSolver 4.0

LocalSolver 4.0

Solveur de programmation mathématique

- Pour l'optimisation combinatoire
- Pour l'optimisation numérique
- Pour l'optimisation en variables mixtes

- Fournit des solutions (bornes supérieures)
- Fournit des bornes inférieures
- Preuve d'infaisabilité, écart ou preuve d'optimalité

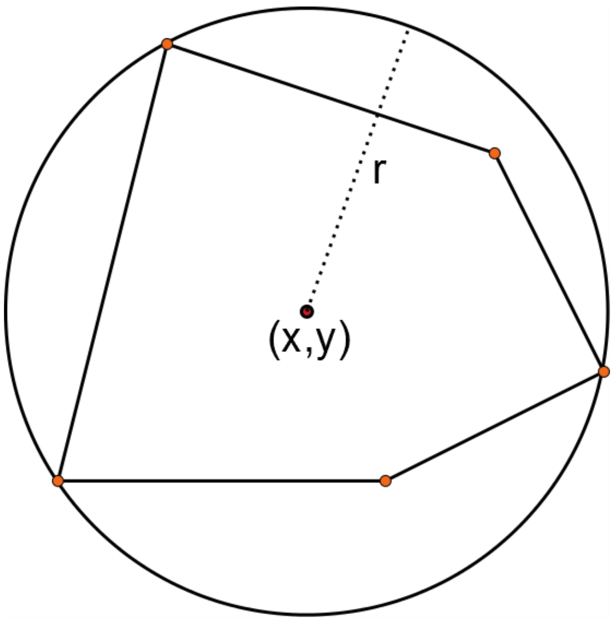
Idéal pour l'optimisation non-convexe à grande échelle

- Millions de variables combinatoires et/ou continues
- Contraintes et/ou objectifs non-convexes
- Temps de résolution courts

Optimisation numérique

Smallest Circle

- Trouver le cercle de rayon minimal contenant un ensemble de points
- Deux variables de décisions continues ***x*** et ***y***
- Le rayon ***r*** : une expression déduite des décisions
- Modèle quadratique simple et naturel



Décision continue

Expression quadratique

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

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

minimize sqrt(r);
```

LocalSolver

Nouveautés

Solveur hybride

- Recherche locale mixte → technologie unique sur le marché
- Propagation de contraintes et inférence
- Programmation linéaire (en nombres entiers)

Preprocessing renforcé

- Détection et reformulation de modèles « orientés MIP »
- Bornes inférieures par inférence
- Détection d'infaisabilité améliorée

Voisinages plus larges

- Composition de voisinages locaux (combinatoires, continus ou mixtes)
- Exploitation de la relaxation linéaire dans les espaces combinatoires
- Exploitation de sous-problèmes linéaires dans les espaces non-convexes

Recherche locale

Idée maîtresse en optimisation combinatoire

- Modification séquentielle d’un petit nombre de décisions
- Maintien de la faisabilité de la solution courante : réparation des violations
- Evaluation incrémentale, généralement en temps $O(1)$

→ Faible probabilité d’amélioration, mais faible coût en temps et espace

Et en optimisation continue ?

- Connue sous un autre nom : *direct = derivative-free = zeroth-order search*
- N’utilise ni gradient (1^{er} ordre) ni Hessien (2^{nd} ordre)
- Ex : l’autre algorithme du simplexe de Nelder-Mead
- Majoritairement utilisée en optimisation non convexe <u>sans contrainte</u>

Recherche locale mixte

Unification combinatoire & continue

- Recherche locale combinatoire, continue, ou mixte
- Même idée que pour le combinatoire
- Une difficulté : le maintien de la faisabilité

À contre-courant du mainstream en optimisation

- Volonté de fournir des garanties de convergence vers un optimum
- Recherche d'un meilleur améliorant (*best improvement*)
- Techniques d'amélioration sophistiquées et très coûteuses en temps

→ Itération en temps quadratique voire cubique en le nombre de variables

Benchmarks

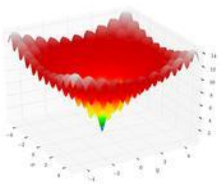
Car Sequencing : ordonnancer une ligne d'assemblage de véhicules

10 sec	100	200	300	400	500
Gurobi 5.5	140	274	X	429	513
LocalSolver 3.1	6	8	9	11	24
LocalSolver 4.0	8	5	8	10	19
60 sec	**100**	**200**	**300**	**400**	**500**
Gurobi 5.5	3	66	1	356	513
LocalSolver 3.1	6	3	3	7	10
LocalSolver 4.0	6	4	3	5	6
600 sec	**100**	**200**	**300**	**400**	**500**
Gurobi 5.5	3	2	*0	1	20
LocalSolver 3.1	6	2	1	2	4
LocalSolver 4.0	4	*0	*0	2	*0

Optimisation non-convexe sans contrainte

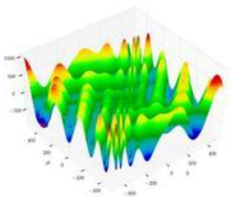
Solutions quasi optimales en quelques secondes sur une cinquantaine de paysages artificiels de la littérature

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



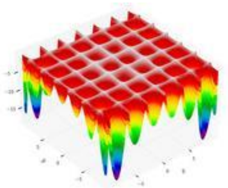
$$f(x,y) = -20\exp\left(-0.2\sqrt{0.5\left(x^2+y^2\right)}\right) - \exp\left(0.5\left(\cos\left(2\pi x\right)+\cos\left(2\pi y\right)\right)\right) + 20 + e.$$

gap (%) < 10^{-6}



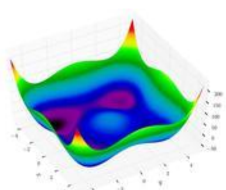
$$f(x,y) = -\left(y+47\right)\sin\left(\sqrt{\left|y+\frac{x}{2}+47\right|}\right) - x\sin\left(\sqrt{\left|x-\left(y+47\right)\right|}\right).$$

gap (%) < 10^{-4}



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

gap (%) < 10^{-4}



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

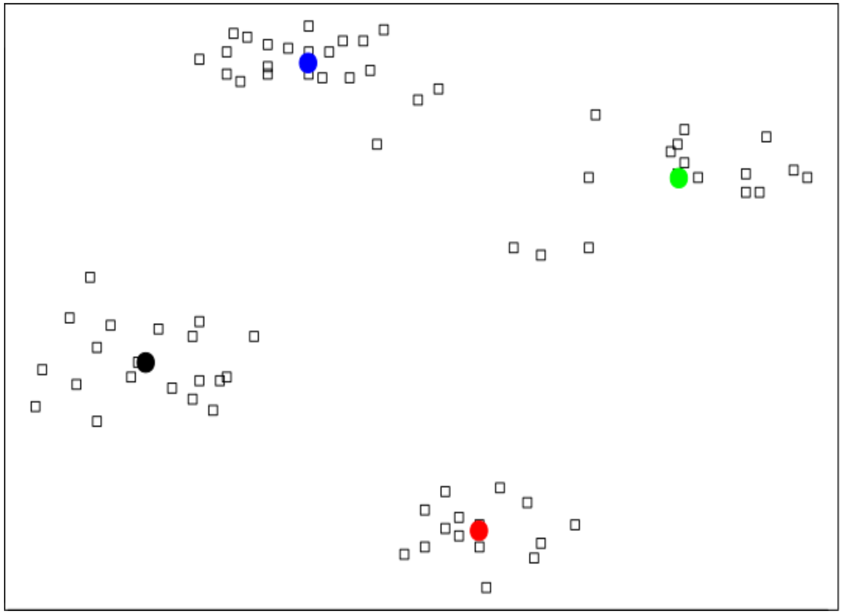
n = 10 → 10000

gap (%) < 10^{-6} → 10^{-1}

Optimisation non-convexe sous contraintes

K-means

- Problème de machine learning
- NP-Difficile, Quadratique
- Solutions en 300 sec



Instance	k	OPT*	LS 4.0	GAP
ruspini	2	89337	89337,9	0,00%
	3	51063,4	51063,5	0,00%
	4	12881	12881,1	0,00%
	5	10126,7	10126,8	0,00%
	6	8575,41	8670,86	1,11%
	7	7126,2	7159,13	0,46%
	8	6149,64	6158,26	0,14%
	9	5181,64	5277,11	1,84%
	10	4446,28	4856,98	9,24%
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	7,06%
	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%
glass	20	114,646	120,048	4,71%
	30	63,2478	74,1251	17,20%
	40	39,4983	58,3912	47,83%
	50	26,7675	52,4679	96,01%

*[Aloise et al. 2012]

Optimisation en variables mixtes

Optimisation des vallées hydrauliques (KEPCO, Japon)

- 4 barrages hydroélectriques avec 2 à 6 pompes par barrage
- 30 unités de production thermique
- Apports d'eau et prix de l'électricité variables dans le temps
- Horizon long : 8000 pas de temps

→ Système dynamique fortement non-linéaire, en variables mixtes, avec contraintes très couplantes, à grande échelle (1 million de décisions)

Cplex, Xpress, Gurobi : pas de solution après 3 h

LocalSolver : solution de qualité en une minute

Vers un solveur « complet »

Un solveur pour attaquer tout type de problèmes

- Optimisation combinatoire, continue, ou mixte
- Problèmes de petite ou grande taille
- Capable de prouver l’optimum ou l’infaisabilité
- Capable de passer à l’échelle de façon heuristique

Un solveur offrant le meilleur de toutes les techniques

- Recherche locale & directe
- Propagation de contraintes et inférence
- Programmation linéaire/en nombres entiers
- Programmation non-linéaire (convexe et non-convexe)
- Programmation dynamique
- Algorithmes spécifiques (chemins, flots, couplages, etc.)

LocalSolver à ROADEF 2014

LocalSolver: schémas de modélisation
Thierry Benoist, jeudi 15h, Bât C Sigalas

Optimisation de réseaux publicitaires avec LocalSolver
Romain Megel, vendredi 11h, Bât B TD 35

Modèle LocalSolver d'ordonnancement
d'une machine unique sous contraintes de bin-packing
Clément Pajean, vendredi 14h, Bât B TD 35

www.localsolver.com **www.innovation24.fr**

LocalSolver