Numerical methods for optimization problems: an application to energetic districts

Hansjörg Wacker Memorial Prize
Elisa Riccietti
Università degli Studi di Firenze
Dipartimento di Matematica e Informatica 'Ulisse Dini'
Supervisor: Prof. Stefania Bellavia, Università di Firenze
Co-Supervisor: Dott. Stefano Sello, Enel Ingegneria e Ricerca

The 19th European Conference on Mathematics for Industry (ECMI),
Santiago de Compostela 13-17 June 2016.

Aim of the thesis

- This thesis is in collaboration with the research center 'Enel Ingegneria e Ricerca' in Pisa, Italy.
- Enel is the main energy provider in Italy.
- Six months stage for my thesis.

Energetic districts

- An energetic district is an industrial complex composed of three different kinds of machines:
  - loads, that consume energy,
  - generators, that produce energy,
  - accumulators, that store energy.
- The energetic district is interconnected to the electrical grid and can buy energy from it or also sell energy, if the amount produced within the district exceed its needs.

Optimization variables

- The problems variables are the machines set-points: quantities related to physical parameters of the machines, like generators power or operating time for loads.
- A plan has to be built the day ahead with the set-points of all the machines of the district, for every 15 minutes of the next day.
- The optimization process has to be performed in a short time, as it should be possible to quickly change the plan.

The software package

- For the optimization process several informations are taken into account:
  - real time information and forecasts on customer energy needs and production,
  - weather forecast affecting the renewable energy sources,
  - forecasts on energy market prices in the next hours.

The software package

- For the optimization process several informations are taken into account:
  - real time information and forecasts on customer energy needs and production,
  - weather forecast affecting the renewable energy sources,
  - forecasts on energy market prices in the next hours.
- Enel developed a software package that builds a model of the energetic district: a cost function and the constraints functions, modeling the physical machines behavior, are built.
- Aim: optimize the energy resources management to reduce the customer energy bill: find the best set-points combination to minimize the objective cost function.

The optimization problem

The arising problem is a non-linear constrained optimization problem:

$$\min_{x \in \mathbb{R}^n} f(x)$$
$$\text{s.t. } x_{\min} \leq x \leq x_{\max},$$
$$g(x) \leq 0,$$

where

$$f : \mathbb{R}^n \to \mathbb{R},$$
$$g : \mathbb{R}^n \to \mathbb{R}^m,$$

and n is the number of set-points to be set, m is the number of non-linear constraints.

The thesis project is divided into three parts:

1. Analysis of the package implemented by Enel: this relies on a Sequential Linear Programming (SLP) solver, that was not theoretically supported and did not rely on a reliable stopping criterion.
   - Design of a variant of the implemented SLP solver, based on the theory of penalty functions, that is proved to converge to first order critical points.
   - Definition of an appropriate stopping criterion for the proposed solver.

The thesis project is divided into three parts:

1. Analysis of the package implemented by Enel: this relies on a Sequential Linear Programming (SLP) solver, that was not theoretically supported and did not rely on a reliable stopping criterion.
   - Design of a variant of the implemented SLP solver, based on the theory of penalty functions, that is proved to converge to first order critical points.
   - Definition of an appropriate stopping criterion for the proposed solver.
2. Study of Particle Swarm Optimization (PSO) methods, to attempt to find the global minimizer of the problem.
   - Identification of PSO most suitable variant for the specific problem.

The thesis project is divided into three parts:

1. Analysis of the package implemented by Enel: this relies on a Sequential Linear Programming (SLP) solver, that was not theoretically supported and did not rely on a reliable stopping criterion.
    - Design of a variant of the implemented SLP solver, based on the theory of penalty functions, that is proved to converge to first order critical points.
    - Definition of an appropriate stopping criterion for the proposed solver.
2. Study of Particle Swarm Optimization (PSO) methods, to attempt to find the global minimizer of the problem.
    - Identification of PSO most suitable variant for the specific problem.
3. Numerical tests with the three developed solvers.

Sequential Linear Programming (SLP)

- Iterative method: at each iteration, the objective function and the constraints functions are linearized around the current iterate x_k:

$$m_k(p) = f(x_k) + \nabla f(x_k)p,$$
$$g_{i,k}(p) = g_i(x_k) + \nabla g_i(x_k)p, \quad i = 1, \dots, m,$$

where $p = x - x_k$.

Sequential Linear Programming (SLP)

- Iterative method: at each iteration, the objective function and the constraints functions are linearized around the current iterate x_k:

$$m_k(p) = f(x_k) + \nabla f(x_k)p,$$
$$g_{i,k}(p) = g_i(x_k) + \nabla g_i(x_k)p, \quad i = 1, \dots, m,$$

  where $p = x - x_k$.
- At each iteration a linear programming subproblem has to be solved:

$$\min_p m_k(p)$$
$$\text{s.t. } g_i(x_k) + \nabla g_i(x_k)p \leq 0, \quad i = 1, \dots, m$$
$$(x_{\min} - x_k) \leq p \leq (x_{\max} - x_k).$$

  to find the step p_k that is used to compute the new iterate: $x_{k+1} = x_k + p_k$.

- The SLP method is coupled with a trust region approach, to obtain global convergence: at each iteration the step is searched in a ball around the current iteration, the trust region.
- Given the trust-region radius Δ_k, the step p_k is computed solving the minimization problem:

$$\begin{aligned} & \min_p m_k(p) \\ \text{s.t. } & g_i(x_k) + \nabla g_i(x_k) p \leq 0, \quad i = 1, \ldots, m \\ & \max((x_{\min} - x_k), -\Delta_k) \leq p \leq \min((x_{\max} - x_k), \Delta_k). \end{aligned}$$

- The SLP method is coupled with a trust region approach, to obtain global convergence: at each iteration the step is searched in a ball around the current iteration, the trust region.
- Given the trust-region radius Δ_k, the step p_k is computed solving the minimization problem:

$$\begin{aligned} &\min_p m_k(p) \\ \text{s.t. } & g_i(x_k) + \nabla g_i(x_k) p \le 0, \quad i = 1, \dots, m \\ & \max((x_{\min} - x_k), -\Delta_k) \le p \le \min((x_{\max} - x_k), \Delta_k). \end{aligned}$$

- Step acceptance:

$$\rho_k = \frac{f(x_k) - f(x_{k+1})}{m_k(0) - m_k(p_k)}.$$

Given $\eta \in (0, 1)$:

  - If $\rho_k < \eta$ then set $\Delta_{k+1} < \Delta_k$ and $x_{k+1} = x_k$.
  - If $\rho_k \ge \eta$ then set $\Delta_{k+1} \ge \Delta_k$ and $x_{k+1} = x_k + p_k$.

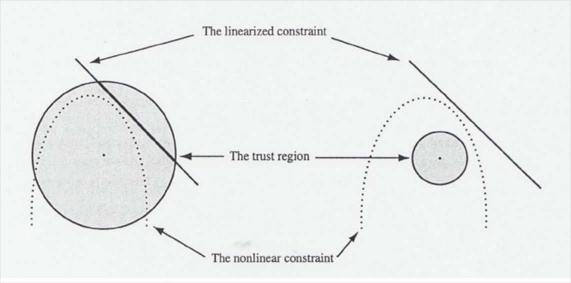
Drawbacks of this approach

- It is not sure that the linearized problem has a solution, the linearized constraints could be inconsistent.
- The new iterate could not satisfy the nonlinear constraints.

Drawbacks of this approach

- It is not sure that the linearized problem has a solution, the linearized constraints could be inconsistent.
- The new iterate could not satisfy the nonlinear constraints.

$\longrightarrow$ In this cases the trust region radius is reduced and the linearized problem is solved again. This is not ensured to work.



Our alternative approach: penalty functions

- A penalty function is used: given $\nu_k > 0$ penalty parameter, the penalty function is defined by

$$\Phi(x, \nu_k) = f(x) + \nu_k \sum_{i=1}^{m} \max(0, g_i(x)).$$

$\rightarrow$ If ν_k is big enough a minimizer of Φ is a solution of the original problem.

Our alternative approach: penalty functions

- A penalty function is used: given $\nu_k > 0$ penalty parameter, the penalty function is defined by

$$\Phi(x, \nu_k) = f(x) + \nu_k \sum_{i=1}^{m} \max(0, g_i(x)).$$

$\rightarrow$ If ν_k is big enough a minimizer of Φ is a solution of the original problem.

- The following subproblem is solved, that certainly has a solution:

$$\min_p l_k(p) = f(x_k) + \nabla f(x_k)^T p + \nu \sum_{i \in \mathcal{I}} \max(0, g_i(x_k) + \nabla g_i(x_k)^T p)$$

$$\text{s.t.} \max((x_{min} - x_k), -\Delta_k) \leq p \leq \min((x_{max} - x_k), \Delta_k).$$

Convergence results

We proved

- Global convergence of the generated sequence to a KKT point of $\Phi(x, \nu_k)$.
- Convergence of the Lagrangian multipliers estimates of the linearized problems to the Lagrangian multipliers of the original problem.

We will denote this approach SLP_2, and the previous one SLP_1.

Convergence to global minima

- The second stage of the project was devoted to investigate the convergence to global minimizers.
- To this aim it was decided to develop a Particle Swarm Optimizer (PSO).
- PSO methods are stochastic methods inspired by the behavior of bird swarms.
- They are derivative free methods, just functions values are necessary for the optimization process.
- A population of particles is evolved in the search space. Each particle is a possible solution of the problem.

PSO is an iterative method. Each particle i has three vectors associated to it in each iteration k:

- a position vector x_i^k,
- a velocity vector v_i^k,
- a vector with the best position reached so far by the particle during the evolutionary process, $P_{i,best}^k$.

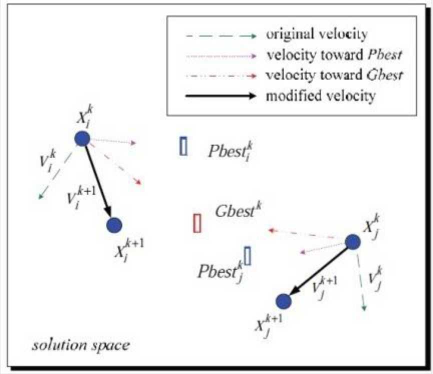
At each iteration the best particle in the swarm is selected:

$$P_{g,best}^k = \operatorname*{argmin}_i f(x_i^k).$$

The three vectors defining each particle are updated:

- $v_i^{k+1} = wv_i^k + c_1 r_1 (P_{i,best}^k - x_i^k) + c_2 r_2 (P_{g,best}^k - x_i^k)$,
- $x_i^{k+1} = x_i^k + v_i^{k+1}$,

w, c_1, c_2 free coefficients to be set, r_1, r_2 random variables.



Constraints

- Bound constraints: let $x_i^k(j)$ be the j-th component of x_i^k,

$$x_i^k(j) = \begin{cases} x_{\min}(j) & \text{if } x_i^k(j) < x_{\min}(j); \\ x_{\max}(j) & \text{if } x_i^k(j) > x_{\max}(j); \end{cases} \qquad v_i^k(j) = -r_3 v_i^k(j),$$

with r_3 random variable.

Constraints

- Bound constraints: let $x_i^k(j)$ be the j-th component of x_i^k,

$$x_i^k(j) = \begin{cases} x_{\min}(j) & \text{if } x_i^k(j) < x_{\min}(j); \\ x_{\max}(j) & \text{if } x_i^k(j) > x_{\max}(j); \end{cases} \quad v_i^k(j) = -r_3 v_i^k(j),$$

  with r_3 random variable.

- Non linear constraints: quadratic penalty function

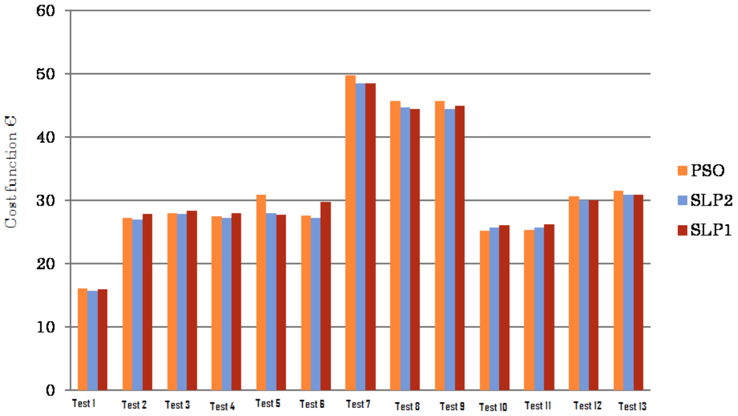
$$\Phi(x, \tau_k) = f(x) + \frac{1}{2\tau_k} \sum_{i=1}^{m} \max(0, g_i(x))^2.$$

  Penalty parameter τ was varied among the iterations: $\tau_0 = 0.1$, $\tau_{k+1} = (1 - 0.01)\tau_k$.

Numerical results of 13 different artificial test cases

The developed solvers were inserted in the Enel package and tested on different realistic examples of energetic districts.

$\longrightarrow$ On average SLP_2 has the best performance.



Numerical results: artificial districts

Results refers to 10 different runs.

Solver	f	σ_f	max f	min f	k	σ_k	max k	min k	time/iter(s)	time(m)
PSO	25.3	0.1	25.5	25.1	1513	214	1714	1238	0.5	12.6
SLP_1	26.2	1.6	30.4	25.3	68	7	82	57	0.9	1.0
SLP_2	25.6	0.4	26.4	25.2	35	2	39	33	0.9	0.5

- PSO does not require derivatives, each iteration is less expensive then a SLP iteration.
- PSO performes many more iteration than SLP.
- Execution time for PSO is larger.

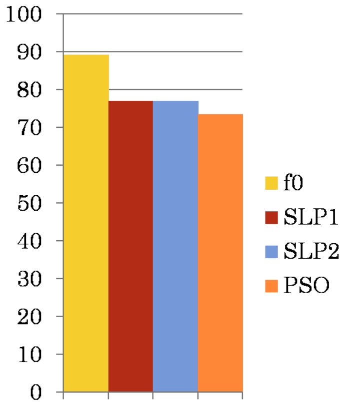
Application to a real energy district

The software package was applied to a real energetic district, in Pisa, Italy. It is composed by:

- an electric generator,
- a photovoltaic generator,
- a wind farm,
- a load connected to heating consumptions,
- a load connected to lighting consumptions,
- a thermal configuration.

The underlying optimization problem has 288 variables, 576 bound constraints and a non linear constraint.

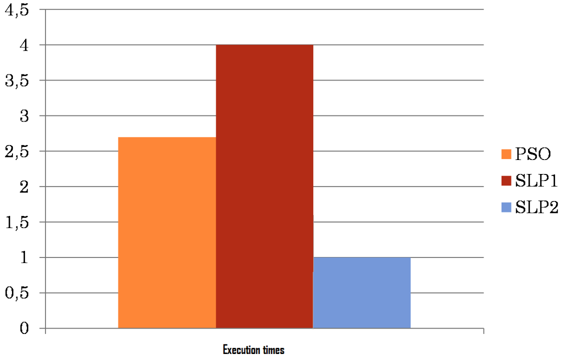
Comparison of the three solvers



- Just one nonlinear constraint: in a much less constrained problem PSO capacity of detecting the global minimum emerges: the constraints handling strategy could be improved.
- The use of the optimization package in this case allows 18% daily savings, with respect to the non-optimized management f_0.

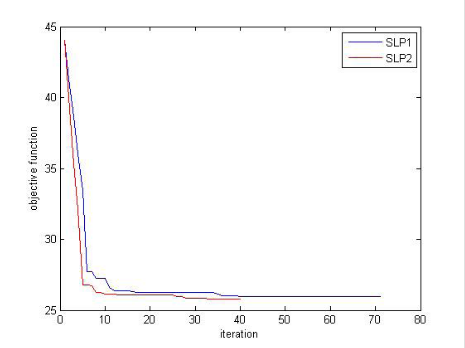
Execution time

In this case PSO execution time is comparable to that of SLP_2, and it is even shorter then that of SLP_1.



Stopping criterion

- SLP_1: maximum number of iteration reached, no criticality measure.
- SLP_2: Optimality measure thanks to Lagrangian multipliers estimates, based on KKT conditions.
- The use of an appropriate stopping criterion in SLP penalty approach (SLP_2) allows to save useless computations.



THANK YOU FOR YOUR ATTENTION!

Function l_k is non-differentiable, but we can rewrite the subproblem as the following equivalent smooth linear program, introducing the vector of slack variables t:

$$\min_{p,t} \nabla f(x_k)^T p + \nu \sum_{i \in \mathcal{I}} t_i;$$

$$\text{s.t.} g_i(x_k) + \nabla g_i(x_k)^T p \le t_i, \quad i \in \mathcal{I}$$

$$\max((x_{min} - x_k), -\Delta_k) \le p \le \min((x_{max} - x_k), \Delta_k),$$

$$t \ge 0.$$

Function l_k is non-differentiable, but we can rewrite the subproblem as the following equivalent smooth linear program, introducing the vector of slack variables t:

$$\min_{p,t} \nabla f(x_k)^T p + \nu \sum_{i \in \mathcal{I}} t_i;$$

$$\text{s.t.} g_i(x_k) + \nabla g_i(x_k)^T p \leq t_i, \quad i \in \mathcal{I}$$

$$\max((x_{min} - x_k), -\Delta_k) \leq p \leq \min((x_{max} - x_k), \Delta_k),$$

$$t \geq 0.$$

- SLP2:

$$\max\{\|\nabla f(x_k) + J(x_k)^T \lambda_k\|_\infty), \|g(x_k)^T \lambda_k\|_\infty\} < 10^{-3}(1 + \|\lambda_k\|_2),$$
$$\max\{\max(0, g_i(x_k)), \max(0, x_{\min} - x_k), \max(0, x_k - x_{\max})\} < 10^{-3}(1$$

- PSO: the function value does not decrease after 40 consecutive iterations:

$$\frac{|f(x_{k-2}) - f(x_k)|}{|f(x_{k-2})|} < 10^{-3}.$$