Globally convergent decomposition algorithm for risk parity problem in portfolio selection

A.Cassioli, G.Cocchi, F.D'Amato, M.Sciandrone February 11, 2016

1 Introduction

2 Preliminary background

Let us consider the following optimization problem:

$$\min_{x,y} \quad f(x,y) \tag{1a}$$

s.t.
$$l \le x \le u$$
 (1b)

$$\mathbf{1}^T x = 1 \tag{1c}$$

where $x \in \mathbb{R}^n$, $y \in \mathbb{R}^k$, f continuously differentiable, $l, u \in \mathbb{R}^n$ with l < u and $\mathbf{1} \in \mathbb{R}^n$ is all composed by ones.

Now we define the feasible set \mathcal{F} of Problem (1):

$$\mathcal{F} = \{ (x, y) \in \mathbb{R}^{n+k} : \mathbf{1}^T x = 1, l \le x \le u \}.$$
 (2)

Since the constraints of Problem (1) respect constraints qualification conditions, a feasible point (x,y) is a stationary point, if the Karush-Kuhn-Tucker (KKT) conditions are satisfied.

Let $L(x,y,\lambda,\mu,\gamma)$ the Lagrangian function associated to Problem (1) then we can write KKT conditions.

Proposition 2.1 (Optimality conditions (Necessary)) Let $(x^*, y^*) \in \mathbb{R}^{n+k}$, with $(x^*, y^*) \in \mathcal{F}$, a local optimum for Problem (1). Then there exist three multipliers $\lambda^* \in \mathbb{R}^n$, $\mu^* \in \mathbb{R}^n$, $\gamma^* \in \mathbb{R}$ such that:

$$\nabla_{x}L(x^{*}, y^{*}\lambda^{*}, \mu^{*}, \gamma^{*}) = \nabla_{x}F(x^{*}, y^{*}) + \lambda^{*} - \mu^{*} + \gamma^{*} = 0$$

$$\nabla_{x}L(x^{*}, y^{*}\lambda^{*}, \mu^{*}, \gamma^{*}) = \nabla_{y}F(x^{*}, y^{*}) = 0$$

$$\lambda_{i}^{*}(l - x_{i}^{*}) = 0, \ \forall i$$

$$\mu_{i}^{*}(x_{i}^{*} - u) = 0, \ \forall i$$

$$\lambda^{*}, \mu^{*} \geq 0$$
(3)

From the first condition we have:

$$\nabla_x f(x^*, y^*) - \lambda^* + \mu^* + \gamma^* = 0 \tag{4}$$

Then there are three possible cases:

$$\frac{\partial F(x^*, y^*)}{dx_i} = \begin{cases}
-\mu_i^* - \gamma & x_i^* = u \\
-\gamma + \lambda_i^* & x_i^* = l \\
-\gamma & l < x_i^* < u
\end{cases} \tag{5}$$

Then if $x_i^* > l$:

$$\frac{\partial f(x^*)}{\partial x_i} \le \frac{\partial f(x^*)}{\partial x_i}, \forall i \tag{6}$$

Let $(x,y) \in \mathcal{F}$, we define a set of all feasible direction in (x,y):

$$\mathcal{D}(x,y) = \{ d \in \mathbb{R}^{n+k} : \mathbf{1}^T d_x = 0, d_i \ge 0 \ \forall i \in L(x), d_i \le 0 \ \forall i \in U(x) \}$$
 (7)

where:

$$L(x) = \{i : x_i = l\}$$

$$U(x) = \{i : x_i = u\}$$
(8)

2.1 Set of sparse feasible directions

Because of y is unconstrainted, we pay attention only on the x variable to find a feasible descent direction w.r.t. x. In our case we want to build a set of sparse feasible direction in order to justify our decomposition approach.

Let $(x,y) \in \mathcal{F}$ non stationary w.r.t. x, then it's easy to see that

$$L(x) \neq \{1, \dots, n\} \tag{9}$$

hence $\exists i$ such that $x_i > 0$ and $j \neq i$ such that:

$$\frac{\partial f(x^*)}{dx_i} > \frac{\partial f(x^*)}{dx_j},\tag{10}$$

Now we define a direction $d^{i,j} \in \mathbb{R}^n$ with only two non-zero components such that:

$$d_h^{i,j} = \begin{cases} 1, & h = j \\ -1, & h = i \\ 0, & \text{otherwise} \end{cases}$$
 (11)

Proposition 2.2 Let (x,y) a feasible point for Problem (1). Then the direction $d^{i,j}$ is feasible and descent direction in x.

Proof For the feasibility it is enough to see that $\mathbf{1}^T d^{i,j} = 1 - 1 = 0$. Then we can apply sufficient conditions for descent direction in x, such that:

$$\nabla_x f(x, y)^T d^{i,j} = \frac{\partial f(x)}{\partial x_j} - \frac{\partial f(x)}{\partial x_i} < 0;$$

As always, we should choose the steepest descent direction composed by only two non-zero components. This can be done computing the Most Violating Pair (i,j) such that $x_i > 0$ and:

$$(i,j) \in \arg\min_{l,m} \left\{ \frac{\partial f(x^*)}{dx_l} - \frac{\partial f(x^*)}{dx_m} \right\}$$
 (12)

If one doesn't want to use decomposition methods, he can define a direction $d_{xy} \in \mathbb{R}^{n+k}$ such that:

$$d_{xy} = \{d^{i,j}, -\nabla_y f(x, y)\}$$
(13)

2.2 Armijo-Type Line Search Algorithm

In this section, we briefly describe the well-known Armijo-type line search along a feasible descent direction. The procedure will be used in the decomposition method presented in the next section. Let $d^k \in \mathcal{D}(x_k)$ $x^k \in \mathcal{F}$. In particular we choose $d^k = d_k^{i,j}$ with MVP (i(k), j(k)). We denote by Δ_k the maximum feasible step along d^k .

It is easy to see that:

$$\Delta_k = \min\{x_{j(k)}^k - l, u - x_{i(k)}^k\}$$

Then at iteration k+1 we have:

$$x_{j(k)}^{k+1} = \begin{cases} l & \alpha_k = x_{j(k)}^k - l \\ x_{j(k)}^k - u + x_{i(k)}^k & \alpha_k = u - x_{i(k)}^k \end{cases}$$

and:

$$x_{i(k)}^{k+1} = \begin{cases} x_{j(k)}^k - l + x_{i(k)}^k & \alpha_k = x_{j(k)}^k - l \\ u & \alpha_k = u - x_{i(k)}^k \end{cases}$$

Exact Line Search 2.3

When we move along the direction d^{i^*,j^*} , defined in (11), we modify only 2 variables (x_{i^*}, x_{i^*}) leaving the others unchanged. Thus, we can see our f(x, y)as a function of two components, i.e. we can rewrite Problem (1) as

$$\min_{x_{i^*}, x_{j^*}} f(x_{i^*}, x_{j^*})$$
s.t. $l_{i^*} \le x_{i^*} \le u_{i^*}$ (14a)

s.t.
$$l_{i^*} < x_{i^*} < u_{i^*}$$
 (14b)

$$l_{j^*} \le x_{j^*} \le u_{j^*} \tag{14c}$$

$$x_{i^*} + x_{j^*} = \underbrace{1 - \sum_{h \neq i^*, j^*} x_h}_{\text{(14d)}}$$

Thanks to the last constraint, we can substitute $x_{i^*} = c - x_{j^*}$ and then we obtain

$$\min_{x_{j^*}} \quad f(x_{j^*}) \tag{15a}$$

s.t.
$$x_{i^*} = c - x_{j^*}$$
 (15b)

$$ll_{j^*} = \max\{l_{j^*}, c - u_{i^*}\} \le x_{j^*} \le \min\{u_{j^*}, c - l_{i^*}\} = uu_{j^*}$$
 (15c)

Because the domain is $I = [ll_{j^*}, uu_{j^*}]$, if $f(x_{j^*})$ is continuous in I, then f has a minimum in I. If $f(x_{j^*})$ is differentiable in I we can compute $f'(x_{j^*})$. Let $R = \{r \mid f'(r) = 0, r \in I\}$ be set set of the real feasible roots of f'. Each $r \in R$ can be a local maximum, minimum or flex; if $R = \{\emptyset\}$, then the minimum of f is on the extreme points of I.

Let $r*=\arg\min_{r\in R}f(r)$, then the optimal step α^* along the direction d^{i^*,j^*} is

$$\alpha^* = x_{j^*} - r^* > 0 \tag{16}$$

TODO: problema di notazione, x_{j^*} rappresenta il valore della componente j^* del vettore x prima della ricerca di linea

3 A decomposition framework

The algorithm follows the Gauss-Seidel scheme with 2 blocks of variables (x and y). At each iteration, we optimize f w.r.t. one block of variables, considering the other block fixed.

The y-block of variables is unconstrained, so we can find

$$y^{(k+1)} = \arg\min_{y \in \mathbb{R}^m} f(x^{(k)}, y)$$
 (17)

using the first-order necessary condition, i.e. TODO: ma questo vale se f è convessa rispetto ad y

$$\nabla_y f(x^{(k)}, y^{(k)}) = 0 \tag{18}$$

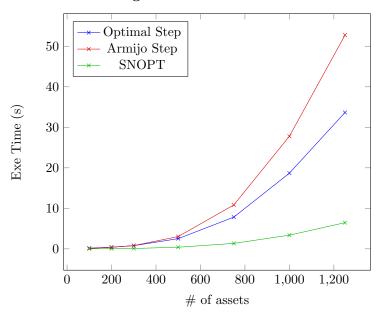
Algorithm 1: Decomposition Algorithm

Data: Given the initial feasible guess $(x^{(0)}, y^{(0)})$

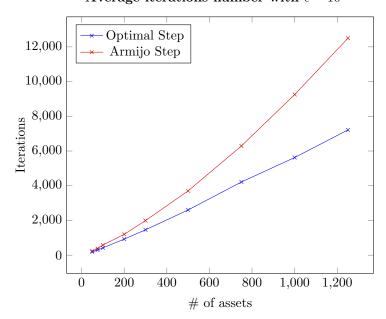
- ı Set k=0
- 2 while (not convergence) do
- **3** Compute $y^{(k+1)}$ as in (17)
- 4 Compute $\nabla_x f(x^{(k)}, y^{(k+1)})$
- 5 Choose indexes $i^*(k), j^*(k)$ using the Gauss-Southwell rules
- 6 Choose a step $\alpha^{(k)}$ along the direction $d^{i^*(k),j^*(k)}$
- 7 Set $x^{(k+1)} = x^{(k)} + \alpha^{(k)} d^{i^*(k), j^*(k)}$
- 8 Set k = k + 1

- 4 Convergence analysis
- 5 Computational experiments

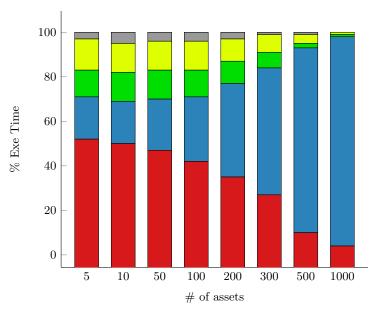
Average execution time with $\epsilon = 10^{-6}$



Average iterations number with $\epsilon=10^{-6}$



Armijo line search



Optimal step computation

