
AAR-based decomposition algorithm for non-linear convex optimisation

Nima Rabiei¹ · Jose J. Muñoz¹

Abstract In this paper we present a method for decomposing a class of convex non-linear programmes which are frequently encountered in engineering plastic analysis. These problems have second-order conic memberships constraints and a single complicating variable in the objective function. The method is based on finding the distance between the feasible sets of the decomposed problems, and updating the global optimal value according to the value of this distance. The latter is found by exploiting the method of averaged alternating reflections, which is here adapted to the optimisation problem at hand. The method is specially suited for non-linear problems and as our numerical results show, its convergence is independent of the number of variables of each sub-domain. We have tested the method with an illustrative example and with problems that have more than 10,000 variables.

Keywords Decomposition · Convex optimisation · Non-linear optimisation · Second-order cone program (SOCP) · Averaged alternating reflections (AAR)

1 Introduction

The analysis of structures with non-linear plastic materials may be posed as the solution of discrete non-linear optimisation problem. The form of this optimisation problem depends on the discretisation scheme and the underlying principle considered [8, 20–

✉ Jose J. Muñoz
j.munoz@upc.edu

Nima Rabiei
nima.rabiei@upc.edu

¹ Laboratory of Numerical Analysis (LaCàN), Universitat Politècnica de Catalunya, Barcelona, Spain

22,24]. This approach has attracted considerable attention in the last decade due to relevance of the results from the engineering standpoint, the accuracy that may be obtained, and recent progresses that have been achieved in non-linear optimisation. However, the analysis of general three-dimensional structures remains as yet prohibitive due to the size of the resulting optimisation problem, which may attain up to 500,000 primal variables. This work aims to propose a decomposition technique which can alleviate the memory requirements and computational cost of this type of problems.

The development of general decomposition techniques has given rise to numerous approaches, which include Benders decomposition [13,16], proximal point strategies [10], dual decomposition [7,15], sub-gradient and smoothing methods [25,26], or block decomposition [23], among many others. In the engineering literature, some common methods inherit either decomposition methods for elliptic problems [18], or proximal point strategies [19], or methods that couple the solutions from overlapping domains [28], which reduce their applicability.

The accuracy of dual decomposition and sub-gradient techniques strongly depend on the step-size control, while the accuracy of proximal point and smoothing techniques depend on the regularisation and smoothing parameters, which are problem dependent and not always easy to choose. Also, and from the experience of the authors, Benders methods have slow converge rates in non-linear optimisation problems due to the outer-linearisation process. These facts have motivated the development of the method presented here, which is specially suited for non-linear optimisation, and in particular exploits the structure of the problems encountered in engineering applications. We aim to solve a convex optimisation problems that can be written in the following form:

$$\lambda^* = \max_{\mathbf{x}_1, \mathbf{x}_2, \lambda} \lambda$$

$$f_1(\mathbf{x}_1, \lambda) = \mathbf{0} \quad (1a)$$

$$f_2(\mathbf{x}_2, \lambda) = \mathbf{0} \quad (1b)$$

$$g_1(\mathbf{x}_1) + g_2(\mathbf{x}_2) = \mathbf{0} \quad (1c)$$

$$\mathbf{x}_1 \in K_1 \subseteq \mathbb{R}^{n_1}, \mathbf{x}_2 \in K_2 \subseteq \mathbb{R}^{n_2}, \lambda \in \mathbb{R}. \quad (1d)$$

where $f_1 : \mathbb{R}^{n_1} \times \mathbb{R} \rightarrow \mathbb{R}^{m_1}$, $f_2 : \mathbb{R}^{n_2} \times \mathbb{R} \rightarrow \mathbb{R}^{m_2}$, $g_1 : \mathbb{R}^{n_1} \rightarrow \mathbb{R}^m$ and $g_2 : \mathbb{R}^{n_2} \rightarrow \mathbb{R}^m$ are given affine functions, and K_1, K_2 are non-empty closed convex sets.

The optimisation problem in (1) has one important feature, which is a requirement of the method presented here: the objective function contains one scalar (global) variable λ , and other (local) variables \mathbf{x}_1 and \mathbf{x}_2 . We remark though that other problems with a linear objective function that has more than one variable may be also posed in the form given above, and therefore may be also solved with the method proposed in this paper. For instance, we consider a new problem with the same constraints as in (1), and whose objective function is equal to $h_1(\mathbf{x}_1, \lambda) + h_2(\mathbf{x}_2, \lambda)$, i.e. we aim to solve the following problem

$$\lambda^* = \max_{\mathbf{x}_1, \mathbf{x}_2, \lambda} h_1(\mathbf{x}_1, \lambda) + h_2(\mathbf{x}_2, \lambda)$$

(1a), (1b), (1c), and (1c),

where h_1 and h_2 are linear functions. We can convert this problem in the form of problem (1) by recasting it in the equivalent form:

$$\begin{aligned} \lambda^* = \max_{\mathbf{x}_1, \mathbf{x}_2, \lambda_0, \lambda_1, \lambda_2} \quad & \lambda_0 \\ & f_1(\mathbf{x}_1, \lambda_1) = \mathbf{0} \\ & \lambda_0 - \lambda_1 = \mathbf{0} \\ & f_2(\mathbf{x}_2, \lambda_2) = \mathbf{0} \\ & \lambda_0 - \lambda_2 = \mathbf{0} \\ & g_1(\mathbf{x}_1) + g_2(\mathbf{x}_2) = \mathbf{0} \\ & \lambda_1 - h_1(\mathbf{x}_1, \lambda_1) - h_2(\mathbf{x}_2, \lambda_2) = 0 \\ & \mathbf{x}_1 \in K_1, \mathbf{x}_2 \in K_2, \lambda_0, \lambda_1, \lambda_2 \in \mathbb{R}. \end{aligned}$$

In this new form, λ_1 and λ_2 are local variables, while λ_0 plays the role of a global variable.

We also point out that the particular form in (1) is a common feature in some problems in engineering such as limit analysis [21, 22, 24] or general plastic analysis [18–20], where λ measures the bearing capacity of a structure or the dissipation power when it collapses. The primal problem in (1) is written as a maximisation of the objective function, in agreement with the engineering applications, but in contrast to the standard notation in optimisation. We will keep the form in (1), but of course, the algorithm explained in this paper may be also described using standard notation.

The main contributions of the paper are: (i) rewriting the constraints in (1) as the intersection of appropriate sets, (ii) decomposing this form of the algorithm into a master problem and two sub-problems, (iii) applying some results of proximal point theory to this new form of the optimisation problem, and (iv) proposing efficient algorithmic strategies to iteratively solve the decomposition algorithm. We prove the convergence properties of the algorithm, and numerically test its efficiency.

Section 2 recalls some well-known results from approximation theory and summarises the method of Averaged Alternating Reflections to find the distance between two sets. The proposed decomposition method is described in Sect. 3, and Sect. 4 tests the method with some representative non-linear problems.

2 Method of averaged alternating reflections

2.1 Best approximation operators

Throughout this paper W and Z are two nonempty closed convex sets in the real space \mathbb{R}^n , with an inner product and an induced norm $\|\cdot\|$. Given a point $\mathbf{x} \in \mathbb{R}^n$, let I denote the identity operator on \mathbb{R}^n , and let $P_W(\mathbf{x})$ and $P_Z(\mathbf{x})$ be the projections (best

approximation operators) onto W and Z . The standard best approximation of \mathbf{x} relative to W is the solution of the following problem [14]:

$$\text{find } \mathbf{w} \in W \text{ such that } \|\mathbf{x} - \mathbf{w}\| = \inf \|\mathbf{x} - W\| := d(\mathbf{x}, W). \quad (2)$$

A natural extension of this problem is to find a best approximation pair relative to (W, Z) . i.e., to

$$\text{find } (\mathbf{w}, \mathbf{z}) \in W \times Z \text{ such that } \|\mathbf{w} - \mathbf{z}\| = \inf \|W - Z\| := d(W, Z). \quad (3)$$

If $W = \{\mathbf{w}\}$, (3) reduces to (2) and its solution is $P_Z(\mathbf{w})$. On the other hand, when the problem is consistent, i.e., $W \cap Z \neq \emptyset$, then (3) reduces to the well-known convex feasibility problem for two sets [4, 12] and its solution set is $\{(\mathbf{x}, \mathbf{x}) \in \mathbb{R}^n \times \mathbb{R}^n | \mathbf{x} \in W \cap Z\}$. The formulation in (3) captures a wide range of problems in applied mathematics and engineering [11, 17, 30].

In Sect. 3 we will rewrite problem (1) as a problem of finding the minimum distance $d(W, Z)$ between two feasibility sets W and Z , as stated in (3). Prior to that though, in the remainder of this section, we present a methodology to find the distance between two sets which will be eventually employed.

2.2 Preliminary definitions

Definition 1 The set of fixed points of an operator $T : X \rightarrow X$ is denoted by $\text{Fix } T$, i.e.,

$$\text{Fix } T = \{x \in X | T(x) = x\}. \quad (4)$$

It is convenient to introduce the following sets, which we will use throughout this section:

$$\begin{aligned} C &= Z - W = \{\mathbf{z} - \mathbf{w} | \mathbf{z} \in Z, \mathbf{w} \in W\}, \\ \mathbf{v} &= P_C(0), \quad G = W \cap (Z - \mathbf{v}), \quad H = (W + \mathbf{v}) \cap Z. \end{aligned} \quad (5)$$

Note also that if $W \cap Z \neq \emptyset$, then $G = H = W \cap Z$. However, even when $W \cap Z = \emptyset$, the sets G and H may be nonempty and they serve as substitutes for the intersection. In words, vector \mathbf{v} joins the two sets Z and W at the point that are at the minimum distance and $\|\mathbf{v}\|$ measures the gap between the sets W and Z .

Lemma 1 From the definitions in (4)–(5), the following identities hold:

- (i) $\|\mathbf{v}\| = \inf \|W - Z\|$
- (ii) $G = \text{Fix } (P_W P_Z)$ and $H = \text{Fix } (P_Z P_W)$.
- (iii) $G + \mathbf{v} = H$.

The proof can be found in [3], Sect. 5.

Lemma 2 Suppose that $(\mathbf{w}_n)_{n \in \mathbb{N}}$ and $(\mathbf{z}_n)_{n \in \mathbb{N}}$ are sequences in W and Z , respectively. Then

$$\mathbf{z}_n - \mathbf{w}_n \rightarrow \mathbf{v} \iff \|\mathbf{z}_n - \mathbf{w}_n\| \rightarrow \|\mathbf{v}\|.$$

Also, assume that $\mathbf{z}_n - \mathbf{w}_n \rightarrow \mathbf{v}$. Then the following identities hold:

- (i) $\mathbf{z}_n - P_W(\mathbf{z}_n) \rightarrow \mathbf{v}$, and $P_Z(\mathbf{w}_n) - \mathbf{w}_n \rightarrow \mathbf{v}$.
- (ii) The weak cluster points of $(\mathbf{w}_n)_{n \in N}$ and $(P_W(\mathbf{z}_n))_{n \in N}$ (resp. $(\mathbf{z}_n)_{n \in N}$ and $(P_Z(\mathbf{w}_n))_{n \in N}$) belong to G (resp. H). Consequently, the weak cluster points of the sequences

$$((\mathbf{w}_n, \mathbf{z}_n))_{n \in N}, ((\mathbf{w}_n), P_Z(\mathbf{w}_n))_{n \in N}, ((P_W(\mathbf{z}_n), \mathbf{z}_n))_{n \in N}$$

are best approximation pairs relative to (W, Z) .

- (iii) If $G = \emptyset$ or, equivalently, $H = \emptyset$, then

$$\min\{\|\mathbf{w}_n\|, \|\mathbf{z}_n\|, \|P_W(\mathbf{z}_n)\|, \|P_Z(\mathbf{w}_n)\|\} \rightarrow \infty.$$

These results are proved in [6].

Definition 2 Let D be a nonempty subset of \mathbb{R}^n and let $T : D \rightarrow \mathbb{R}^n$. Then T is

- (i) firmly nonexpansive if

$$(\forall \mathbf{x} \in D), (\forall \mathbf{y} \in D) : \|T(\mathbf{x}) - T(\mathbf{y})\|^2 + \|(I - T)(\mathbf{x}) - (I - T)(\mathbf{y})\|^2 \leq \|\mathbf{x} - \mathbf{y}\|^2,$$

- (ii) nonexpansive if

$$(\forall \mathbf{x} \in D), (\forall \mathbf{y} \in D) : \|T(\mathbf{x}) - T(\mathbf{y})\| \leq \|\mathbf{x} - \mathbf{y}\|.$$

Lemma 3 Let W be a nonempty closed convex subset of \mathbb{R}^n . Then

- (i) the projector P_W is firmly nonexpansive;
- (ii) $2P_W - I$ is nonexpansive.

See [5] for a proof of this lemma. The transformation $2P_W - I$ is named the reflection operator with respect to W and will be denoted by R_W .

Lemma 4 Let $T_1 : \mathbb{R}^n \rightarrow \mathbb{R}^n$ and $T_2 : \mathbb{R}^n \rightarrow \mathbb{R}^n$ be firmly nonexpansive and set

$$T = \frac{(2T_1 - I)(2T_2 - I) + I}{2}.$$

Then the following results hold:

- (i) T is firmly nonexpansive.
- (ii) $\text{Fix } T = \text{Fix } (2T_1 - I)(2T_2 - I)$.

See [5] for a proof of this lemma.

2.3 Averaged alternating reflections (AAR)

Definition 3 We define the so-called averaged alternating reflections (AAR) operator, denoted by T and given by,

$$T = \frac{R_W R_Z + I}{2}. \quad (6)$$

where $R_W = 2P_W - I$ and $R_Z = 2P_Z - I$. In view of Lemmas 3 and 4, and since P_Z and P_W are firmly nonexpansive, we infer that T is nonexpansive and

$$\text{Fix } T = \text{Fix } R_W R_Z.$$

Proposition 1 *Let W and Z be nonempty closed convex subsets of \mathbb{R}^n and let T be the operator in (6). Then the following results hold:*

- (i) $I - T = P_Z - P_W R_Z$.
- (ii) $W \cap Z \neq \emptyset$ if and only if $\text{Fix } T \neq \emptyset$.

Proof (i)

$$\begin{aligned} T &= \frac{R_W R_Z + I}{2} = \frac{2P_W R_Z - R_Z + I}{2} = \frac{2P_W R_Z - 2P_Z + I + I}{2} \\ &= P_W R_Z - P_Z + I \Rightarrow I - T = P_Z - P_W R_Z. \end{aligned}$$

- (ii) Assume that $\mathbf{x} \in W \cap Z$. Clearly, we then have that $P_W(\mathbf{x}) = \mathbf{x}$ and $P_Z(\mathbf{x}) = \mathbf{x}$, which in turn imply that $R_W(\mathbf{x}) = \mathbf{x}$ and $R_Z(\mathbf{x}) = \mathbf{x}$. Using the definition in (6), it follows that

$$T(\mathbf{x}) = \mathbf{x},$$

so that $\text{Fix } T \neq \emptyset$.

Conversely, if $\mathbf{x} \in \text{Fix } T$, then $T(\mathbf{x}) = \mathbf{x}$, and then according to (i), we have, that $P_Z(\mathbf{x}) = P_W R_Z(\mathbf{x})$, and therefore $P_Z(\mathbf{x}) \in Z$ and $P_Z(\mathbf{x}) \in W$, which is equivalent to $P_Z(\mathbf{x}) \in W \cap Z$. \square

We now recall the well known convergence results for the method of Averaged Alternating Reflections (AAR).

Lemma 5 (Convergence of AAR method) *Consider the following successive approximation method: Take $\mathbf{t}_0 \in \mathbb{R}^n$, and set*

$$\mathbf{t}_n = T^n(\mathbf{t}_0) = T(\mathbf{t}_{n-1}), \quad n = 1, 2, \dots$$

where T is defined in (6), and W, Z are nonempty closed convex subsets of \mathbb{R}^n . Then the following results hold

- (i) $\text{Fix } T \neq \emptyset \iff (T^n(\mathbf{t}_0))_{n \in \mathbb{N}}$ converges to some point in $\text{Fix } T$.

- (ii) Fix $T = \emptyset \iff \|T^n(\mathbf{t}_0)\| \rightarrow \infty$, when $n \rightarrow \infty$.
- (iii) $(\|P_Z(\mathbf{t}_n) - P_W P_Z(\mathbf{t}_n)\|)$ converges to $\inf \|W - Z\|$, and $(\|P_Z(\mathbf{t}_n) - P_W R_Z(\mathbf{t}_n)\|)$ converges to $\inf \|W - Z\|$.
- (iv) $\frac{\|\mathbf{t}_n\|}{n} \rightarrow \inf \|W - Z\|$.

Proof (i) and (ii) are demonstrated in [2, 9, 27] and (iii) in [6], while (iv) is demonstrated in [29].

3 Decomposition algorithm

3.1 Alternative definition of global feasibility region

The objective of this section is to provide a description of a decomposition method that exploits the results of the AAR method given above. The method is suitable for convex optimisation problems that have the structure given in (1), which we aim to transform into the computation of the minimal distance between two feasible sets. For this aim, we rewrite problem (1) by introducing a new complicating variable \mathbf{t} as follows:

$$\begin{aligned}
 \lambda^* &= \max_{\mathbf{x}_1, \mathbf{x}_2, \lambda, \mathbf{t}} \lambda \\
 f_1(\mathbf{x}_1, \lambda) &= 0 \\
 f_2(\mathbf{x}_2, \lambda) &= 0 \\
 g_1(\mathbf{x}_1) &= \mathbf{t} \\
 g_2(\mathbf{x}_2) &= -\mathbf{t} \\
 \mathbf{x}_1 &\in K_1, \mathbf{x}_2 \in K_2, \lambda \in \mathbb{R}.
 \end{aligned} \tag{7}$$

We next define the feasibility region of this problem with the help of the following definitions:

Definition 4 Consider the following two feasibility regions:

$$\begin{aligned}
 X_1(\lambda) &= \{\mathbf{x}_1 | f_1(\mathbf{x}_1, \lambda) = 0\} \cap K_1, \\
 X_2(\lambda) &= \{\mathbf{x}_2 | f_2(\mathbf{x}_2, \lambda) = 0\} \cap K_2,
 \end{aligned} \tag{8}$$

and also let $\lambda = \bar{\lambda}$ be a given real value. Then, we define the following feasibility sets for variable \mathbf{t} :

$$\begin{aligned}
 Z(\bar{\lambda}) &= g_1(X_1(\bar{\lambda})) = \{g_1(\mathbf{x}_1) | \mathbf{x}_1 \in X_1(\bar{\lambda})\}, \\
 W(\bar{\lambda}) &= -g_2(X_2(\bar{\lambda})) = \{-g_2(\mathbf{x}_2) | \mathbf{x}_2 \in X_2(\bar{\lambda})\}.
 \end{aligned} \tag{9}$$

Throughout the subsequent sections, the sets $Z(\lambda)$ and $W(\lambda)$ defined in (9) are assumed convex, (which can be ensured through the convexity of g_1 and g_2).

By using definitions (8) and (9), the optimisation problem in (7) may be recast as,

$$\begin{aligned}
 \lambda^* &= \max_{\lambda} \lambda \\
 Z(\lambda) \cap W(\lambda) &\neq \emptyset
 \end{aligned}$$

The algorithm proposed in this paper is based on the form above. In brief, the algorithm consists on updating the value of $\bar{\lambda}$ (master problem) and analysing in the sub-problems whether the intersection between the sets $Z(\bar{\lambda})$ and $W(\bar{\lambda})$ is empty or not. In order to determine this and compute upper bounds of the global problem in (7), we will need the following two propositions:

Proposition 2 *Let λ_0 and $\bar{\lambda}$ be given real values such that $(\mathbf{x}_{01}, \mathbf{x}_{02}, \mathbf{t}_0, \lambda_0)$ is a feasible solution for problem (7), and $\lambda_0 < \bar{\lambda}$. After using the definition in (9), the following relation holds:*

$$Z(\bar{\lambda}) \cap W(\bar{\lambda}) \neq \emptyset \iff \bar{\lambda} \leq \lambda^*.$$

Proof First suppose that $Z(\bar{\lambda}) \cap W(\bar{\lambda}) \neq \emptyset$ and $\bar{\mathbf{t}}$ belongs to $Z(\bar{\lambda}) \cap W(\bar{\lambda})$, thus there exist $\mathbf{x}_1 \in X_1(\bar{\lambda})$ and $\mathbf{x}_2 \in X_2(\bar{\lambda})$ such that $g_1(\mathbf{x}_1) = \bar{\mathbf{t}}$ and $-g_2(\mathbf{x}_2) = \bar{\mathbf{t}}$. Therefore, in view of (8), $(\mathbf{x}_1, \mathbf{x}_2, \bar{\lambda}, \bar{\mathbf{t}})$ is a feasible solution for problem (7), and consequently $\bar{\lambda} \leq \lambda^*$.

Conversely, let $\lambda_0 < \bar{\lambda} \leq \lambda^*$. Hence, there exists $\gamma \in (0, 1]$ such that $\bar{\lambda} = (1 - \gamma)\lambda_0 + \gamma\lambda^*$. Since $(\mathbf{x}_{01}, \mathbf{x}_{02}, \mathbf{t}_0, \lambda_0)$ and $(\mathbf{x}_1^*, \mathbf{x}_2^*, \mathbf{t}^*, \lambda^*)$ are feasible solutions for problem (7) and since the feasible region of problem (7) is convex, it follows that the convex combination of these two points is a feasible solution for problem (7). Formally, we have that

$$\begin{aligned} & (1 - \gamma)(\mathbf{x}_{01}, \mathbf{x}_{02}, \mathbf{t}_0, \lambda_0) + \gamma(\mathbf{x}_1^*, \mathbf{x}_2^*, \mathbf{t}^*, \lambda^*) \\ &= ((1 - \gamma)\mathbf{x}_{01} + \gamma\mathbf{x}_1^*, (1 - \gamma)\mathbf{x}_{02} + \gamma\mathbf{x}_2^*, (1 - \gamma)\mathbf{t}_0 + \gamma\mathbf{t}^*, (1 - \gamma)\lambda_0 + \gamma\lambda^*) \\ &= ((1 - \gamma)\mathbf{x}_{01} + \gamma\mathbf{x}_1^*, (1 - \gamma)\mathbf{x}_{02} + \gamma\mathbf{x}_2^*, (1 - \gamma)\mathbf{t}_0 + \gamma\mathbf{t}^*, \bar{\lambda}), \end{aligned}$$

which shows that $(1 - \gamma)\mathbf{t}_0 + \gamma\mathbf{t}^*$ belongs to $Z(\bar{\lambda}) \cap W(\bar{\lambda})$, i.e. $Z(\bar{\lambda}) \cap W(\bar{\lambda}) \neq \emptyset$. \square

Proposition 3 *Let $(\bar{\mathbf{t}}, \bar{\lambda})$ be an arbitrary given vector and $\Delta \bar{s}_i$ be an optimal solution of the following optimisation problems:*

$$\begin{aligned} & \max_{\mathbf{x}_i, \Delta \mathbf{w}_i, \Delta s_i} \Delta s_i \\ & f_i(\mathbf{x}_i, \bar{\lambda} + \Delta s_i) = \mathbf{0} \\ & g_i(\mathbf{x}_i) = (-1)^{i+1}(\bar{\mathbf{t}} + \Delta \mathbf{w}_i) \\ & \mathbf{x}_i \in K_i, \Delta \mathbf{w}_i \in \mathbb{R}^{n_m}, \Delta s_i \in \mathbb{R}. \end{aligned} \tag{10}$$

with $i=1,2$. Then $\lambda^* \leq \bar{\lambda} + \Delta \bar{s}_i$ for $i = 1, 2$.

Proof There exists a real value Δs^* and a vector $\Delta \mathbf{w}^*$ such that

$$\begin{aligned} \lambda^* &= \bar{\lambda} + \Delta s^*; \\ \mathbf{t}^* &= \bar{\mathbf{t}} + \Delta \mathbf{w}^*. \end{aligned} \tag{11}$$

Since $(\mathbf{x}_1^*, \mathbf{x}_2^*, \mathbf{t}^*, \lambda^*)$ is a feasible solution for problem (7), and in view of (11) we have that

$$\begin{cases} f_i(\mathbf{x}_i^*, \lambda^*) = 0 \\ g_i(\mathbf{x}_i^*) = (-1)^{i+1} \mathbf{t}^* \\ \mathbf{x}_i^* \in K_i \end{cases} \Rightarrow \begin{cases} f_i(\mathbf{x}_i^*, \bar{\lambda} + \Delta s^*) = 0 \\ g_i(\mathbf{x}_i^*) = (-1)^{i+1} (\bar{\mathbf{t}} + \Delta \mathbf{w}^*) \\ \mathbf{x}_i^* \in K_i, \Delta \mathbf{w}^* \in \mathbb{R}^{n_m}, \Delta \mathbf{s}^* \in \mathbb{R}. \end{cases}$$

Since $\Delta \bar{s}_i$ is an optimal solution of problem (10) and in view of (11) then

$$\Delta s^* \leq \Delta \bar{s}_i \Rightarrow \bar{\lambda} + \Delta s^* \leq \bar{\lambda} + \Delta \bar{s}_i \Rightarrow \lambda^* \leq \bar{\lambda} + \Delta \bar{s}_i.$$

□

It will become convenient to consider the dual form of the global problem (1),

$$q^* = \inf_{\mathbf{y}_1, \mathbf{y}_2, \mathbf{y}_3} q(\mathbf{y}_1, \mathbf{y}_2, \mathbf{y}_3) \quad (12)$$

where

$$\begin{aligned} q(\mathbf{y}_1, \mathbf{y}_2, \mathbf{y}_3) = & \sup_{\mathbf{x}_1, \mathbf{x}_2, \lambda} \mathbf{y}_1 \cdot f_1(\mathbf{x}_1, \lambda) + \mathbf{y}_2 \cdot f_2(\mathbf{x}_2, \lambda) \\ & + \mathbf{y}_3 \cdot (g_1(\mathbf{x}_1) + g_2(\mathbf{x}_2)) + \lambda \\ & \mathbf{x}_1 \in K_1, \mathbf{x}_2 \in K_2, \lambda \in \mathbb{R}. \end{aligned} \quad (13)$$

3.2 Definition of sub-problems

Let λ_0 and $\bar{\lambda}$ be given real values such that $(\mathbf{x}_{01}, \mathbf{x}_{02}, \lambda_0)$ is a feasible solution for (1) and $\lambda_0 < \bar{\lambda}$, which means that $W(\bar{\lambda})$ and $Z(\bar{\lambda})$ are nonempty (closed convex) sets. Take \mathbf{t}_0 and set

$$\mathbf{t}_n = T^n(\mathbf{t}_0) = T(\mathbf{t}_{n-1}), \quad n = 1, 2, 3, \dots,$$

where

$$T = \frac{R_W R_Z + I}{2} = P_W R_Z - P_Z + I. \quad (14)$$

We next define the optimisation sub-problems that will allow us to retrieve the projections P_Z and P_W , required for computing the transformation T . In view of (2), $P_Z(\mathbf{t}_n)$ can be obtained by solving the following optimisation problem:

$$\min_{\mathbf{z} \in Z(\bar{\lambda})} \|\mathbf{z} - \mathbf{t}_n\|$$

which is equivalent to the following so-called Sub-problem 1 :

$$\begin{aligned} \min_{\mathbf{x}_1, \mathbf{d}^1} & \|\mathbf{d}^1\| \\ & f_1(\mathbf{x}_1, \bar{\lambda}) = 0 \\ & g_1(\mathbf{x}_1) - \mathbf{d}^1 = \mathbf{t}_n \\ & \mathbf{x}_1 \in K_1. \end{aligned} \quad (15)$$

From the optimal solution of Sub-problem 1, \mathbf{d}_n^1 , we can compute the projection $P_Z(\mathbf{t}_n)$ and reflection $R_Z(\mathbf{t}_n)$ as,

$$\begin{aligned} P_Z(\mathbf{t}_n) &= g_1(\mathbf{x}_{1n}) = \mathbf{t}_n + \mathbf{d}_n^1, \text{ with } \mathbf{x}_{1n} \in X_1(\bar{\lambda}), \\ R_Z(\mathbf{t}_n) &= 2P_Z(\mathbf{t}_n) - \mathbf{t}_n = \mathbf{t}_n + 2\mathbf{d}_n^1. \end{aligned} \quad (16)$$

From $P_Z(\mathbf{t}_n)$, the point $P_W R_Z(\mathbf{t}_n) = P_W(\mathbf{t}_n + 2\mathbf{d}_n^1)$ is obtained by solving the following optimisation problem:

$$\begin{aligned} \min_{\mathbf{w}} & \|\mathbf{w} - R_Z(\mathbf{t}_n)\| \\ & \mathbf{w} \in W(\bar{\lambda}) \end{aligned}$$

which is equivalent to the following so-called Sub-problem 2:

$$\begin{aligned} \min_{\mathbf{x}_2, \mathbf{d}^2} & \|\mathbf{d}^2\| \\ & f_2(\mathbf{x}_2, \bar{\lambda}) = 0 \\ & g_2(\mathbf{x}_2) + \mathbf{d}^2 = -R_Z(\mathbf{t}_n) \\ & \mathbf{x}_2 \in K_2. \end{aligned} \quad (17)$$

After solving this problem we have that,

$$\begin{aligned} P_W R_Z(\mathbf{t}_n) &= -g_2(\mathbf{x}_{2n}) = R_Z(\mathbf{t}_n) + \mathbf{d}_n^2 = \mathbf{t}_n + 2\mathbf{d}_n^1 + \mathbf{d}_n^2, \text{ with } \mathbf{x}_{2n} \in X_2(\bar{\lambda}) \\ R_W R_Z(\mathbf{t}_n) &= 2P_W(R_Z(\mathbf{t}_n)) - R_Z(\mathbf{t}_n) = \mathbf{t}_n + 2\mathbf{d}_n^1 + 2\mathbf{d}_n^2, \end{aligned} \quad (18)$$

and according to (14), (16) and (18),

$$\mathbf{t}_{n+1} = T(\mathbf{t}_n) = \mathbf{t}_n + \mathbf{d}_n^1 + \mathbf{d}_n^2, \quad (19)$$

with \mathbf{d}_n^1 and \mathbf{d}_n^2 optimal solutions of (15) and (17), respectively. This iterative process and the associated projections and reflections are illustrated in Fig. 1. Since T is nonexpansive, and in view of (14), (16), (18) and (19), we have the following results,

$$\begin{aligned} \text{(i)} \quad \mathbf{t}_{n+1} - \mathbf{t}_n &= \mathbf{d}_n^1 + \mathbf{d}_n^2 = T(\mathbf{t}_n) - \mathbf{t}_n \\ &= P_W(R_Z(\mathbf{t}_n)) - P_Z(\mathbf{t}_n) = -g_2(\mathbf{x}_{2n}) - g_1(\mathbf{x}_{1n}). \end{aligned} \quad (20)$$

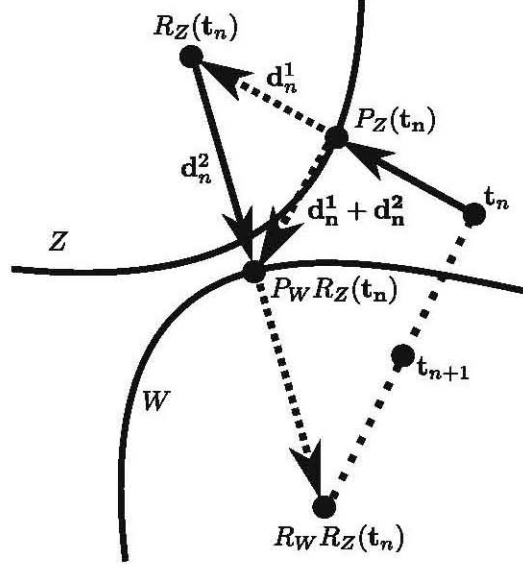
$$\begin{aligned} \text{(ii)} \quad \|\mathbf{d}_n^1 + \mathbf{d}_n^2\| &= \|\mathbf{t}_{n+1} - \mathbf{t}_n\| = \|T(\mathbf{t}_n) - T(\mathbf{t}_{n-1})\| \\ &\leq \|\mathbf{t}_n - \mathbf{t}_{n-1}\| = \|\mathbf{d}_{n-1}^1 + \mathbf{d}_{n-1}^2\|. \end{aligned} \quad (21)$$

According to (21), the sequence $(\|\mathbf{d}_n^1 + \mathbf{d}_n^2\|)_{n \in \mathbb{N}}$ is decreasing. Therefore

$$\alpha = \inf_{n \geq 1} \|\mathbf{d}_n^1 + \mathbf{d}_n^2\| = \lim_{n \rightarrow \infty} \|\mathbf{d}_n^1 + \mathbf{d}_n^2\| \in [0, \infty). \quad (22)$$

The value of α measures the distance $d(W(\bar{\lambda}), Z(\bar{\lambda}))$. The next theorem relates α to the optimal objective λ^* :

Fig. 1 Illustration of the iterative process



Theorem 1 Consider Sub-problem 1 and Sub-problem 2 defined in (15) and (17) respectively, and λ^* the optimal solution of the global problem in (1). Then, with α defined in (22), and if $\lambda^* = q^*$, i.e. there is no duality gap in (12), the following implications hold:

- (i) $\alpha = 0$ if and only if $\bar{\lambda} \leq \lambda^*$,
- (ii) $\alpha > 0$ if and only if $\bar{\lambda} > \lambda^*$.

Proof (i): If $\alpha = 0$, we have from (20) that

$$\lim_{n \rightarrow \infty} \|\mathbf{d}_n^1 + \mathbf{d}_n^2\| = \lim_{n \rightarrow \infty} \|g_1(\mathbf{x}_{1n}) + g_2(\mathbf{x}_{2n})\| = 0. \quad (23)$$

Since $(\mathbf{x}_{1n}, \mathbf{x}_{2n}) \in X_1(\bar{\lambda}) \times X_2(\bar{\lambda})$, by setting $\lambda_n = \bar{\lambda}$, for all $n \in N$, the vector $(\mathbf{x}_{1n}, \mathbf{x}_{2n}, \lambda_n)$ satisfies the following conditions:

$$\begin{aligned} f_1(\mathbf{x}_{1n}, \lambda_n) &= 0 \\ f_2(\mathbf{x}_{2n}, \lambda_n) &= 0 \\ \mathbf{x}_{1n} &\in K_1, \mathbf{x}_{2n} \in K_2. \end{aligned} \quad (24)$$

Suppose that $(\mathbf{y}_1, \mathbf{y}_2, \mathbf{y}_3)$ is an arbitrary feasible solution for the dual problem in (12). Since $(\mathbf{x}_{1n}, \mathbf{x}_{2n}, \lambda_n) \in X_1(\bar{\lambda}) \times X_2(\bar{\lambda}) \times \mathbb{R}$, according to (13) we have that

$$\begin{aligned} q(\mathbf{y}_1, \mathbf{y}_2, \mathbf{y}_3) &\geq \mathbf{y}_1 \cdot f_1(\mathbf{x}_{1n}, \lambda_n) + \mathbf{y}_2 \cdot f_2(\mathbf{x}_{2n}, \lambda_n) \\ &\quad + \mathbf{y}_3 \cdot (g_1(\mathbf{x}_{1n}) + g_2(\mathbf{x}_{2n})) + \lambda_n \\ &= \mathbf{y}_3 \cdot (g_1(\mathbf{x}_{1n}) + g_2(\mathbf{x}_{2n})) + \lambda_n, \end{aligned}$$

and then

$$q(\mathbf{y}_1, \mathbf{y}_2, \mathbf{y}_3) \geq \lim_{n \rightarrow \infty} \mathbf{y}_3 \cdot (g_1(\mathbf{x}_{1n}) + g_2(\mathbf{x}_{2n})) + \lambda_n.$$

Therefore in view of (23), we have that $q(\mathbf{y}_1, \mathbf{y}_2, \mathbf{y}_3) \geq \bar{\lambda}$. On the other hand, since $(\mathbf{y}_1, \mathbf{y}_2, \mathbf{y}_3)$ is an arbitrary feasible solution for the dual problem, and due to assumption $q^* = \lambda^*$, we have the following result :

$$\bar{\lambda} \leq \inf_{\mathbf{y}_1, \mathbf{y}_2, \mathbf{y}_3} q(\mathbf{y}_1, \mathbf{y}_2, \mathbf{y}_3) = q^* = \lambda^* \Rightarrow \bar{\lambda} \leq \lambda^*.$$

Conversely, assume $\bar{\lambda} \leq \lambda^*$. Thus in view of Lemma 5 and Propositions 1 and 2, we can infer that the sequence $(\mathbf{t}_n)_{n \in \mathbb{N}}$ converges to a point in $\text{Fix } T$, i.e.

$$\lim_{n \rightarrow \infty} \mathbf{t}_n = \bar{\mathbf{t}} \in \text{Fix } T.$$

Since $\mathbf{t}_{n+1} - \mathbf{t}_n = \mathbf{d}_n^1 + \mathbf{d}_n^2$, we deduce that,

$$\lim_{n \rightarrow \infty} \mathbf{d}_n^1 + \mathbf{d}_n^2 = \lim_{n \rightarrow \infty} \mathbf{t}_{n+1} - \mathbf{t}_n = \bar{\mathbf{t}} - \bar{\mathbf{t}} = 0 \Rightarrow \alpha = \lim_{n \rightarrow \infty} \|\mathbf{d}_n^1 + \mathbf{d}_n^2\| = 0.$$

(ii) Since $\alpha \geq 0$, the result in (ii) follows from (i). \square

The result of Theorem 1 is illustrated in Fig. 2, which represents the distance of the sets $Z(\bar{\lambda})$ and $W(\bar{\lambda})$ for the cases $\bar{\lambda} \leq \lambda^*$ and $\bar{\lambda} > \lambda^*$. Figure 3 also shows the same idea but on the (λ, \mathbf{t}) plane.

Note that for a given value of $\bar{\lambda}$, from the result in Theorem 1(ii), Proposition 2, and Lemma 5, we infer the following corollary :

Corollary 1 $\alpha > 0 \iff \|\mathbf{t}_n\| \rightarrow \infty$.

According to this corollary and Lemma 5(iv), the values of $\|\frac{\mathbf{t}_n}{n}\|$ and $\|\mathbf{t}_n\|$ could be used to monitor the gap between $Z(\bar{\lambda})$ and $W(\bar{\lambda})$. However, we will use other parameters to monitor this distance, as explained in the next section.

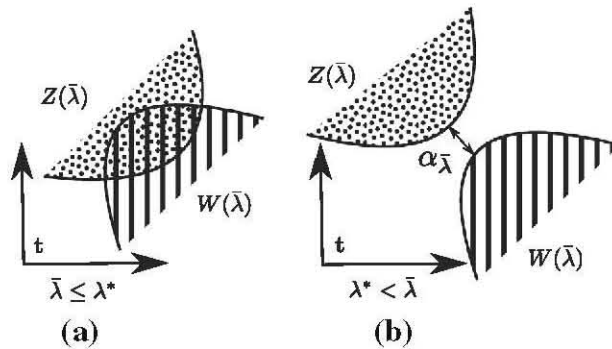
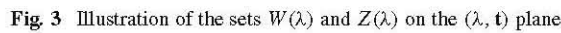


Fig. 2 Illustration of the sets $Z(\bar{\lambda})$ and $W(\bar{\lambda})$ for the case $\bar{\lambda} \leq \lambda^*$ and $\bar{\lambda} > \lambda^*$



As it has been explained in the previous sections, the objective is to solve an optimisation problem with the structure in (1), recasted in the form in (7).

The procedure of the master and sub-problems are detailed in next paragraphs, which use the following notation:

- Also $\lambda_{lb}^k, \lambda_{ub}^k$ denote algorithmic lower and upper bounds of λ^* in the master iteration k , respectively. Finally we define $\Delta\lambda^k = \lambda_{ub}^k - \lambda_{lb}^k$.

The steps that define the master problem are given in Box 1.

M4. If $|\lambda_{up}^k - \lambda_{lb}^k| \leq \epsilon_\lambda$

- $\lambda^* \approx \lambda^k$,
- Stop.

Else

- $k = k + 1$,
- Go to M1,

End.

Box 1. Master problem

In step M1, λ_{lb}^0 is a feasible solution for problem (7), which means that it exists a vector \mathbf{t} such that $\mathbf{t} \in Z(\lambda_{lb}^0) \cap W(\lambda_{lb}^0)$, with $W(\lambda_{lb}^0)$ and $Z(\lambda_{lb}^0)$ defined in (9). λ_{ub}^0 is an arbitrary upper bound for λ^* that can be obtained via any possible way. In this article, we use Proposition 3 to obtain an upper bound of λ^* : we solve two sub-problems defined in (10), and we obtain two upper bounds λ_{1ub}^0 and λ_{2ub}^0 . Then we set $\lambda_{ub}^0 = \min(\lambda_{1ub}^0, \lambda_{2ub}^0)$.

In step M1, if $X_1(\lambda_{ub}^{k-1}) \neq \emptyset$ and $X_2(\lambda_{ub}^{k-1}) \neq \emptyset$ defined in (8), we clearly have that $X_1(\lambda^k) \neq \emptyset$ and $X_2(\lambda^k) \neq \emptyset$, since f_1, f_2, g_1, g_2 are affine functions and K_1, K_2 are convex sets.

3.3.2 Sub-problems

The iterative process of each sub-problem is summarised in Box 2.

S1. Set $\bar{\lambda} = \lambda^k, \mathbf{t}_0^k = \mathbf{t}^{k-1}, n = 0$.

S2. Solve Sub-problem 1 defined in (15). Obtain \mathbf{d}_n^1 and set $\mathbf{t}_n^k = \mathbf{t}_{n-1}^k + 2\mathbf{d}_n^1$.

S3. Solve Sub-problem 2 defined in (17). Obtain \mathbf{d}_n^2 and set $\mathbf{t}_n^k = \mathbf{t}_n^k + 2\mathbf{d}_n^2$.

S4. Set $\beta_n = \|\mathbf{d}_n^1\| + \|\mathbf{d}_n^2\|$ and $\alpha_n = \|\mathbf{d}_n^1 + \mathbf{d}_n^2\|$.

S5. If $\beta_n < \beta_{n-1}$ or $\alpha_n \leq \epsilon_0$

- $\lambda_{lb}^k = \lambda^k, \lambda_{ub}^k = \lambda_{ub}^{k-1}$,
- Go to Box 1.M1.

Else if $\beta_n > \beta_{n-1}$ and $\alpha_n > \frac{1}{k}|\Delta\lambda^k|$ and $\frac{|\Delta\alpha_n|}{|\alpha_n|} < \epsilon_1$

- $\lambda_{lb}^k = \lambda_{lb}^{k-1}, \lambda_{ub}^k = \lambda^k$,
- Go to Box 1.M1.

Else

- $\mathbf{t}_n^k = \frac{\mathbf{t}_{n-1}^k + \mathbf{t}_n^k}{2}, n = n + 1$,
- Go to S2.

End

Box 2. Sub-problem

The algorithm in Box 2 uses the control parameter β_n to detect whether λ^k is an upper bound or a lower bound of λ^* . Indeed, the numerical results show that β_n

decreases when λ^k is a lower bound of λ^* . The values of β_n satisfy in fact the following corollary:

Corollary 2 Assume that $\bar{\lambda}$ is a given real value and $W = W(\bar{\lambda})$, $Z = Z(\bar{\lambda})$ defined in (9). Then the following relations hold.

- (i) If $\mathbf{0} \in \text{int}(W \cap Z)$ then $\lim_{n \rightarrow \infty} \beta_n = 0$.
- (ii) If $\lim_{n \rightarrow \infty} \beta_n = 0$ then $W \cap Z \neq \emptyset$.

Proof (i) Since $\mathbf{0} \in \text{int}(W \cap Z)$, then $W \cap Z \neq \emptyset$, and it follows that $\text{Fix } P_W \neq \emptyset$, $\text{Fix } P_Z \neq \emptyset$, and then we have that $W \cap Z = \text{Fix } P_W \cap \text{Fix } P_Z = \text{Fix } P_W P_Z$ (see [5], p. 71).

According to the AAR method, since $W \cap Z \neq \emptyset$, we have that

$$\lim_{n \rightarrow \infty} \mathbf{t}_n = \bar{\mathbf{t}} \in \text{Fix } T.$$

On the other hand, since $\mathbf{0} \in \text{int}(W \cap Z)$, then $\text{Fix } T = W \cap Z$ [6], and therefore we have that in turn,

$$P_Z(\bar{\mathbf{t}}) = \bar{\mathbf{t}}, P_W(\bar{\mathbf{t}}) = \bar{\mathbf{t}}, R_Z(\bar{\mathbf{t}}) = 2P_Z(\bar{\mathbf{t}}) - \bar{\mathbf{t}} = \bar{\mathbf{t}}. \quad (25)$$

In view of (16), (19) and (25), the following results can be derived:

$$\begin{aligned} \lim_{n \rightarrow \infty} \mathbf{d}_n^1 &= \lim_{n \rightarrow \infty} P_Z(\mathbf{t}_n) - \mathbf{t}_n = P_Z(\bar{\mathbf{t}}) - \bar{\mathbf{t}} = \mathbf{0} \\ \lim_{n \rightarrow \infty} \mathbf{d}_n^2 &= \lim_{n \rightarrow \infty} \mathbf{t}_{n+1} - \mathbf{t}_n - \mathbf{d}_n^1 = \mathbf{0}. \end{aligned} \quad (26)$$

Consequently, in view of (26), we infer that

$$\lim_{n \rightarrow \infty} \beta_n = \lim_{n \rightarrow \infty} \|\mathbf{d}_n^1\| + \|\mathbf{d}_n^2\| = 0. \quad (27)$$

- (ii) For each iteration n we have that $\alpha_n = \|\mathbf{d}_n^1 + \mathbf{d}_n^2\| \leq \|\mathbf{d}_n^1\| + \|\mathbf{d}_n^2\| = \beta_n$. Therefore,

$$0 = \lim_{n \rightarrow \infty} \beta_n \geq \lim_{n \rightarrow \infty} \alpha_n = \alpha \geq 0, \quad (28)$$

which implies that $\alpha = 0$. Consequently, in view of Proposition 2 and Theorem 1, we infer that $W \cap Z \neq \emptyset$. \square

We note that the update of λ^k in step M1 of the master problem mimics the update process of the bisection method. Other faster updates could be envisaged, but at the expense of estimating more accurately the distance between the sets $W(\lambda^k)$ and $Z(\lambda^k)$. In our implementation of the sub-problems, we just detect from the trends of β_n and α_n whether the set $W(\lambda^k) \cap Z(\lambda^k)$ is empty or not, but do not actually compute the distance $d(W(\lambda^k), Z(\lambda^k))$. The accurate computation of the distance would require far more sub-iterations, and in the authors experience, this extra cost does not compensate the gain when more sophisticated updates for λ^k are implemented.

The algorithms in Box 1 and 2 use three tolerance parameters: ϵ_λ , ϵ_0 and ϵ_1 . Their meaning is the following:

- ϵ_λ : this is the desired tolerance for the objective λ , and it is such that $\lambda^* \in [\lambda_{lb}, \lambda_{ub}]$, with $|\lambda_{ub} - \lambda_{lb}| < \epsilon_\lambda$.
- ϵ_0 : is a tolerance for $d(W(\lambda^k), Z(\lambda^k))$. If $d(W(\lambda^k), Z(\lambda^k)) < \epsilon_0$, we will consider that $W(\lambda^k) \cap Z(\lambda^k) \neq \emptyset$.
- ϵ_1 is used to detect when the sequence α_n has converged

4 Numerical results

In all our numerical tests, we have used the values $(\epsilon_\lambda, \epsilon_0, \epsilon_1) = (1E - 3, 5E - 4, 3E - 2)$. The vectors $(\mathbf{t}^k, \mathbf{x}_1^k, \mathbf{x}_2^k)$ resulting from the sub-problems for the highest (latest) value of λ_{lb}^k furnish the algorithmic approximations of $(\mathbf{t}^*, \mathbf{x}_1^*, \mathbf{x}_2^*)$.

4.1 Illustrative example

We illustrate the AAR-based decomposition technique explained in previous section with a toy non-linear convex problem given by,

$$\begin{aligned} \lambda^* = \max_{\lambda, \mathbf{x}_1, \mathbf{x}_2} \lambda \\ \mathbf{A}_1 \mathbf{x}_1 + \lambda \mathbf{F}_1 = \mathbf{b}_1 \\ \mathbf{A}_2 \mathbf{x}_2 + \lambda \mathbf{F}_2 = \mathbf{b}_2 \\ \mathbf{G}_1 \mathbf{x}_1 + \mathbf{G}_2 \mathbf{x}_2 = \mathbf{b} \\ \mathbf{x}_1 \in K_1, \mathbf{x}_2 \in K_2, \lambda \in \mathbb{R}, \end{aligned} \quad (29)$$

where $\mathbf{x}_1 \in \mathbb{R}^4$, $\mathbf{x}_2 \in \mathbb{R}^4$, and K_1 and K_2 are second-order cones, which are defined next.

Definition 5 The second-order (convex) cone of dimension $n > 1$ is defined as

$$K = \{(t, \mathbf{x}) : (t, \mathbf{x}) \in \mathbb{R} \times \mathbb{R}^{n-1}, ||\mathbf{x}|| \leq t\} \quad (30)$$

which is also called the quadratic, ice-cream or Lorentz cone. For $n = 1$ we define the unit second-order cone as

$$K = \{t : t \in \mathbb{R}, 0 \leq t\}.$$

The values of matrix \mathbf{A}_i , \mathbf{G}_i and vectors \mathbf{F}_i , \mathbf{b}_i and \mathbf{b} are given in the Appendix. The optimal value of this problem is $\lambda^* \approx 1.1965$, which has been computed by solving the global problem in (29) with package MOSEK.

By selecting arbitrary vectors \mathbf{b}_3 and \mathbf{b}_4 such that $\mathbf{b} = \mathbf{b}_3 + \mathbf{b}_4$ and introducing a new variable \mathbf{t} , the problem (29) is written in the following form,

$$\begin{aligned}
 \max_{\lambda, \mathbf{x}_1, \mathbf{x}_2, \lambda_t} \quad & \lambda \\
 \text{s.t.} \quad & \mathbf{A}_1 \mathbf{x}_1 + \lambda \mathbf{F}_1 = \mathbf{b}_1 \\
 & \mathbf{G}_1 \mathbf{x}_1 - \mathbf{b}_3 = \mathbf{t} \\
 & \mathbf{A}_2 \mathbf{x}_2 + \lambda \mathbf{F}_2 = \mathbf{b}_2 \\
 & \mathbf{G}_2 \mathbf{x}_2 - \mathbf{b}_4 = -\mathbf{t} \\
 & \mathbf{x}_1 \in K_1, \mathbf{x}_2 \in K_2, \lambda \in \mathbb{R}, \mathbf{t} \in \mathbb{R}^m.
 \end{aligned} \tag{31}$$

In view of (31), for any given $\lambda = \bar{\lambda}$, we have the two feasible sets defined in (9), which now take the following form:

$$\begin{aligned}
 Z(\bar{\lambda}) &= \{\mathbf{t} \mid \mathbf{G}_1 \mathbf{x}_1 - \mathbf{b}_3 = \mathbf{t}, \mathbf{A}_1 \mathbf{x}_1 + \bar{\lambda} \mathbf{F}_1 = \mathbf{b}_1 \text{ for some } \mathbf{x}_1 \in K_1\}, \\
 W(\bar{\lambda}) &= \{\mathbf{t} \mid \mathbf{G}_2 \mathbf{x}_2 - \mathbf{b}_4 = -\mathbf{t}, \mathbf{A}_2 \mathbf{x}_2 + \bar{\lambda} \mathbf{F}_2 = \mathbf{b}_2 \text{ for some } \mathbf{x}_2 \in K_2\}.
 \end{aligned}$$

For solving this problem we first take $(\lambda_{lb}^0, \mathbf{t}) = (0, \mathbf{0})$ and, according to Proposition 3, after solving the two optimisation problem in (10), we obtain two upper bounds $(\lambda_{1ub}, \lambda_{2ub}) = (1.2000, 3.2000)$. Then we set $\lambda_{ub}^0 = \min\{\lambda_{1ub}, \lambda_{2ub}\} = 1.2000$, and consequently $\lambda^* \in [\lambda_{lb}^0, \lambda_{ub}^0] = [0, 1.2000]$. The algorithm comes to an end when $|\Delta\lambda^k| = |\lambda_{up}^k - \lambda_{lb}^k| < \epsilon_\lambda = 1E - 3$.

The numerical results of the algorithm are reported in Table 1, where k indicates the number of master iterations, and n_k is the number of iterations taken by the sub-problems at each master iteration k . The second and third column indicate the highest lower bound and lowest upper bound at each master iteration, in such a way that $\lambda^* \in [\lambda_{lb}^{k-1}, \lambda_{ub}^{k-1}]$, and $\lambda^k = (\lambda_{lb}^{k-1} + \lambda_{ub}^{k-1})/2$ is an estimate of λ^* . Numbers in bold font indicate that λ^k is an upper bound, and $\Delta\lambda^{k-1} = \lambda_{ub}^{k-1} - \lambda_{lb}^{k-1}$. It can be observed that after a total of 24 sub-iterations (sum of all iterations n_k), the gap between the λ^* and the approximate value of λ^* is $3E - 4$. Figure 4 shows the optimal solution of the toy problem and the evolution of λ^k , which converges towards the optimal value $\lambda^* = 1.1965$.

We note that the choice of \mathbf{b}_3 and \mathbf{b}_4 is arbitrary, but that the same optimal optimum is obtained for different choices, even of the feasibility sets $W(\lambda^k)$ and $Z(\lambda^k)$ depend on these choices.

4.2 Example 2

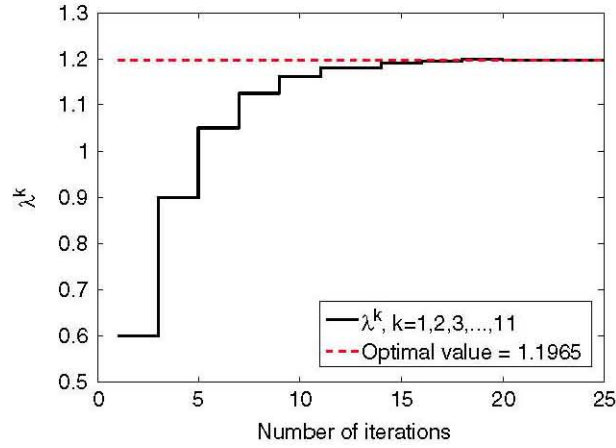
We consider an optimisation problem with the same structure in (29):

$$\begin{aligned}
 \lambda^* &= \max_{\lambda, \mathbf{x}_1, \mathbf{x}_2} \lambda \\
 \text{s.t.} \quad & \mathbf{A}_1 \mathbf{x}_1 + \lambda \mathbf{F}_1 = \mathbf{b}_1 \\
 & \mathbf{A}_2 \mathbf{x}_2 + \lambda \mathbf{F}_2 = \mathbf{b}_2 \\
 & \mathbf{G}_1 \mathbf{x}_1 + \mathbf{G}_2 \mathbf{x}_2 = \mathbf{0} \\
 & \mathbf{x}_1 \in K_1, \mathbf{x}_2 \in K_2, \lambda \geq 0.
 \end{aligned} \tag{32}$$

Table 1 Results of toy problem by using AAR-based decomposition method

Toy problem						
$\lambda^* = 1.1965$						
k	λ_{lb}^{k-1}	λ_{ub}^{k-1}	λ^k	$\frac{ \lambda^* - \lambda^k }{ \lambda^* }$	n_k	$\Delta\lambda^{k-1}$
1	0	1.2000	0.6000	0.4985	2	1.2000
2	0.6000	1.2000	0.9000	0.2478	2	0.6000
3	0.9000	1.2000	1.0500	0.1224	2	0.3000
4	1.0500	1.2000	1.1250	0.0597	2	0.1500
5	1.1250	1.2000	1.1625	0.0284	2	0.0750
6	1.1625	1.2000	1.1812	0.0127	3	0.0375
7	1.1812	1.2000	1.1906	0.0049	2	0.0187
8	1.1906	1.2000	1.1953	0.0010	2	0.0094
9	1.1953	1.2000	1.1977	0.0010	2	0.0047
10	1.1953	1.1977	1.1965	0.0000	2	0.0023
11	1.1965	1.1977	1.1971	0.0005	3	0.0012
12	1.1965	1.1971	$\lambda^* \simeq 1.1968$	0.0003	$\sum_{k=1}^{12} n_k = 24$	0.0007

Numbers in bold font indicate upper bounds of λ^*

**Fig. 4** Evolution of λ^k for the toy problem

but with increasing sizes of the constraint matrices and the number of variables, as indicated in Table 2. In this case, K_1 and K_2 are given by

$$K_1 = \mathbb{R}^{n_1} \times k_{11} \times k_{12} \times \cdots \times k_{1p}, \quad p \geq 1,$$

$$K_2 = \mathbb{R}^{n_2} \times k_{21} \times k_{22} \times \cdots \times k_{2q}, \quad q \geq 1,$$

where k_{ij} are three-dimensional second-order cones, and matrices \mathbf{A}_i , and vectors \mathbf{F}_i , \mathbf{b}_i are those resulting from a discretised limit analysis problem with finite elements,

Table 2 Size, total CPU time and total number of sub-iterations of each problem solved using MOSEK [1] and SDPT3 [31]

Problem	n	m	CPU time (s)		$\sum_{k=1} n_k$	
			SDPT3	MOSEK	SDPT3	MOSEK
1	2240	2401	163	30	45	45
2	4368	4705	281	84	43	42
3	8880	9601	913	243	40	42
4	12768	13825	1754	543	47	49
5	14979	16225	*	707	*	48

* In Problem 5, SDPT3 failed to give accurate results at iteration $k = 2$

similar to those in [21,24]. The variables \mathbf{x}_i correspond to nodal stresses and λ is the load factor which multiplies the applied external loads and which is maximised. The constraint equations correspond to equilibrium conditions (i) inside each finite element, (ii) between adjacent elements, and (iii) with the applied loads. The cones represent admissibility values of the stresses, as it is required for plastic materials. The reader is referred for instance to [21,24] for further details on how the optimisation problem is obtained.

The problem (32) can be written in the general standard form:

$$\begin{aligned}
 \lambda^* &= \max_{\mathbf{x}} \mathbf{c}^T \mathbf{x} \\
 \mathbf{A}\mathbf{x} &= \mathbf{b} \\
 \mathbf{x} &\in K,
 \end{aligned} \tag{33}$$

where K is a convex cone and $\mathbf{x} = (\mathbf{x}_1, \mathbf{x}_2, \lambda) \in \mathbb{R}^{n_1} \times \mathbb{R}^{n_2} \times \mathbb{R}$, with $1200 \leq n_1, n_2 \leq 8112$. The optimal solution λ^* for each problem has been computed solving the global problem in (32) with MOSEK.

Now we apply the AAR-based decomposition method for the following five nonlinear convex problems with the same structures defined in (32). The size of the problems are given in Table 2, where n, m denote the number of rows and columns of matrix \mathbf{A} in (33), and $\sum_{k=1} n_k$ is the total number of sub-iterations.

Each one of the sub-problems has been solved using the optimisation software MOSEK [1] and SPT3 [31]. In all the cases, the the total CPU time employed by using MOSEK was lower and yielded more accurate results (smaller gap between primal and dual problems). For the larger Problem 5, SDPT3 failed to give accurate results after the master iteration $k = 2$. The exact value has been computed by solving the global problem (32) with the MOSEK package.

Like in the toy problem, we have used Proposition 3 in order to obtain an upper bound of λ^* at the master iteration zero λ_{ub}^0 . The numerical results of problems 1–5 are reported in Tables 3, 4, 5, 6 and 7, respectively. It can be observed that when λ^k is a lower bound, the number of sub-iterations is very low, while in those cases that λ^k is an upper bound (indicated in bold font), the number of iterations increases

Table 3 Numerical results of Problem 1

Problem 1					$\Delta\lambda^{k-1}$		
$\lambda^* = 0.5008$					n_k		
k	λ_{lb}^{k-1}	λ_{ub}^{k-1}	λ^k	$\frac{ \lambda^* - \lambda^k }{ \lambda^* }$	SDPT3	MOSEK	
1	0	0.7071	0.3536	0.2940	2	2	0.7071
2	0.3536	0.7071	0.5303	0.0589	13	13	0.3536
3	0.3536	0.5303	0.4419	0.1176	2	2	0.1768
4	0.4419	0.5303	0.4861	0.0293	2	2	0.0884
5	0.4861	0.5303	0.5082	0.0148	5	4	0.0442
6	0.4861	0.5082	0.4972	0.0073	2	2	0.0221
7	0.4972	0.5082	0.5027	0.0038	10	10	0.0110
8	0.4972	0.5027	0.4999	0.0017	2	2	0.0055
9	0.4999	0.5027	0.5013	0.0010	5	6	0.0028
10	0.4999	0.5013	0.5006	0.0004	2	2	0.0014
	0.5006	0.5013	$\lambda^* \simeq 0.5010$	0.0003	45	45	0.0007

Table 4 Numerical results of Problem 2

Problem 2					n_k		$\Delta\lambda^{k-1}$
k	λ_{lb}^{k-1}	λ_{ub}^{k-1}	λ^k	$\frac{ \lambda^*-\lambda^k }{ \lambda^* }$	SDPT3	MOSEK	
1	0	0.7071	0.3536	0.2990	2	2	0.7071
2	0.3536	0.7071	0.5303	0.0514	14	14	0.3536
3	0.3536	0.5303	0.4419	0.1238	2	2	0.1768
4	0.4419	0.5303	0.4861	0.0362	2	2	0.0884
5	0.4861	0.5303	0.5082	0.0076	6	6	0.0442
6	0.4861	0.5082	0.4972	0.0143	2	2	0.0221
7	0.4972	0.5082	0.5027	0.0033	2	2	0.0110
8	0.5027	0.5082	0.5055	0.0021	5	5	0.0055
9	0.5027	0.5055	0.5041	0.0006	2	2	0.0028
10	0.5041	0.5055	0.5048	0.0008	6	5	0.0014
	0.5041	0.5048	$\lambda^* \simeq 0.5044$	0.0001	43	42	0.0007

notoriously. Figure 5 shows λ^* , λ^k and β_n related to Problem 3 as a function of number of sub-iterations. For the other problems, similar trends of these variables have been obtained. It can be observed that when λ^k is a lower bound (below the dotted line in Fig. 5a), β_n decreases steadily, and when λ^k is an upper bound (above the dotted line in Fig. 5b), β_n increases. The increase of β does not necessarily make λ^k an upper bound. However, when β increases and α converges to a positive value, then λ is detected as an upper bound.

Table 5 Numerical results of Problem 3

Problem 3					$\Delta\lambda^{k-1}$		
$\lambda^* = 0.5063$					n_k		
k	λ_{lb}^{k-1}	λ_{ub}^{k-1}	λ^k	$\frac{ \lambda^*-\lambda^k }{ \lambda^* }$	SDPT3	MOSEK	
1	0	0.7071	0.3536	0.3017	2	2	0.7071
2	0.3536	0.7071	0.5303	0.0475	14	15	0.3536
3	0.3536	0.5303	0.4419	0.1271	2	2	0.1768
4	0.4419	0.5303	0.4861	0.0398	2	2	0.0884
5	0.4861	0.5303	0.5082	0.0039	8	7	0.0442
6	0.4861	0.5082	0.4972	0.0180	2	2	0.0221
7	0.4972	0.5082	0.5027	0.0071	2	2	0.0110
8	0.5027	0.5082	0.5055	0.0016	2	2	0.0055
9	0.5055	0.5082	0.5069	0.0011	4	6	0.0028
10	0.5055	0.50569	0.5062	0.0002	2	2	0.0014
	0.5062	0.5069	$\lambda^* \simeq 0.5065$	0.0004	40	42	0.0007

Table 6 Numerical results of Problem 4

Problem 4					$\Delta\lambda^{k-1}$		
$\lambda^* = 0.5071$					n_k		
k	λ_{lb}^{k-1}	λ_{ub}^{k-1}	λ^k	$\frac{ \lambda^* - \lambda^k }{ \lambda^* }$	SDPT3	MOSEK	
1	0	0.7071	0.3536	0.3028	2	2	0.7071
2	0.3536	0.7071	0.5303	0.0458	14	15	0.3536
3	0.3536	0.5303	0.4419	0.1285	2	2	0.1768
4	0.4419	0.5303	0.4861	0.0414	2	2	0.0884
5	0.4861	0.5303	0.5082	0.0022	15	16	0.0442
6	0.4861	0.5082	0.4972	0.0196	2	2	0.0221
7	0.4972	0.5082	0.5027	0.0087	2	2	0.0110
8	0.5027	0.5082	0.5055	0.0032	2	2	0.0055
9	0.5055	0.5082	0.5069	0.0005	2	2	0.0028
10	0.5069	0.5082	0.5075	0.0008	4	4	0.0014
	0.5069	0.5075	$\lambda^* \simeq 0.5072$	0.0002	47	49	0.0007

Figure 6 shows the relative error of the resulting values λ^k for each master iterations k . The Figure shows that λ^k converges linearly towards the exact solution λ^* . The linear convergence trend is characteristic of the bisection method employed for the update of λ^k in step M1, Box 1. We have also tested to the update of master problem using the secant method. This consists on computing the new estimate λ^k from the two lowest upper bounds λ_{ub}^1 and λ_{ub}^2 and the corresponding distances α_{ub}^1 and α_{ub}^2 as

Table 7 Numerical results of Problem 5

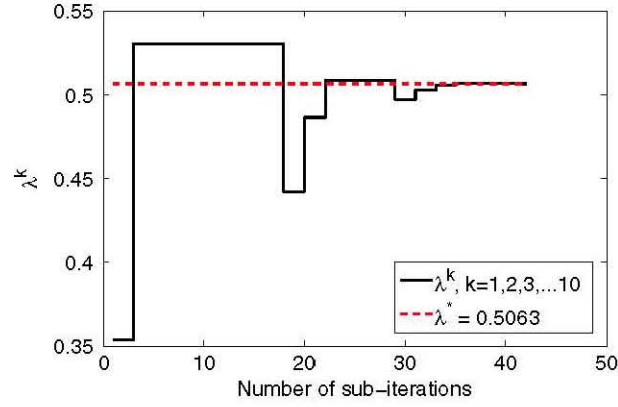
Problem 5					$\Delta\lambda^{k-1}$	
$\lambda^* = 0.5074$					n_k	
k	λ_{lb}^{k-1}	λ_{ub}^{k-1}	λ^k	$\frac{ \lambda^* - \lambda^k }{ \lambda^* }$	MOSEK	
1	0	0.7071	0.3536	0.3032	2	0.7071
2	0.3536	0.7071	0.5303	0.0452	14	0.3536
3	0.3536	0.5303	0.4419	0.1290	2	0.1768
4	0.4419	0.5303	0.4861	0.0419	2	0.0884
5	0.4861	0.5303	0.5082	0.0017	16	0.0442
6	0.4861	0.5082	0.4972	0.0201	2	0.0221
7	0.4972	0.5082	0.5027	0.0092	2	0.0110
8	0.5027	0.5082	0.5055	0.0038	2	0.0055
9	0.5055	0.5082	0.5069	0.0011	2	0.0028
10	0.5069	0.5082	0.5075	0.0003	4	0.0014
	0.5069	0.5075	$\lambda^* \simeq 0.5072$	0.0004	48	0.0007

$$\lambda^k = \lambda_{ub}^2 - \frac{\lambda_{ub}^1 - \lambda_{ub}^2}{\alpha_{ub}^1 - \alpha_{ub}^2} \alpha_{ub}^2 \quad (34)$$

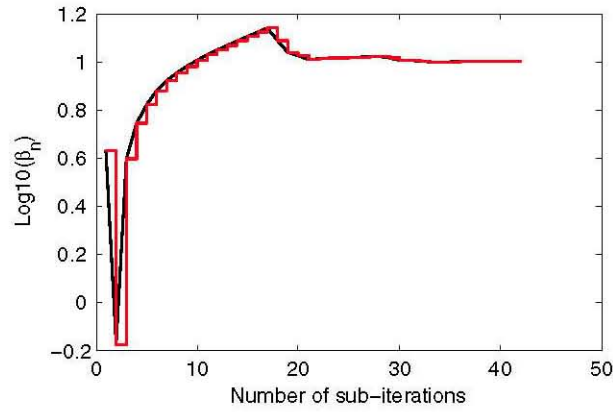
that is, to use a linear interpolation of the relation between λ and α and set $\alpha = 0$. This update may become more efficient if the values of α_{ub}^1 and α_{ub}^2 are sufficiently accurate. Otherwise, it may be found that the new value λ^k computed in (34) may fall outside of the smallest bracketing interval $[\lambda_{lb}, \lambda_{ub}]$, in which case we have used the bisection method. We have implemented the secant update in (34) and applied it to Problem 3. The results are shown in Table 8. It can be observed that for the same tolerances, the number of master iterations has reduced from 10 to 8, and that n_k is also slightly lower (from 40 and 42 to 35 when using both, SDPT3 and MOSEK). This reduction cannot be automatically extrapolated for all cases, but shows that other more sophisticated updates may further reduce the number of iterations, assuming that the values of α_{ub}^1 and α_{ub}^2 are accurate enough, that is, if enough iterations are employed in the sub-problems when λ^k is an upper bound.

We note that the use of Benders decomposition, and for the same convergence tolerance, these non-linear problems required more than 200 iterations in all cases, and for example, Problem 3 needed more than 800 iterations. Furthermore, and in contrast to our method, the number of iterations of the Benders implementation scaled with the problem size.

The CPU time for solving an optimisation problem grows quadratically with respect to the number of variables. Therefore, given the total number of iterations shown in the previous tables, the direct solution of the global problem in (1) requires less CPU time than the sequential solution of the proposed strategy using the master problem and the sub-problems (which contain approximately one half of the total



(a) Problem 3



(b) Problem 3

Fig. 5 **a** Evolution of λ^k for Problem 3, and **b** β_n as a function of the total number of sub-iterations

number of variables). Current research on further reducing the number of iterations in the sub-problems is being undertaken. Nonetheless, we point out that the memory requirements are halved when using our decomposed approach, and that a parallel implementation of the master iterations for different values of λ^k may reduce substantially the total number of iterations n_k .

5 Conclusions

In the paper we have proposed a method to decompose convex non-linear problems that contain only one complicating variable in the objective function. This type of problems includes many engineering applications in plastic structural analysis.

The method consists on interpreting the optimisation problem as the maximisation (or minimisation) of the variable subjected to a non-empty intersection set. The numerical results show that the total number of iterations does not scale with the number of

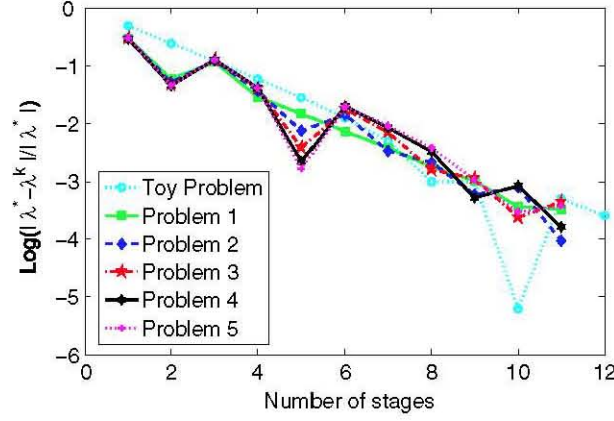


Fig. 6 Evolution of the relative error for each master iteration

Table 8 Numerical results of Problem 3 using the secant method instead of the bisection method in the master problem

Problem 3 (Secant method)						$\Delta\lambda^{k-1}$	
$\lambda^* = 0.5063$							
k	λ_{lb}^{k-1}	λ_{ub}^{k-1}	λ^k	$\frac{ \lambda^*-\lambda^k }{ \lambda^* }$	n_k	SDPT3	MOSEK
1	0	0.7071	0.3536	0.3017	2	2	0.7071
2	0.3536	0.7071	0.5303	0.0475	14	15	0.3536
3	0.3536	0.5303	0.4419	0.1271	2	2	0.1768
4	0.4419	0.5303	0.4861	0.0398	2	2	0.0884
5	0.4861	0.5303	0.5082	0.0038	8	7	0.0442
6	0.4861	0.5082	0.5055	0.0015	2	2	0.0221
7	0.5055	0.5082	0.5069	0.0011	3	3	0.0027
8	0.5055	0.5069	0.5062	0.0002	2	2	0.0014
	0.5062	0.5069	$\lambda^* \simeq 0.5065$	0.0005	35	35	0.0007

variables. The extension of the method for a larger number of sub-problems requires the application of the AAR method for a larger number of sets, as explained in [6], p. 189. This approach is currently being investigated.

Acknowledgments The authors acknowledge the financial support of the Spanish Ministry of Economy and Competitiveness, under the Grant Nr. DPI2013-43727-R.

Matrices and vectors in the toy problem

The explicit expressions of the matrices and vectors employed in the toy problem in (29) are the following:

$$\begin{aligned}
& \max_{\lambda, \mathbf{x}_1, \mathbf{x}_2} \lambda \\
& \overbrace{\begin{bmatrix} 1 & -1 & 0 & 0 \\ 1 & 0 & 0 & 1 \end{bmatrix}}^{\mathbf{A}_1} \mathbf{x}_1 + \lambda \overbrace{\begin{bmatrix} 1 \\ 1 \end{bmatrix}}^{\mathbf{F}_1} = \overbrace{\begin{bmatrix} 1.2 \\ 2.4 \end{bmatrix}}^{\mathbf{b}_1} \\
& \overbrace{\begin{bmatrix} -1 & 1 & 0 & 0 \\ 1 & 0 & 0 & 1 \end{bmatrix}}^{\mathbf{A}_2} \mathbf{x}_2 + \lambda \overbrace{\begin{bmatrix} 1 \\ 1 \end{bmatrix}}^{\mathbf{F}_2} = \overbrace{\begin{bmatrix} 3.2 \\ 0.2 \end{bmatrix}}^{\mathbf{b}_2} \\
& \overbrace{\begin{bmatrix} 0 & 2 & 1 & 0 \\ 0 & 2 & 0 & 1 \end{bmatrix}}^{\mathbf{G}_1} \mathbf{x}_1 + \overbrace{\begin{bmatrix} 0 & 2 & 1 & 1 \\ 0 & 2 & 0 & 1 \end{bmatrix}}^{\mathbf{G}_2} \mathbf{x}_2 = \overbrace{\begin{bmatrix} 1.2 \\ 6.2 \end{bmatrix}}^{\mathbf{b}} = \overbrace{\begin{bmatrix} -0.8 \\ 3.2 \end{bmatrix}}^{\mathbf{b}_3} + \overbrace{\begin{bmatrix} 2 \\ 3 \end{bmatrix}}^{\mathbf{b}_4} \\
& \mathbf{x}_1 \in K_1, \mathbf{x}_2 \in K_2, \lambda \in \mathbb{R}
\end{aligned}$$

References

1. Andersen, E.D., Roos, C., Terlaky, T.: On implementing a primal-dual interior-point method for conic quadratic optimization. *Math. Progr.* **95**(2), 249–277 (2003)
2. Baillon, J.B., Bruck, R.E., Reich, S.: On the asymptotic behavior of nonexpansive mappings and semigroups in Banach spaces. *Houst. J. Math.* **4**, 1–9 (1978)
3. Bauschke, H.H., Borwein, J.M.: On the convergence of von Neumann's Alternating Projection Algorithm for two sets. *Set-Valued Anal.* **1**, 185–212 (1993)
4. Bauschke, H.H., Borwein, J.M.: On projection algorithms for solving convex feasibility problems. *SIAM Rev.* **38**, 367–426 (1996)
5. Bauschke, H.H., Combettes, P.L.: *Convex Analysis and Monotone Operator Theory in Hilbert Spaces*. Springer, New York (2010)
6. Bauschke, H.H., Combettes, P.L., Luke, D.R.: Finding best approximation pairs relative to two closed convex sets in Hilbert spaces. *J. Approx. Theory* **127**, 178–192 (2004)
7. Bertsekas, D.P., Tsitsiklis, J.N.: *Parallel and Distributed Computation: Numerical Methods*. Prentice Hall, New York (1989)
8. Bilotta, A., Leonetti, L., Garcea, G.: An algorithm for incremental elastoplastic analysis using equality constrained sequential quadratic programming. *Comput. Struct.* **102–03**, 97–107 (2012)
9. Bruck, R.E., Reich, S.: Nonexpansive projections and resolvents of accretive operators in Banach spaces. *Houst. J. Math.* **3**, 459–470 (1977)
10. Chen, G., Teboulle, M.: A proximal-based decomposition method for convex minimization problems. *Math. Progr. A* **64**, 81–101 (1994)
11. Combettes, P.L.: Inconsistent signal feasibility problems: least-squares solutions in a product space. *IEEE Trans. Signal Process.* **42**, 2955–2966 (1994)
12. Combettes, P.L.: Hilbertian convex feasibility problem: convergence of projection methods. *Appl. Math. Optim.* **35**, 311–330 (1997)
13. Conejo, A.J., Castillo, E., Mínguez, R., García-Bertrand, R.: *Decomposition Techniques in Mathematical Programming*. Springer, Berlin (2006)
14. Deutsch, F.: *Best Approximation in Inner Product Spaces*. Springer, New York (2001)
15. Dinh, Q.T., Savorgnan, C., Diehl, M.: Combining Lagrangian decomposition and excessive gap smoothing technique for solving large-scale separable convex optimization problems. *Comput. Optim. Appl.* **55**, 75–111 (2013)
16. Gabriel, S.A., Shim, Y., Conejo, A.J., de la Torre, S., García-Bertrand, R.: A Benders decomposition method for discretely-constrained mathematical programs with equilibrium constraints. *J. Oper. Res. Soc.* **61**(9), 1404–1419 (2010)
17. Goldberg, M., Marks, R.J.: Signal synthesis in the presence of an inconsistent set of constraints. *IEEE Trans. Circuits Syst.* **32**, 647–663 (1985)

18. Huang, J., Xu, W., Thomson, P., Di, S.: A general rigid-plastic/rigid-viscoplastic FEM for metal-forming processes based on the potential reduction interior point method. *Int. J. Mach. Tool Manuf.* **43**, 379–389 (2003)
19. Kaneko, I.: A decomposition procedure for large-scale optimum plastic design problems. *Int. J. Numer. Methods Eng.* **19**, 873–889 (1983)
20. Krabbenhöft, K., Lyamin, A.V., Sloan, S.W.: Formulation and solution of some plasticity problems as conic programs. *Int. J. Solids Struct.* **44**, 1533–1549 (2007)
21. Lyamin, A.V., Sloan, S.W.: Lower bound limit analysis using non-linear programming. *Int. J. Numer. Methods Eng.* **55**, 576–611 (2002)
22. Makrodimopoulos, A., Martin, C.M.: Lower bound limit analysis of cohesive-frictional materials using second-order cone programming. *Int. J. Numer. Methods Eng.* **66**, 604–634 (2006)
23. Monteiro, R.D.C., Ortiz, C., Svaiter, B.F.: Implementation of a block-decomposition algorithm for solving large-scale conic semidefinite programming problems. *Comput. Optim. Appl.* **57**, 45–69 (2013)
24. Muñoz, J.J., Bonet, J., Huerta, A., Peraire, J.: Upper and lower bounds in limit analysis: adaptive meshing strategies and discontinuous loading. *Int. J. Numer. Methods Eng.* **77**, 471–501 (2009)
25. Necoara, I., Suyken, J.A.K.: Application of a smoothing technique to decomposition in convex optimization. *IEEE Trans. Autom. Control* **53**, 2674–2679 (2008)
26. Nesterov, Y.: Smooth minimization of non-smooth functions. *Math. Progr. A* **103**(1), 127–152 (2005)
27. Opial, Z.: Weak convergence of the sequence of successive approximations for nonexpansive mappings. *Bull. Am. Math. Soc.* **73**, 591–597 (1967)
28. Pastor, F., Loute, E., Pastor, J., Trillat, M.: Mixed method and convex optimization for limit analysis of homogeneous Gurson materials: a kinematical approach. *Eur. J. Mech. A* **28**, 25–35 (2009)
29. Pazy, A.: Asymptotic behavior of contractions in Hilbert space. *Isr. J. Math.* **9**, 235–240 (1971)
30. Pesquet, J.C., Combettes, P.L.: Wavelet synthesis by alternating projections. *IEEE Trans. Signal Process.* **44**, 728–732 (1996)
31. Tütüncü, R.H., Toh, K.C., Todd, M.J.: Solving semidefinite-quadratic-linear programs using SDPT3. *Math. Progr. B* **95**, 189–217 (2003). Available <http://www.math.nus.edu.sg/mattohkc/sdpt3.html>