Computational Optimization and Applications manuscript No.
(will be inserted by the editor)

Decomposition Algorithms for Some Deterministic and Two-Stage Stochastic Single-Leader Multi-Follower Games

Pedro Borges · Claudia Sagastizábal · Mikhail Solodov

Received: May 2020 / Accepted: date

Abstract We consider a certain class of hierarchical decision problems that can be viewed as single-leader multi-follower games, and be represented by a virtual market coordinator trying to set a price system for traded goods, according to some criterion that balances supply and demand. The objective function of the market coordinator involves the decisions of many agents, which are taken independently by solving convex optimization problems that depend on the price configuration and on realizations of future states of the economy. One traditional way of solving this problem is via a mixed complementarity formulation. However, this approach can become impractical when the numbers of agents and/or scenarios become large. This work concerns agent-wise and scenario-wise decomposition algorithms to solve the equilibrium problems in question, assuming that the solutions of the agents' problems are unique, which is natural in many applications (when solutions are not unique, the approximating problems are still well-defined, but the convergence properties of the algorithm are not established). The algorithm is based on a previous work of the authors, where a suitable regularization of solution mappings of fully parameterized convex problems is developed. Here, we show one specific strategy to manage the regularization parameter, extend some theoretical results to the current setting, and prove that the smooth approximations of the market coordinator's problem converge epigraphically to the original problem. Numerical experiments and some comparisons with the complementarity solver PATH are shown for the two-stage stochastic Walrasian equilibrium problem.

Keywords Stochastic Equilibrium · Decomposition Algorithms · Bilevel Optimization · Smoothing Methods · Interior Penalty Methods · Tikhonov Regularization.

1 Introduction and motivation

The so-called setting of *multiple optimization problems with equilibrium constraints* (MOPEC) serves as a broad framework to encompass various types of hierarchical problems that arise often in applications, most notably in energy optimization; see [PFW16], [Sag12], and references therein. Here, we consider a certain class of MOPECs that can be viewed as single-leader multi-follower games.

We are interested in problems where, for a given parameter $p \in \mathbb{R}^q$, agents a in a set $\mathcal{A}$ determine their decisions $x_{\mathcal{A}}(p) = (x_a(p) \in \mathbb{R}^{n_a}, a \in \mathcal{A})$ by solving independently convex optimization problems of the form

$$x_a(p) = \arg\min_x \{f_a(x,p) : B_a(p)x = b_a(p), \quad g_a(x,p) \leq 0\}. \tag{1}$$

The convex objective function f_a and the affine equality and convex inequality constraints are such that in (1) the minimizer $x_a(p)$ is unique. The goal is to find the optimal parameter p^*, a price signal that is observed

Pedro Borges
Instituto de Matemática Pura e Aplicada, Rio de Janeiro, RJ, Brazil
E-mail: pborges@impa.br

Claudia Sagastizábal
IMECC/UNICAMP, Campinas, São Paulo, Brazil
E-mail: sagastiz@unicamp.br

Mikhail Solodov
Instituto de Matemática Pura e Aplicada, Rio de Janeiro, RJ, Brazil
E-mail: solodov@impa.br

when coupling all the agents' decisions, by minimizing a criterion $F : \mathbb{R}^N \to \mathbb{R}$ where $N = \sum_a n_a$ over a set $\mathit{\Pi}$:

$$p^* \in \operatorname{Arg\,min}_p \{F(x_{\mathcal{A}}(p)) : p \in \mathit{\Pi}\}. \tag{2}$$

The notation Arg min in (2) refers to a set, while arg min in (1) is a singleton.

Finding the optimal price p^* and decisions $x_{\mathcal{A}}(p^*)$ that solve (1)-(2) gives a particular instance of a MOPEC. We do not address the MOPEC setting in all of its generality, in which the agents behave strategically, taking into account the other agents' decisions in their individual optimization problems, as in a generalized Nash game. But the solution approach presented below, based on a special smoothing of the solution mappings of the agents' problems, should still be applicable for a general MOPEC. To simplify the presentation, we focus on the specific framework (1)-(2), suitable for the applications we have in mind. One example is the stochastic Walrasian Equilibrium Problem (WEP); see [DJW17; JJBW02].

As it occurs often in hierarchical optimization, in the WEP the agents' decisions are only defined for parameters in the set $\mathit{\Pi}$. This is because for such problems the feasible set is given by certain *price system*, specific to the considered economy. The willingness of the agents to trade the goods available in the economy is cost-minimizing, based on the price of the goods and possibly subject to budget constraints. Agents' decisions are taken only for positive prices, and the solution set in (1) is empty if $p \notin \mathit{\Pi}$. In order to determine the equilibrium price, in problem (2) a virtual market coordinator minimizes the mismatch between supply and demand.

Even in the setting of (1), less general than MOPEC, the global problem (2) can be nonsmooth and nonconvex, and computing a local minimizer is a difficult task. Our proposal considers the agent's problem (1) as a convex smooth program parameterized by the price, and applies the smoothing and regularization procedure of [BSS20] to the possibly nonsmooth solution mappings of (1). The idea, simple to explain (but not so simple to analyze theoretically), boils down to replacing in (2) the agents' decisions $x_a(p)$ (which may fail to have classical derivatives at all points) with approximating functions that are smooth. Accordingly, for the given smoothing parameter $\varepsilon > 0$ and regularization parameter $\mu = \mu(\varepsilon) \geq 0$, we build functions $x_a^{\varepsilon}(p) = x_a^{\varepsilon,\mu(\varepsilon)}(p)$ that are well-defined at least on $\mathit{\Pi}$, are smooth on a certain related set $\tilde{\mathit{\Pi}}$, and converge in the following sense:

$$\lim_{\varepsilon \searrow 0, p' \in \mathit{\Pi}, p' \to p} x_a^{\varepsilon}(p') = x_a(p) \quad \text{for all } p \in \mathit{\Pi} \text{ and } a \in \mathcal{A}. \tag{3}$$

The regularization parameter $\mu = \mu(\varepsilon) \geq 0$ is, in general, a function of the smoothing parameter $\varepsilon > 0$, bounded around zero. There are three important cases: $\mu = 0$, $\mu > 0$ but fixed, and variable $\mu > 0$ tending to zero as $\varepsilon \searrow 0$ (e.g., $\mu = \sqrt{\varepsilon}$). The crucial property (3) connects our approach with some classical concepts from variational analysis. Epigraphical convergence [RW09, Chapter 7] implies approximation properties for solutions of problem (2). In particular, when (3) holds, the approximating functions define a smoothing, in the sense of [Che12].

An important feature of our approach is that the derivatives of the approximating functions can be computed numerically. This is important to efficiently solve the approximations of the global problem (2). For an appropriate sequence of parameters $(\varepsilon_k \searrow 0, \mu_k = \mu(\varepsilon_k) \geq 0)$ and smooth functions $x_{\mathcal{A}}^{\varepsilon_k}(p) = x_{\mathcal{A}}^{\varepsilon_k,\mu_k}(p)$, our method computes a sequence p_k of approximate local solutions for the smooth problems

$$\min_p \{F(x_{\mathcal{A}}^{\varepsilon_k}(p)) : p \in \mathit{\Pi} \cap \tilde{\mathit{\Pi}}\}. \tag{4}$$

Depending on how the approximating functions are built, the smoothness set can be larger or smaller than $\mathit{\Pi}$. Having access to the derivatives of $x_{\mathcal{A}}^{\varepsilon_k}(p)$, first-order information for the objective function in (4) is available and a stationary point can be computed, for example using Ipopt [WB05]. In our numerical experiments in Section 5.2 we identified two issues that impact the performance of the method. First, a low accuracy in the output of the smoothed problems results in low quality derivatives for the smooth mappings $x_{\mathcal{A}}^{\varepsilon_k}(p)$ and this sometimes hinders the solution process. Second, the parameter $\varepsilon_k > 0$ defining the smoothing needs to be carefully chosen, so that the objectives of (2) and (4) are sufficiently close across iterations.

When compared with the method in [BSS20] for nonconvex two-stage stochastic programming problems, the most important conceptual difference of the current proposal is the following. In [BSS20], there is a master problem akin to (2) and subproblems like (1). The smoothing [BSS20] inserts in the master problem the *optimal value function* of the smoothed subproblems. By contrast, in (4), we rather use the smoothed *solution mappings*. This makes the situation much different for the convergence analysis of (2), because approximation properties for solution mappings are weaker than those available for optimal value functions. There are also important differences for computational implementations. For example, a single value for

the smoothing parameter $\varepsilon > 0$ (sufficiently small), was often sufficient in the computational experience of [BSS20]. By contrast, for problem (2) a proper management of the sequence of the values of ε_k becomes crucial.

Our approach is particularly well suited for decomposition. When solving (4), an agent-wise decomposition is readily available because the approximating functions are defined independently across agents. Furthermore, when the agents' problems (1) are two-stage stochastic programs, our construction allows for a decomposition method across both agents and scenarios. Decomposition methods for variational inequalities with Dantzig-Wolfe and Benders-type structure were developed in [LSS13], [LSS12], [LSS16]; extending [FC05], [CF10], [GF10]. For generalized Nash games, we refer to [FPS11] and [KS12]; and for sparse affine variational inequalities, see [KHF17] and [KF19]. Other agent-wise decomposition methods with applications in communications engineering are given in [ASP14; Scu+13; Scu+11].

Regarding the direct solution of a problem like (1)-(2), there are at least four major classes of methods. These are the complementarity-based or the variational inequality algorithms using PATH [FM99], the augmented Lagrangian methods [And+08], [Sch12], [KS16], the more recent derivative-free approach in [DJW17], and the smoothing techniques [XY13], [XWY14], [XYZ14], [XYZ15], relying on a smoothing given by an integral in a multidimensional space. Although not clear how to implement them efficiently in practice, integral-based smoothings are gradient consistent in the sense of [Che12], [BHK13]. This property guarantees that limits of stationary points of the smoothed problems are stationary points of the original problem (with stationarity being understood in the generalized sense of nonsmooth analysis). We do not show that our method is gradient consistent, which is the reason we focus only on the epigraphical convergence of the approximating problems. However, the property of gradient consistency is not a necessary condition for the method to work well in practice, as is also clear from the theory.

The rest of the paper is organized as follows. In Section 2, we give some preliminary results and notation. In Section 3, we adapt and extend some properties of the smoothing in [BSS20] to the current equilibrium setting. In Section 3.3, we describe our agent-wise decomposition method. Section 4 discusses the same two-stage WEP found in [DJW17]. Numerical experiments for the decomposition across agents are reported in Section 5.2. The agent-wise and scenario-wise decomposition method is shown in 5.3. Concluding remarks and comments are given in Section 6.

2 Background material

Notation-wise, we mostly follow [RW09], with $\overline{\mathbb{R}}$ being the extended real numbers, $B(u, \delta)$ the ball around u of radius $\delta > 0$, and with all norms being Euclidean (the respective spaces are always clear from the context). The symbol $o(t)$ denotes any expression such that $t^{-1}o(t) \to 0$ whenever $t \searrow 0$. The symbol conv D stands for the convex hull of the set D.

2.1 Properties of set-valued mappings

The set-valued map $\mathcal{X} : \mathbb{R}^q \hookrightarrow \mathbb{R}^n$ is pointwise bounded if $\mathcal{X}(p)$ is a bounded set for all $p \in \mathbb{R}^q$. It is locally bounded around p if there exists an open set $V \subset \mathbb{R}^q$ containing p and such that $\cup_{p' \in V} \mathcal{X}(p')$ is a bounded set. The domain of the set-valued map $\mathcal{X}$ is dom $\mathcal{X} := \{p \in \mathbb{R}^q : \mathcal{X}(p) \neq \emptyset\}$. The outer closure of the set-valued map $\mathcal{X}$ at $p \in \mathbb{R}^q$ is

$$\limsup_{p' \to p} \mathcal{X}(p') := \left\{ x \in \mathbb{R}^n : \exists p_k \to p, x_k \in \mathcal{X}(p_k) \text{ s.t. } x_k \to x \right\}.$$

If the set-valued map $\mathcal{X}$ is a singleton over dom $\mathcal{X}$, we represent it by a function over its domain, using a lower case letter. For example, if $\mathcal{X}(p) = \{x(p)\}$ for all p, we more often use $x(p)$.

The effective domain, or the domain, of a function $v : \mathbb{R}^q \to \overline{\mathbb{R}}$ is dom $v := \{p \in \mathbb{R}^q : v(p) < +\infty\}$. A function v is said to be continuous relative to $V \subset \mathbb{R}^q$ if for all $p \in V$ we have $v(p') \to v(p)$ whenever $p' \to p$ such that $p' \in V$. For instance, we could say that v is continuous relative to its domain, or relative to the interior of its domain, or relative to the interior of the domain of some set-valued map.

For a function $v : \mathbb{R}^q \to \overline{\mathbb{R}}$ continuous at a point $p \in \mathbb{R}^q$, the regular subdifferential at p is defined by

$$\hat{\partial} v(p) := \left\{ \psi \in \mathbb{R}^q : v(p') \geq v(p) + \psi^\top (p' - p) + o(\|p' - p\|) \right\},$$

and the limiting subdifferential by

$$\partial v(p) := \limsup_{p' \to p} \hat{\partial} v(p') \,.$$

If v is locally Lipschitz at p, then the Clarke subdifferential is given by

$$\partial_{\mathrm{c}} v(p) := \operatorname{conv} \partial v(p).$$

The following proposition, specializing [RW09, Theorems 9.13 and 9.2] to our setting, characterizes local boundedness of subdifferentials.

Proposition 1 *Given an open set V, for any function $v : V \to \mathbb{R}$ the following properties are equivalent:*

(i) *v is locally Lipschitz at p;*
(ii) *$\hat{\partial} v$ is locally bounded at p;*
(iii) *$\partial_c v$ is locally bounded at p.*

As is well known, if $\overline{p}$ is an unconstrained local minimizer of v, then $0 \in \hat{\partial} v(\overline{p})$ and $0 \in \partial_{\mathrm{c}} v(\overline{p})$.

2.2 Properties of smoothing functions and epigraphical convergence

Let $v : V \to \mathbb{R}$ be continuous on the open set $V \subset \mathbb{R}^q$, and assume that for $\varepsilon > 0$ we are given smooth functions $v^\varepsilon : V \to \mathbb{R}$ such that their lower semicontinuous closure, defined as

$$\operatorname{lsc} v^\varepsilon(p) := \liminf_{\varepsilon \searrow 0, p' \to p} v^\varepsilon(p') \,,$$

satisfies the identity

$$\operatorname{lsc} v^\varepsilon(p) = v(p) \ \text{ for all } p \in V \,. \tag{5}$$

As is well known, see [BHK13], it holds that

$$\partial v(p) \subset \operatorname{conv} \left\{ \limsup_{\varepsilon \searrow 0, p' \to p} \nabla v^\varepsilon(p') \right\} \ \text{ for all } p \in V \,.$$

The smoothing functions v^ε are said to be gradient consistent with v, when $\varepsilon \searrow 0$, if

$$\partial_{\mathrm{c}} v(p) \supset \operatorname{conv} \left\{ \limsup_{\varepsilon \searrow 0, p' \to p} \nabla v^\varepsilon(p') \right\} \ \text{ for all } p \in V \,.$$

For more details about smoothing functions, see [BHK13], [BH16]. The next result, that holds independently of the property of gradient consistency, shows that "unexpected" things may happen only when the Lipschitz constants of the smoothing gradients diverge. The result gives an indication that the parameter $\varepsilon > 0$ should be managed carefully. Notice also that, at points of nonsmoothness of the function, the Lipschitz constants of the gradient of the smoothings indeed "explode".

Proposition 2 *Let v and v^ε be given as above. Fix $p \in V$. Assume that there exists a constant $L > 0$, called a uniform Lipschitz constant for the gradients of v^ε at p, and there exists $\delta > 0$, such that*

$$\|\nabla v^\varepsilon(p_1) - \nabla v^\varepsilon(p_2)\| \le L \|p_1 - p_2\| \quad \textit{for all} \quad p_1, p_2 \in B(p, \delta) \quad \textit{and} \quad \varepsilon \in (0, \delta).$$

Then, $\nabla v(p)$ exists at $p \in V$ and

$$\lim_{\varepsilon \searrow 0, p' \to p} \nabla v^\varepsilon(p') = \nabla v(p).$$

In particular, if v is not differentiable at p, there is no uniform Lipschitz constant for the gradients of v^ε at p, independently of how the smoothing sequence v^ε is constructed.

Proof Using the uniform Lipschitz constant, the Newton-Leibniz formula (e.g., [IS14, Lemma A.11]) implies, for $t \in (-\frac{\delta}{2}, \frac{\delta}{2})$ and p' close to p and $d \in \mathbb{R}^q$ such that $\|d\| \le 1$, that

$$|v^\varepsilon(p' + td) - v^\varepsilon(p') - t \nabla v^\varepsilon(p')^\top d| \le \frac{L t^2}{2} \quad \text{for all } \varepsilon \in (0, \delta). \tag{6}$$

Also, because there is a local uniform Lipschitz constant, the smoothing gradients $\nabla v^\varepsilon(p')$ are locally bounded for small $\varepsilon > 0$ and p' close to p. This implies that there is $\overline{v} \in \limsup_{\varepsilon \searrow 0, p' \to p} \nabla v^\varepsilon(p')$. Taking limits on

(6) when $\varepsilon \searrow 0$, $p' \to p$ and using (5), we conclude that all partial derivatives of v exist at p and that $\overline{v}$ is unique and is the gradient vector of v at p. □

For an extended-valued function $v : \mathbb{R}^q \to \overline{\mathbb{R}}$ and a parameter $\varepsilon \geq 0$, we define the set-valued mapping of ε-approximate unconstrained minimizers of the function v as

$$\varepsilon - \operatorname{Arg\,min}_p v := \{p \in \mathbb{R}^q : v(p) \leq \varepsilon + \inf_p v\}.$$

A sequence of functions $v_k : \mathbb{R}^q \to \overline{\mathbb{R}}$ is said to converge epigraphically, see [RW09, Proposition 7.2], to $v : \mathbb{R}^q \to \overline{\mathbb{R}}$ if the following two conditions hold for all $p \in \mathbb{R}^q$:

$$\liminf_k v_k(p_k) \geq v(p) \quad \text{for all} \quad p_k \to p\,,$$

and

$$\limsup_k v_k(p_k) \leq v(p) \quad \text{for some} \quad p_k \to p\,.$$

The notion of epigraphical convergence of functions is tightly related to the convergence of minimizers thanks to the following theorem, adapted from [RW09, Theorem 7.31].

Theorem 1 *For a sequence of extended-valued functions v_k converging epigraphically to v, the following holds.*

(i) $\inf v_k \to \inf v$ *if and only if for all $\varepsilon > 0$ there is a compact set $B \subset \mathbb{R}^q$ such that $\inf_B v_k \leq \inf v + \varepsilon$ for all k large enough,*

(ii) $\limsup_k \{\varepsilon_k - \mathit{Arg}\min v_k\} \subset \mathit{Arg}\min v$ *for all $\varepsilon_k \to 0$.*

3 Decomposition method induced by our smoothing technique

The approach outlined in this section was introduced in [BSS20] as a tool to smooth, approximate, and regularize value functions of fully parameterized convex optimization problems. Here, we revisit/adapt those results under the light of smoothed solution mappings (rather than the value function), considering a prototypical problem, dropping the subindex a in (1) and the assumption about uniqueness of solutions. Thanks to the continuity and differentiability properties of the smoothing, the decomposition method presented below generates an epigraphically convergent sequence, as in Theorem 1.

3.1 Defining the smoothed problems

The conditions necessary for the theory in [BSS20] are gathered below; for their precise role and use see [BSS20].

Assumption 2 In the prototypical problem

$$\mathcal{X}^*(p) := \operatorname{Arg\,min}_x \{f(x,p) : B(p)x = b(p), \quad g_i(x,p) \leq 0 \ \text{ for } i = 1,\dots,m\}\,, \tag{7}$$

the functions $f(\cdot,p)$ and $g_i(\cdot,p)$, $i = 1,\dots,m$, are convex for all $p \in \mathbb{R}^q$, and all the data is sufficiently smooth both in x and in p. Also, the rows of the matrices $B(p)$ are linearly independent for all p and the Slater condition holds for all p: for each p there exists $\hat{x}(p)$ such that $B(p)\hat{x}(p) = b(p)$ and $g_i(\hat{x}(p),p) < 0$ for $i = 1,\dots,m$.

Since problem (1) is not assumed to have a solution for all p, we may have dom $\mathcal{X}^* \subsetneq \mathbb{R}^q$. Note that while we allow the solution set $\mathcal{X}^*(p)$ to be empty, the feasible set of problem (7) cannot be empty.

In what follows, we adopt the convention that $\log \alpha = -\infty$ if $\alpha \leq 0$. Given the parameters $\varepsilon > 0$ and $\mu \geq 0$, a parametric Tikhonov-regularized log-barrier penalty function is used to approximate problem (7) by

$$x^{\varepsilon,\mu}(p) := \arg\min_x \{f(x,p) + \varepsilon\phi^\mu(x,p) : B(p)x = b(p)\}\,, \tag{8}$$

where

$$\phi^\mu(x,p) := -\sum_{i=1}^m \log\{-g_i(x,p)\} + \frac{\mu}{2}\|x\|^2\,.$$

The regularized solution $x^{\varepsilon,\mu}(p)$ is written in lower case, because it is unique when it exists (under our assumptions). If $\mu = 0$, the solution $x^{\varepsilon,\mu}(p)$ exists if $\mathcal{X}^*(p)$ is bounded [DS99], [MZ98]. The solution always exists if $\mu > 0$ (even when $\mathcal{X}^*(p) = \emptyset$), because in that case the objective function of (8) is strongly convex and problem (8) is assumed to have a nonempty feasible set.

Theorem 3 (Proposition 2 in [BSS20]) *Let $v(p)$ and $v^{\varepsilon,\mu}(p)$ be the optimal value functions of problems* (7) *and* (8)*, respectively. Under Assumption 2, for all $p \in D :=$ dom $\mathcal{X}^* \cap$ dom $x^{\varepsilon,\mu}$ it holds that:*

$$v(p) \leq v^{\varepsilon,\mu}(p) \leq v(p) + m\varepsilon + \frac{\varepsilon\mu}{2} \min_{x \in \mathcal{X}^*(p)} \|x\|^2, \tag{9}$$

and

$$\frac{\mu}{2} \min_{x \in \mathcal{X}^*(p)} \|x\|^2 + m \geq \frac{\mu}{2} \|x^{\varepsilon,\mu}(p)\|^2 . \tag{10}$$

Remark 1 The inequality $v(p) \leq v^{\varepsilon,\mu}(p)$ always holds, even if $x^{\varepsilon,\mu}(p)$ is not well-defined or $\mathcal{X}^*(p) = \emptyset$. If $x^{\varepsilon,\mu}(p)$ is not well-defined, then $v^{\varepsilon,\mu}(p)$ is understood as an infimum. It should also be noted that the domain D is independent of both $\varepsilon > 0$ and $\mu \geq 0$.

We are naturally interested in conditions under which the lower semicontinuous closure of the smoothing coincides with v, i.e., (5) holds. Theorem 3 is instrumental to identify a path to follow. We would first have to guarantee boundedness of the term $\min_{x \in \mathcal{X}^*(p)} \|x\|^2$. Second, we would need to show that v is continuous relative to subsets of dom $\mathcal{X}^*$. Then, condition (5) would follow from taking limits in (9).

The first requirement depends on the following weak assumption, called restricted inf-compactness condition in [Guo+14] (later rediscovered independently in [BSS20], through an unrelated computationally oriented approach):

$$\limsup_{p' \in \text{dom } \mathcal{X}^*, p' \to p} \left\{ \min_{x \in \mathcal{X}^*(p')} \|x\|^2 \right\} < +\infty \quad \text{for all } p \in \text{dom } \mathcal{X}^* . \tag{11}$$

Condition (11) means that the solution mapping $\mathcal{X}^*$ has a locally bounded selection over its domain. Note that (11) holds automatically if the feasible sets in (7) are uniformly bounded for $p \in$ dom $\mathcal{X}^*$ (which by itself is a rather natural assumption, holding in many cases of interest).

When the regularization parameter $\mu \geq 0$ is taken as a function of the smoothing parameter ε, we shall use the computational version of (11), given by

$$\limsup_{\varepsilon \searrow 0, p' \in D, p' \to p} \|x^{\varepsilon,\mu(\varepsilon)}(p')\| < +\infty \quad \text{for all} \quad p \in D := \text{dom } \mathcal{X}^* \cap \text{dom } x^{\varepsilon,\mu(\varepsilon)}. \tag{12}$$

Note that assumption (12) does not need $\varepsilon\mu(\varepsilon) \to 0$ to make sense. It refers only to boundedness, not to convergence of the regularized solutions to the actual solution.

Regarding the second issue above, related to continuity of the value functions, the following simple example shows the behaviour of the lower semicontinuous closure of the smoothing for parameters outside of dom $\mathcal{X}^*$.

Example 1 Consider the problem

$$\mathcal{X}^*(p) = \text{Arg} \min_x \{px : x \geq 0\} = \begin{cases} \emptyset & \text{if } p < 0, \\ \{x : x \geq 0\} & \text{if } p = 0, \\ \{0\} & \text{if } p > 0 . \end{cases}$$

The value function $v(p) = \inf_x \{px : x \geq 0\}$ fails to be lower semicontinuous at $p = 0$. Let us consider $\mu > 0$ fixed. Regarding the smoothing, it holds that

$$\text{lsc } v^{\varepsilon,\mu}(p) = \begin{cases} -\infty \text{ if} & p \leq 0, \\ 0 \quad \text{if} & p > 0 . \end{cases}$$

As is easy to see, if $\mu > 0$, for all problems satisfying Assumption 2 we have that

$$\text{lsc } v^{\varepsilon,\mu}(p) = -\infty \quad \text{for all} \quad p \in \text{int } (\mathbb{R}^q \setminus \text{dom } \mathcal{X}^*).$$

The lower semicontinuous closure lsc v of the value function coincides with v at all points except at $p = 0$, where lsc $v(0) = -\infty$.

Since under Assumption 2 the value function v can fail to be lower semicontinuous, it may not coincide with its lower semicontinous closure. Analyzing how smoothing behaves for the example above led

us to the following interesting result, that reveals the role of our smoothing as a tool to change a problem like $\min_{p\in P} v(p)$, which can fail to have a solution, to a sequence of problems $\min_{p\in P} v^{\varepsilon,\mu}(p)$, all having solutions.

Theorem 4 (On lower semicontinuous closures) *Under Assumption 2 and* (11)*, for $\mu > 0$ fixed, it holds that*

$$\text{lsc } v^{\varepsilon,\mu}(p) = \text{lsc } v(p) := \liminf_{p'\to p} v(p') \quad \textit{for all} \quad p \in \mathbb{R}^q.$$

Proof If $p \in \text{int dom } \mathcal{X}^*$ or $p \in \text{int } (\mathbb{R}^q \setminus \text{dom } \mathcal{X}^*)$, the statement is trivial because the following stronger property holds:

$$\lim_{\varepsilon\searrow 0, p'\to p} v^{\varepsilon,\mu}(p') = v(p) = \text{lsc } v(p).$$

Take p on the boundary of dom $\mathcal{X}^*$. Note that lsc $v(p) = -\infty$. Take sequences $\varepsilon_k \searrow 0$ and $p_k \to p$. We trivially have that $\liminf_k v^{\varepsilon_k,\mu}(p_k) \geq -\infty = \text{lsc } v(p)$. Now note that there is a sequence $p_k \to p$ such that $v(p_k) = -\infty$. Recall that for all fixed $p \in \mathbb{R}^q$ we have $v^{\varepsilon,\mu}(p) \to v(p)$ when $\varepsilon \searrow 0$. Therefore, for each k we can find $\varepsilon_k > 0$ such that $v^{\varepsilon_k,\mu}(p) < -k$. We just proved that there are sequences $\varepsilon_k \searrow 0$ and $p_k \to p$ such that $\limsup_k v^{\varepsilon_k,\mu}(p_k) = -\infty \leq v(p)$. □

Our next two examples illustrate the fact that non-convex value functions may not be continuous on the interior of their domain.

Example 2 Take $v(p) = \min_x\{px : x \geq 0\}$. Note that $v(p) = 0$ if $p \geq 0$ and $v(p) = -\infty$ if $p < 0$ and that $-v$ is a convex function. Now note that the effective domain of v is $\mathbb{R}$, while the one of $-v$ is $\{p : p \geq 0\}$. The function $-v$ is continuous on the interior of its domain, because it is convex. But the example shows that the same statement is false for a concave function. However, looking at the solution mapping $\mathcal{X}^*(p) = \text{Arg}\min_x\{px : x \geq 0\}$ we can realize that both v and $-v$ are continuous relative to the domain of the solution mapping, which is $\{p : p \geq 0\}$. In general, we are only able to prove Lipschitz continuity locally, on int dom $\mathcal{X}^*$, and continuity relative to compact subsets of dom $\mathcal{X}^*$.

Example 3 Consider the problem $\min\{x : x \geq 0\}$. Let us look at the dual function $\psi(p)$ as a value function. Then, $\psi(p) = \inf_x\{(1-p)x\}$ is such that $\psi(p) = 0$ if $p = 1$ and $\psi(p) = -\infty$ otherwise. Again, ψ is not continuous on the interior of its domain, but is so relative to the domain of the solution mapping $\mathcal{X}^*(p)$ of the dual problem, which is $\{1\}$. For this example, int dom $\mathcal{X}^* = \emptyset$, while the claim about continuity on compact subsets of the domain is useful.

3.2 Continuity and differentiability of the objects induced by smoothing

We proceed by stating further continuity/differentiability properties of our smooth approximations.

Theorem 5 (Continuity of smoothed value function and solution mapping) *Under Assumption 2, we have:*

(i) If condition (11) *holds, the value function of problem* (7) *is locally Lipschitz continuous relative to* int dom $\mathcal{X}^*$*, and continuous relative to any compact subset of* dom $\mathcal{X}^*$.

(ii) Suppose the solution mapping of (7) *is a singleton ($\mathcal{X}^*(p) = x(p)$ for all $p \in$ dom $\mathcal{X}^*$). Assume, in addition, that:*

- *(a) either $\mu = 0$ and the feasible set of problem* (7) *is uniformly bounded on* dom $\mathcal{X}^*$*,*
- *(b) or $\mu > 0$ is fixed and condition* (11) *holds,*
- *(c) or, most generally, conditions* (11) *and* (12) *hold, and $\mu(\varepsilon)$ is bounded for small $\varepsilon > 0$.*

Then for any compact set $D \subset$ dom $\mathcal{X}^* \cap$ dom $x^{\varepsilon,\mu(\varepsilon)}$, it holds that

$$\lim_{\varepsilon\searrow 0, p'\in D, p'\to p} x^{\varepsilon,\mu(\varepsilon)}(p') = x(p) \quad \textit{for all } p \in D. \tag{13}$$

Proof Let us prove the first assertion in item (i). Fix any $\mu > 0$. Note that dom $v^{\varepsilon,\mu} = \mathbb{R}^q$, and that the local boundedness of $\nabla_p v^{\varepsilon,\mu}(\cdot)$ for small $\varepsilon > 0$ on int dom $\mathcal{X}^*$ follows from [BSS20, Theorem 2]. Then the claim follows from [Che12] and Proposition 1, noting that for $p \in$ int dom $\mathcal{X}^*$ we have

$$\partial_c v(p) \subset \text{conv}\left\{\limsup_{\varepsilon\searrow 0, p'\to p} \nabla_p v^{\varepsilon,\mu}(p')\right\}.$$

For the second assertion in item (i), fix again any $\mu > 0$. Note that dom $v^{\varepsilon,\mu} = \mathbb{R}^q$ and $v^{\varepsilon,\mu}$ is a smooth function on $\mathbb{R}^q$ (see [BSS20, Corollary 1]). Formulas (9) and (11) imply that $v^{\varepsilon,\mu}(p)$ converges uniformly to $v(p)$ over compact subsets of dom $\mathcal{X}^*$. Because $v(p)$ is the uniform limit of the smooth functions $v^{\varepsilon,\mu}(p)$ on compact regions of dom $\mathcal{X}^*$, it has to be continuous relative to the compact subset of dom $\mathcal{X}^*$ taken.

Regarding item (ii), the first step is to show that (12) holds. This is the case if (a) holds, or if (b) holds via (10) or if (c) holds. In other words, the conditions (a), (b) or (c) imply that no subsequence of $x^{\varepsilon,\mu(\varepsilon)}(p')$ becomes unbounded when $\varepsilon \searrow 0$ and $p' \to p$ for $p' \in D$. Then, condition (9) and the continuity of v on D (shown in item (i)) imply that all subsequences of $x^{\varepsilon,\mu(\varepsilon)}(p')$ accumulate on the singleton $\mathcal{X}^*(p)$. In other words, (9) implies that

$$v(p') \leq f(x^{\varepsilon,\mu(\varepsilon)}(p'), p') \leq v(p') + m\varepsilon + \frac{\varepsilon\mu(\varepsilon)}{2} \min_{x \in \mathcal{X}^*(p')} \|x\|^2.$$

Taking limits on the last inequality under (a), (b) or (c) shows that $x^{\varepsilon,\mu(\varepsilon)}(p')$ converges as claimed. Note that the continuity of v over D is used. □

The multipliers of the affine equality constraints in (7) and (8), denoted respectively by $\lambda(p)$ and $\lambda^{\varepsilon,\mu}(p)$, play an important role in the calculations of the first-order derivatives of the value function and solution mapping. Because $B(p)$ has linearly independent rows, $\lambda^{\varepsilon,\mu}(p)$ exists and is unique whenever $x^{\varepsilon,\mu}(p)$ exists.

Differentiability properties of the regularization, relevant to solving problems (8) (and thus eventually (4)) computationally, are summarized below. In particular, explicit formulas for the derivatives of the primal-dual solutions of (8) are given. To this end, in the optimality conditions for (8) the sign of the Lagrange multiplier $\lambda^{\varepsilon,\mu}(p)$ is taken so that the following identity holds:

$$\nabla_x f(x^{\varepsilon,\mu}(p), p) + \varepsilon \nabla_x \phi^{\mu}(x^{\varepsilon,\mu}(p), p) - B(p)^\top \lambda^{\varepsilon,\mu}(p) = 0.$$

Theorem 6 (Theorem 1 in [BSS20]) *Under Assumption 2, the following holds.*

(i) If $\mu > 0$, then $x^{\varepsilon,\mu}(p)$ and $\lambda^{\varepsilon,\mu}(p)$ are well-defined and are continuously differentiable in p, for all $p \in \mathbb{R}^q$.

(ii) If $\mu = 0$ and the constraints $x \geq 0$ are present in problem (7), suppose that $\mathcal{X}^*(p)$ is pointwise bounded on dom $\mathcal{X}^*$*. Then $x^{\varepsilon,\mu}(p)$ and $\lambda^{\varepsilon,\mu}(p)$ are well-defined on* dom $\mathcal{X}^*$*, and are continuously differentiable in p, for all $p \in$* int dom $\mathcal{X}^*$*.*

(iii) Whenever $x^{\varepsilon,\mu}(p)$ and $\lambda^{\varepsilon,\mu}(p)$ are well-defined and their partial derivatives with respect to the r-th coordinate of $p \in \mathbb{R}^q$ exist, they are given as the solution of the linear system below:

$$\begin{bmatrix} J^{\varepsilon,\mu}(p) & -B(p)^\top \\ B(p) & 0 \end{bmatrix} \begin{bmatrix} \partial_r x^{\varepsilon,\mu}(p) \\ \partial_r \lambda^{\varepsilon,\mu}(p) \end{bmatrix} = \begin{bmatrix} \theta_r^{\varepsilon,\mu}(p) + \varepsilon \varphi_r^{\varepsilon,\mu}(p) \\ \beta_r^{\varepsilon,\mu}(p) \end{bmatrix}, \tag{14}$$

where, for $r = 1, \ldots, q$,

$$\begin{aligned}
J^{\varepsilon,\mu}(p) &:= \nabla^2_{xx} f(x,p) + \varepsilon \nabla^2_{xx} \phi^{\mu}(x,p)\Big|_{x = x^{\varepsilon,\mu}(p)}, \\
\theta_r^{\varepsilon,\mu}(p) &:= -\frac{\partial \nabla_x f(x,p)}{\partial p_r}\Big|_{x = x^{\varepsilon,\mu}(p)} + \frac{\partial B(p)^\top}{\partial p_r} \lambda^{\varepsilon,\mu}(p), \\
\varphi_r^{\varepsilon,\mu}(p) &:= -\frac{\partial \nabla_x \phi^{\mu}(x,p)}{\partial p_j}\Big|_{x = x^{\varepsilon,\mu}(p)} \\
\beta_r^{\varepsilon,\mu}(x) &:= \frac{\partial b(p)}{\partial p_r} - \frac{\partial B(p)}{\partial p_r} x^{\varepsilon,\mu}(p).
\end{aligned}$$

3.3 Decomposition method across the agents

When in our problem (1)-(2) the total number $|\mathcal{A}|$ of agents is large, a direct solution approach can become too time consuming, and possibly even impossible. Having laid out the elements of our proposal, we are now in position to give our solution method, that converges epigraphically and allows for decomposition across agents. The method is first described conceptually, and then each step is explained further. Details of the actual implementation are not discussed, for now.

Algorithm 7 (Smoothing decomposition of equilibrium problems)

Input and initialization. Choose $p_1 \in \mathbb{R}^q$, $\sigma_1 > 0$, $\varepsilon_1 > 0$ and $\mu_1 \geq 0$. Set $k := 1$.
Step 1: Price Iterate. Find p_{k+1} as an approximate local solution of

$$\operatorname{Arg\,min}_p \left\{ F(x_{\mathcal{A}}^{\varepsilon_k,\mu_k}(p)) : p \in \Pi \cap \tilde{\Pi} \right\},$$

taking p_k as the starting point for the solver applied (for instance, Ipopt). During this solve, when the value and gradient of the objective function F are needed (at points $\hat{p} \neq p_k$), e.g., when the callbacks for the objective are made, solve the subproblem below to get the required information.

Step 1.1: Smoothed Subproblems. For each $a \in \mathcal{A}$ solve (for instance, with Ipopt)

$$\min_x \left\{ f_a(x,\hat{p}) - \varepsilon_k \sum_{i=1}^{m} \log\{-g_{ai}(x,\hat{p})\} + \frac{\varepsilon_k \mu_k}{2} \|x\|^2 : B_a(\hat{p})x = b_a(\hat{p}) \right\}.$$

Let $x_a^{\varepsilon_k,\mu_k}(\hat{p})$ denote the unique minimizer in this problem, with the associated unique Lagrange multiplier $\lambda_a^{\varepsilon_k,\mu_k}(\hat{p})$. Compute the Jacobians of $x_a^{\varepsilon_k,\mu_k}(\hat{p})$ and $\lambda_a^{\varepsilon_k,\mu_k}(\hat{p})$, and use this information to compute the gradient of $F(x_{\mathcal{A}}^{\varepsilon_k,\mu_k}(\hat{p}))$.

Step 2: Stopping Test. If $F(x_{\mathcal{A}}^{\varepsilon_k,\mu_k}(p_{k+1}))$ stabilized relative to previous iterations, Stop.
Step 3: Update Smoothing Parameters. Determine σ_{k+1}, $\varepsilon_{k+1} > 0$ and $\mu_{k+1} \geq 0$, so that

$$|F(x_{\mathcal{A}}^{\varepsilon_{k+1},\mu_{k+1}}(p_{k+1})) - F(x_{\mathcal{A}}(p_{k+1}))| \leq \sigma_{k+1}.$$

Step 4: Loop. Set $k := k + 1$ and go back to Step 1.

In practice, the rule to choose the parameters ε_k, μ_k has to be computationally inexpensive. In our case, once ε_k is determined, possible options are taking a fixed $\mu_k = \mu > 0$ or setting $\mu_k = \sqrt{\varepsilon_k}$, or some similar simple choice. We use a one-dimensional bisection procedure to select ε_k, and therefore having a one-dimensional relation for μ_k is useful. For simplicity, in what follows we take $\mu_k = \mu > 0$. Other situations are dealt with by means of the parameterization just explained. In order to define a proper management of the sequence $\varepsilon_k \searrow 0$, we define the quantity

$$d^{\varepsilon}(p) := |v^{\varepsilon,\mu}(p) - v(p)|, \quad \mu = \mu(\varepsilon),$$

where

$$v^{\varepsilon,\mu}(p) := \begin{cases} F(x_{\mathcal{A}}^{\varepsilon,\mu}(p)) & p \in \Pi, \\ +\infty & p \notin \Pi, \end{cases} \quad \text{and} \quad v(p) := \begin{cases} F(x_{\mathcal{A}}(p)) & p \in \Pi, \\ +\infty & p \notin \Pi. \end{cases}$$

The need to handle dynamically ε_k appeared because the output can be very poor when using some fixed sequence determined beforehand. This fact was confirmed by some runs in which $\varepsilon_k = 1/k$ resulted in slow convergence, while setting $\varepsilon_k = 1/(k^2)$ decreased the parameter too fast, making numerical errors dominate the iterative process. Fixing a priori exogenous values for that sequence appeared not to be suitable, particularly regarding accuracy. In some experiments, for a given p_k, we noticed that $\varepsilon_k = 10^{-6}$ is the maximal value for which $d^{\varepsilon_k}(p_k) \leq 10^{-2}$. As a result, setting $\varepsilon_k = 1/k$ would require 10^6 iterations to bring the regularized agent's problem sufficiently close to the original one.

For these reasons, in Step 3 of Algorithm 7 we use the available information, and manage the parameter so that ε_k is about the largest value possible for which $d^{\varepsilon_k}(p_k) \leq \sigma_k$, for σ_k a decreasing sequence going to zero, depending only on certain available past information. Of course, we also do not want to spend an unreasonable amount of time calibrating ε_k based on the last point p_k. The values $p_1 \in \Pi$ and $\varepsilon_1, \mu_1, \sigma_1 > 0$ are an input of the algorithm. We take ε_1 "large". Given $p_{k+1} \in \Pi$ and $\varepsilon_k > 0$, for $k \geq 1$ we set

$$\sigma_{k+1} = \min \left\{ \frac{0.5 d^{\varepsilon_k}(p_{k+1})}{k+1}, \sigma_k \right\}.$$

The value $\varepsilon_{k+1} > 0$ is obtained as follows. We know that setting $\varepsilon = 0$ yields $d^{\varepsilon}(p_{k+1}) = 0$ and want to find ε_{k+1} such that $d^{\varepsilon_{k+1}}(p_{k+1}) \leq \sigma_{k+1}$ taking into account that $d^{\varepsilon}(p)$ is continuous in ε. The interval to make this search is $[0, C_{k+1}]$ with $C_{k+1} = 1.3\varepsilon_k$. We allow for $\varepsilon_{k+1} > \varepsilon_k$. Then, we start a bisection procedure on $[0, C_{k+1}]$ trying to match the value σ_{k+1}, and stop once we find a value ε_{k+1} such that $d^{\varepsilon_{k+1}}(p_{k+1}) \leq \sigma_{k+1}$, or the maximal number of trials is reached, or $d^{\varepsilon_{k+1}}(p_{k+1})$ is close enough to σ_{k+1}. Note that close enough here is understood loosely. For example, with distance between $0.1\sigma_{k+1}$ and $0.3\sigma_{k+1}$. The point is that this calculation does not need to be precise. However, it has to be precise enough to guarantee that $d^{\varepsilon_{k+1}}(p_{k+1})$

decreases to zero, hence ensuring that the regularized models get closer and closer to the true model near an accumulation point of the sequence p_{k+1}.

Another relevant issue for implementation is when to stop the minimization process when solving problems in Step 1. In order not to blindly rely on the stopping criteria of the solver, the focus should be put on robust decrease. If the price sequence converges, then $d^{\varepsilon_k}(p_k)$ yields a current estimate of how close the regularized model is to the true model. If while solving (4) consecutive iterates have objective function values differing in less than $0.5d^{\varepsilon_k}(p_k)$, this hints that the current ε_k, μ_k may not be meaningful in providing solutions for the original problem. These rules are based on comparison of functional values and not on gradient information (with our epigraphical convergence approach to the problem, the unknown property of gradient consistency cannot be exploited algorithmically).

We finish with some useful practical considerations for the implementation. In practice we want to avoid being subject to numerical instabilities associated with $\varepsilon_k > 0$ being too small. To circumvent this issue, we add triggers to store the record (the best iterate while solving one instance of the price problem in Step 1) as well as the best iterates between different instances of that problem. Also, while building the matrix of the linear system (14), we have to make sure that the iterate lies in the interior of the feasible set with some safeguards, because factors like $g_{ai}(x_a^{\varepsilon,\mu}(p), p)^{-2}$ appear in the diagonal (and so numerical errors are amplified to the square). For instance, if $g_{ai}(x_a^{\varepsilon,\mu}(p), p) = -10^{-5} < 0$, the term in the diagonal is 10^{10}. Currently, the management of these numerical errors is done the simplest way possible, because it is enough for the applications we tried. However, as Algorithm 7 may need to solve thousands of optimization problems, being robust to failure is essential. These safeguards are not detailed in our description, but the guiding principles are listed below:

1. use an adaptive rule for ε_k, μ_k and σ_k so that problems in Step 1 are closer and closer to the model (2),
2. early stop the subproblem solution in Step 1 if the difference in objective function values on consecutive iterations is smaller than a fraction of $d^{\varepsilon_k}(p_k)$, and
3. carefully manage numerical errors and failures, taking into account that the algorithm is solving thousands of optimization problems, and that even if a fraction of those fail, the algorithm has to keep running.

The convergence result below is based on relation (13) holding for the solution mappings $x_{\mathcal{A}}^{\varepsilon,\mu}(p)$ and $x_{\mathcal{A}}(p)$. There are two qualitatively distinct cases to keep in mind, these are: $\liminf \mu_k > 0$ and $\liminf \mu_k = 0$. The assumption that the feasible sets of the problems are uniformly bounded is enough for both cases. Note that if $\liminf \mu_k > 0$, we can allow for unbounded solution sets as well, enforcing condition (ii)(b) or (ii)(c) of Theorem 5.

Theorem 8 (Convergence result for Algorithm 7) *Fix a compact set $K \subset \Pi$. Let Assumption 2 for the agents' problems* (1) *hold, and assume in addition that*

(i) either the feasible sets of problems (1) *are uniformly bounded for all $p \in K$,*
(ii) or if in Algorithm 7 we have $\liminf \mu_k > 0$, the solution mappings of problems (1) *have locally bounded selections for all $p \in K$,*
(iii) or, most generally, conditions (12) *and* (11) *hold, and $K \subset \operatorname{dom} \mathcal{X}_a^* \cap \operatorname{dom} x_a^{\varepsilon_k,\mu_k}$ for all $a \in \mathcal{A}$.*

Then, the sequence of functions v^{ε_k,μ_k} converges epigraphically to v over K.

Proof Recall that $\{\mu_k\}$ is bounded. If (i) holds, then conditions (12) and (11) hold. Therefore, (ii)(c) of Theorem 5 holds and the conclusion follows. If (ii) holds, then (12) holds because of (10) and the fact that solution mappings have locally bounded selections, which is condition (11). Therefore, the conclusion follows again. Finally, (iii) easily implies (ii)(c) of Theorem 5. □

The importance of the set D considered in (13) lies in ensuring epigraphical convergence on a feasible region containing an accumulation point of the sequence p_k. For instance, for the WEP defined below the epigraphical convergence result does not apply on the boundary of Π. Also note that condition in item (ii) of the theorem above is equivalent to requiring satisfaction of (11) for problems (1), uniformly over $p \in K$.

4 Solving deterministic equilibrium problems

As mentioned in the introduction, PATH [FM99] is an established code to solve equilibrium problems. Being a Newton-type method (see, e.g., [IS14, Chapter 5.2.2]), when PATH works, it tends to get higher precision than our smoothing. However, when solving Walrasian equilibrium problems, it appears that one needs to

calibrate PATH parameters very carefully, because certain utility functions in the agent's problems (1) are ill-conditioned and degeneracies occur (singular-basis, using PATH-related language). Even with careful tuning, this leads to failures with a certain frequency, especially for larger instances.

We next describe a first family of problems employed in our experiments, and then benchmark the performance of Algorithm 7 against PATH.

4.1 Deterministic Walrasian equilibrium problems

A Walrasian Equilibrium Problem is defined for an economy with agents in a set $\mathcal{A}$, of cardinality $|\mathcal{A}|$. There are n goods whose prices form a vector $p \in \mathbb{R}^n$. The agent's consumption is a vector $x \in \mathbb{R}^n_+$, so in (1)-(2) the dimensions are $q = n$ and $n_a = n$ for all $a \in \mathcal{A}$. Each agent has at its disposal an initial amount of goods $e_a \in \mathbb{R}^n$, called endowment, that is worth $p^\top e_a$. The consumption benefit is measured using some strictly concave utility function $u_a(\cdot)$. Accordingly, given a price p, the agent's optimal decisions are

$$x_a(p) := \arg\max_{x \geq 0} \{u_a(x) : p^\top x \leq p^\top e_a\}. \tag{15}$$

With respect to problem (1), the objective function therein is $f_a(x, p) := -u_a(x)$. As explained below, for some utilities the budget constraint can be replaced by an equality constraint, so the only inequalities in (1) refer to non-negativity of the decision variable. These are handled by the penalty ϕ^μ introduced in (8).

The global problem (2) minimizes the excess supply, which results in the following:

$$p^* \in \arg\min_{p \in \Delta} \frac{1}{|\mathcal{A}|^2} \left\| \sum_{a \in \mathcal{A}} x_a(p) - \sum_{a \in \mathcal{A}} e_a \right\|^2 , \text{ where } \Delta := \{\tau > 0 : \sum_{j=1}^n \tau_j = 1\}. \tag{16}$$

In particular, Δ is the unit simplex of strictly positive prices. Notice that letting

$$\bar{x}_{\mathcal{A}}(p) := \frac{1}{|\mathcal{A}|} \sum_{a \in \mathcal{A}} x_a(p) \quad \text{and} \quad \bar{e}_{\mathcal{A}}(p) := \frac{1}{|\mathcal{A}|} \sum_{a \in \mathcal{A}} e_a(p), \tag{17}$$

the objective function in (16) amounts to measuring the distance between consumption and endowment, averaging over all agents in the economy. If the price succeeds in clearing the market, the optimal value in (16) is zero, the information that we employ to assess the quality of the output of the solution methods compared in our experiments.

As already mentioned above, the solution set in (15) is empty for some $p \notin \Delta$. This depends on a property of the utility function, said to be *non-satiable*. In economics, non-satiation is the assumption that a consumer will always benefit from additional consumption. In consumer theory, the utility refered to as having constant elasticity of substitution (CES) satisfies the property of non-satiation. This strictly concave function combines in one number the preference of consuming n types of goods, assuming they have an elasticity of substitution $1 \neq b > 0$ and that a number $\gamma_j > 0$ indicates the preference for the j-th good:

$$\texttt{CES}(x) := \left(\sum_{j=1}^n \gamma_j^{\frac{1}{b}} x_j^{\frac{b-1}{b}} \right)^{\frac{b}{b-1}} \quad \text{for all} \quad x > 0. \tag{18}$$

When $b \to \infty$ the goods behave like perfect substitutes and as $b \searrow 0$ they behave like perfect complements. The property that $\texttt{CES}(\alpha x) = \alpha \texttt{CES}(x)$ for $\alpha > 0$ and $x > 0$ reflects the belief that the intrinsic utility of a vector of goods lies on the proportion between the goods, rather than on their magnitude. This is confirmed by the fact that the consumption $x_a(p)$ solving problem (15) written with utility CES satisfies the relation $x_a(\alpha p) = x_a(p)$ for any $\alpha > 0$ and $p \in \Delta$. The domain of the CES utility function is the whole space for $b > 1$.

The deterministic WEP model (15)-(16) represents an exchange economy, without production. Agents cannot spend more than the worth of the initial endowment. When the price of some good becomes non-positive, the feasible set in (15) gets unbounded. In this case, if the utility $u_a(x)$ is of non-satiable type, the agent will try to spend more and more, yielding decisions $x_a(p)$ that are not well-defined.

These issues with the utility functions would result in failure for methods that need the problem data to be defined, and have Lipschitz derivatives, on the whole space. By contrast, they do not affect our method, because the smooth regularized solutions of problem (8) use the utility derivatives only locally, near the smoothed consumption of the agents. Another advantage of our approach is that, in practice, larger values of

$\varepsilon > 0$ tend to imply less abrupt changes on the functions involved in (20). As a result, the agents' problems are somehow easier to solve. Computationally, as the method moves $\varepsilon > 0$ from larger to smaller values, the agent problem (15) goes from easier to harder.

4.2 First numerical benchmark

For non-satiable utilities, the budget inequality constraint in (15) can be changed by an equality without changing the solution set. The agents' problems take the form

$$x_a(p) = \arg\max_{x \geq 0} \{\texttt{CES}_a(x) : p^\top x = p^\top e_a\},$$

and so our smoothing penalizes the non-negativity constraint. The problems in Step 1.1 of Algorithm 7 are given by

$$x_a^{\varepsilon_k,\mu_k}(p_k) := \arg\min_x \{-\texttt{CES}_a(x, p_k) - \varepsilon_k \sum_{i=1}^{n} \log(x_i) + \frac{\varepsilon_k \mu_k}{2}\|x\|^2 : p_k^\top x = p_k^\top e_a\}.$$

These problems are solved with Ipopt [WB05], setting the mu-target option available for the solver so that $x_a^{\varepsilon_k,\mu_k}(p_k)$ is automatically computed instead of $x_a(p)$. An optimized build of Ipopt with state-of-the-art linear algebra software Pardiso [KFS18] is essential for reproducing performance. Otherwise, speed may be sacrificed. See [KFS18] for one of the advanced applications of Pardiso.

Our smoothing method, denoted by Alg. 7 in the tables, was coded in CPP (g++ 7.5.0) with the initial value $\varepsilon_1 = 1$ and keeping $\mu_k = 1$ fixed. All experiments are run on an Intel i7 1.90GHz machine, but using only one thread when comparing with PATH. The operating system is Ubuntu 18.04.3 LTS.

We start with an example from [DJW17], with the same CES utility for all the agents, taking in (18) the preferences $\gamma_j = 1.0$ and the power $b = 0.5$. Endowments are also the same, $e_a = (1, \ldots, 1)$ for all $a \in \mathcal{A}$.

In this setting, the equilibrium price is unique and known, $p^* = (\frac{1}{n}, \ldots, \frac{1}{n})$, where n is the number of goods. In other words, if all agents have the same buying power and the same preferences, all the goods have the same price at equilibrium.

We consider $|\mathcal{A}| = 2$ agents exchanging $n \in \{2, 10, 20, 30\}$ goods. Notice that the objective function in (16) measures the capacity of the economy in clearing the market. Hence, the columns "Initial/Clearing" (both for Alg. 7 and PATH-Clearing) correspond, respectively, to the initial and final objective function values (found with our approach and with PATH). For each column, the output in Table 4.1 reports the average and standard deviation, computed by repeating the experiments of each configuration four times. For this case, four repetitions is enough because the standard deviation is small, as can be seen in Table 4.1.

n	Initial Clearing	Alg. 7-Time (sec)	Alg. 7-Clearing	PATH-Time (sec)	PATH-Clearing
2	0.03 / 0.04	0.03 / 0.01	10^{-6} / 10^{-6}	0.01 / 0.00	10^{-33} / 10^{-33}
10	2.67 / 1.96	0.13 / 0.01	10^{-5} / 10^{-5}	0.57 / 1.09	10^{-33} / 10^{-33}
20	3.43 / 1.79	0.16 / 0.02	10^{-4} / 10^{-4}	0.07 / 0.00	10^{-32} / 10^{-32}
30	4.19 / 0.83	0.21 / 0.02	10^{-4} / 10^{-4}	0.13 / 0.00	10^{-32} / 10^{-32}

Table 4.1: Comparison between Alg. 7 and PATH for deterministic WEP with symmetric agents.

Except for the experiments with $n = 10$, where PATH seems to have struggled for one run, the Newtonian updates in PATH make the output more precise and faster for this set of problems, as expected.

The second experiment, originally from [Sca73] and also reported in [DJW17], has $|\mathcal{A}| = 5$ agents and $n = 10$ goods. In (18), the CES utilities for the agents are defined with the following values for the power

$$b_1 = 2.0\,, \quad b_2 = 1.3\,, \quad b_3 = 3.0\,, \quad b_4 = 0.2\,, \quad b_5 = 0.6\,,$$

and the coefficients γ reported in Table 4.2. The table also contains the initial endowments of the agents.

We sampled ten prices in Δ, and both algorithms were executed starting from these prices. The average and standard deviation for the quantities of interest are reported in Table 4.3. The final prices agree with those found in [DJW17]. Note that while PATH is still more precise, the smoothing approach is now faster (to termination).

a	γ_1	γ_2	γ_3	γ_4	γ_5	γ_6	γ_7	γ_8	γ_9	γ_{10}
1	1.0	1.0	3.0	0.1	0.1	1.2	2.0	1.0	1.0	0.7
2	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0
3	9.9	0.1	5.0	0.2	6.0	0.2	8.0	1.0	1.0	0.2
4	1.0	2.0	3.0	4.0	5.0	6.0	7.0	8.0	9.0	10.0
5	1.0	13.0	11.0	9.0	4.0	0.9	8.0	1.0	2.0	10.0
a	$e_{a,1}$	$e_{a,2}$	$e_{a,3}$	$e_{a,4}$	$e_{a,5}$	$e_{a,6}$	$e_{a,7}$	$e_{a,8}$	$e_{a,9}$	$e_{a,10}$
1	0.6	0.2	0.2	20.0	0.1	2.0	9.0	5.0	5.0	15.0
2	0.2	11.0	12.0	13.0	14.0	15.0	16.0	5.0	5.0	9.0
3	0.4	9.0	8.0	7.0	6.0	5.0	4.0	5.0	7.0	12.0
4	1.0	5.0	5.0	5.0	5.0	5.0	5.0	8.0	3.0	17.0
5	8.0	1.0	22.0	10.0	0.3	0.9	5.1	0.1	6.2	11.0

Table 4.2: CES utility coefficients and initial endowments for the agents.

	Initial Clearing	Alg. 7-Time (sec)	Alg. 7-Clearing	PATH-Time (sec)	PATH-Clearing
Avg.	7056.23	2.30	10^{-5}	10.24	10^{-10}
Std.	3401.70	1.76	10^{-2}	5.45	10^{-8}

Table 4.3: Comparison between Alg. 7 and PATH for [Sca73] problem.

4.3 Scaling capabilities of Algorithm 7

In order to explore decomposition with respect to the number of agents, we extend the previous example to an economy with $n = 80$ goods, for $|\mathcal{A}|$ ranging from 2 to 640 agents.

$\lvert\mathcal{A}\rvert$	Initial Clearing	Alg. 7-Clearing	Alg. 7-Time (sec)
2	15.56 / 5.11	0.66 / 0.21	10.26 / 2.63
4	10.00 / 10.59	0.12 / 0.08	8.74 / 4.45
8	5.17 / 1.28	0.02 / 0.01	22.77 / 13.47
10	5.43 / 0.61	0.02 / 0.00	18.20 / 9.98
20	6.79 / 2.15	0.01 / 0.02	17.61 / 9.32
40	5.79 / 2.05	0.01 / 0.01	45.70 / 38.24
80	4.45 / 1.79	0.00 / 0.00	37.21 / 26.06
160	3.91 / 0.92	0.01 / 0.01	53.06 / 15.26
320	4.90 / 0.58	0.04 / 0.01	246.49 / 168.07
640	4.36 / 0.50	0.06 / 0.02	654.76 / 387.20

Table 4.4: Illustration of the decomposition properties of the smoothing with respect to the number of agents.

We sample the γ coefficients of the utilities in the box $[0.1, 1]^n$, and sample b in $[0.1, 0.9]$ for all agents. For each configuration we start from four random initial prices. The results, reported as average / standard deviation, are shown in Table 4.4 and in Figure 4.1. We use eight threads.

5 Decomposition of stochastic hierarchical problems

Another model for the WEP represents an economy with production subject to uncertain delivery, that we handle in two stages. We give the corresponding formulation, then solve some instances using Algorithm 7, and finish with a decomposition method that exploits the two-stage structure of the agents' problems.

5.1 Stochastic Walrasian Equilibrium

In this model, consumption decisions taken in the first stage are x_a^0 as well as another vector z_a of production activity levels. The production effort takes place in the first stage and the resulting goods are delivered in the second stage in uncertain amounts, represented with a set $\mathcal{S}$ of equiprobable scenarios, of cardinality $|\mathcal{S}|$. The consumption decided in the second stage for the s-th scenario is x_a^s.

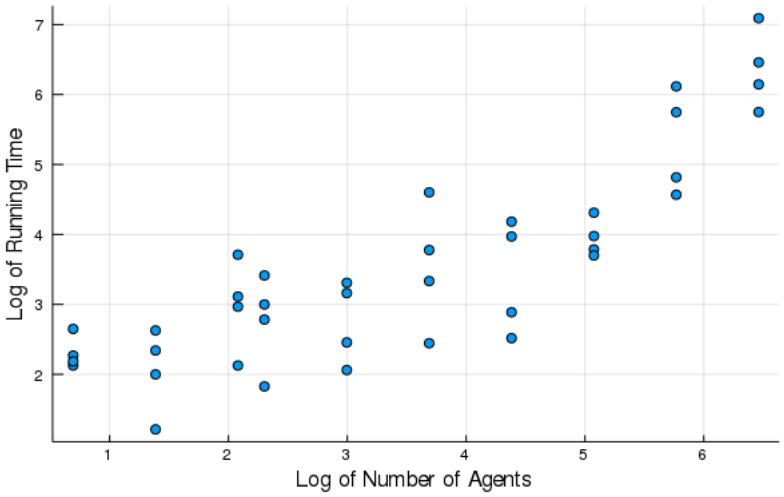

Fig. 4.1: In log scale, the running times grow linearly with respect to the number of agents. Each dot is one experiment. (We use log scale because the total number of agents grows exponentially.)

In the first stage, we need $B_a^0 z$ goods to start the production, so that the resulting amount of goods for each second-stage scenario is $B_a^s z$. The cost of the production effort in the first stage is $(p^0)^\top B_a^0 z$, and the profit in the s-th scenario is $(p^s)^\top B_a^s z$. The resulting stochastic version of the agent's problem (taken from [DJW17]) with CES utilities and corresponding equality in the budget constraint is

$$\left(z_a(p), x_a^0(p), x_a^1(p), \ldots, x_a^{|\mathcal{S}|}(p)\right) = \begin{cases} \underset{z, x^0, x^1, \ldots, x^{|\mathcal{S}|}}{\arg\max} & \texttt{CES}_a^0(x^0) + \dfrac{1}{|\mathcal{S}|} \displaystyle\sum_{s \in \mathcal{S}} \texttt{CES}_a^s(x^s) \\ \text{s.t.} & (p^0)^\top x^0 = (p^0)^\top e_a^0 - (p^0)^\top B_a^0 z, \\ & (p^s)^\top x^s = (p^s)^\top e_a^s + (p^s)^\top B_a^s z \quad \text{for } s \in \mathcal{S}, \\ & z, x^0, x^1, \ldots, x^{|\mathcal{S}|} \geq 0. \end{cases} \tag{19}$$

Again, we need to analyze whether the feasible set of problem (19) is bounded near a strictly positive price p. By examining the first constraint of (19), it suffices to take the matrices B_a^s so that all production activities z_a have a positive cost if the prices of all goods are positive. Under this condition, feasible sets are uniformly bounded and each agent problem (19) satisfies Assumption 2 because the equality constraints are indeed linearly independent.

In (19), uniqueness of the optimal consumption follows from strict concavity of the utility functions. Uniqueness of the production levels $z_a(p)$ depends on the existence of a unique solution in z to the system

$$z \geq 0, \quad (p^0)^\top B_a^0 z = (p^0)^\top e_a^0 - (p^0)^\top x^0, \quad (p^s)^\top B_a^s z = -(p^s)^\top e_a^s + (p^s)^\top x^s \quad \text{for } s \in \mathcal{S}\,,$$

for fixed values $x_a^0, \ldots, x_a^{|\mathcal{S}|}$. This issue is related to market completeness. In an incomplete market, the total of activities (sometimes called financial instruments) is less than the amount of scenarios (or future states of the world). In this case, the system above is over-determined and uniqueness of the production levels depends on the scenario realizations. If the market is complete (there are at least as many activities as scenarios), $z_a(p)$ may be non-unique because the system above is under-determined. In any case, the smoothing of the solution mappings is always well-defined. The assumption on the single-valuedness of $x_a(p)$ is used only for the convergence analysis. In practice, independently of assumptions, the algorithm will minimize the objective.

The price that best balances demand and supply is determined similarly to (16), considering the expected value of the clearing for the second stage

$$p^* \in \arg\min_{p \in \Pi} \left\{ \|\bar{x}_{\mathcal{A}}^0(p) - \bar{e}_{\mathcal{A}}^0 + \overline{B}^0 z_{\mathcal{A}}(p)\|^2 + \frac{1}{|\mathcal{S}|} \sum_{s \in \mathcal{S}} \|\bar{x}_{\mathcal{A}}^s(p) - \bar{e}_{\mathcal{A}}^s - \overline{B}^s z_{\mathcal{A}}(p)\|^2 \right\}. \tag{20}$$

Similarly to (17), in this problem we defined the mean consumption, endowment and production transfers, averaged for all the economy, for each scenario $s \in \{0\} \cup \mathcal{S}$:

$$\bar{e}^s_{\mathcal{A}} := \frac{1}{|\mathcal{A}|}\sum_{a\in\mathcal{A}} e^s_a\,, \qquad \bar{x}^s_{\mathcal{A}}(p) := \frac{1}{|\mathcal{A}|}\sum_{a\in\mathcal{A}} x^s_a(p)\,, \quad \overline{B}^s z_{\mathcal{A}}(p) := \frac{1}{|\mathcal{A}|}\sum_{a\in\mathcal{A}} B^s_a z_a(p)$$

(for simplicity, the first-stage is referred to as the 0th scenario).

5.2 Numerical experiments

We report on performance of the proposed method for some examples found in the literature [Sca73; DJW17; Sch12], and analyze the impact of properly managing the parameter ε_k. Since the method described in [DJW17] is of derivative-free type, it needs to solve many subproblems to estimate the derivatives, and thus is not comparable with our approach. The stochastic setting leads to larger problems, and PATH starts to fail too often to collect meaningful information. Thus, PATH results are not reported. As for the method in [Sch12], it cannot be used to obtain decomposition.

The results for the deterministic Scarf's instance are shown in Figure 5.1. It reports the probability of each strategy delivering a reduction of x% of the initial clearing within the normalized budget time in $[0, 1]$. Because the strategy $1/k$ does not drive ε_k to zero fast enough, it lacks precision.

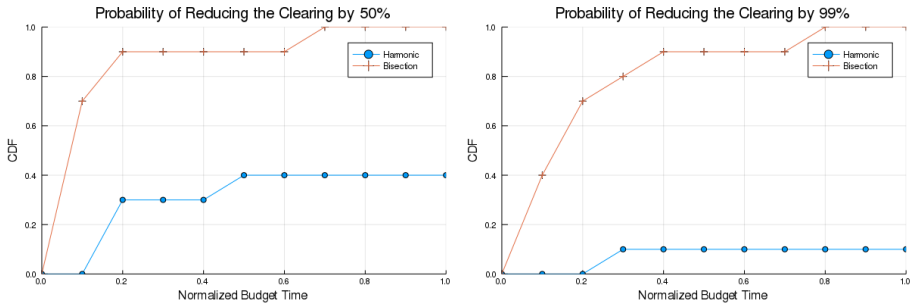

Fig. 5.1: Data profile showing the impact on performance of the management of the parameter ε_k for the Scarf's instance.

In the second test-case, our smoothing method is applied to the stochastic WEP described in [DJW17]. It has $|\mathcal{S}| = 9$ scenarios, $|\mathcal{A}| = 5$ agents, $n = 7$ goods, all with production. The problem data for the stochastic instance was obtained directly from the authors of [DJW17]. For this example we also use only one thread. The results are reported in Table 5.2. Note that the final clearing is not as close to zero as in the deterministic setting (for instance, see Table 4.4). Here, the algorithm converged to a price with positive clearing. This can happen, since the method is guaranteed to find only local solutions. Moreover, for the stochastic setting the equilibrium is not guaranteed to exist due to issues with complete and incomplete markets.

	Initial Clearing	Alg. 7-Time (sec)	Alg. 7-Clearing
Avg.	28512.10	81.27	5.41
Std.	60275.63	14.54	1.61

Table 5.1: Results for the stochastic WEP from [DJW17].

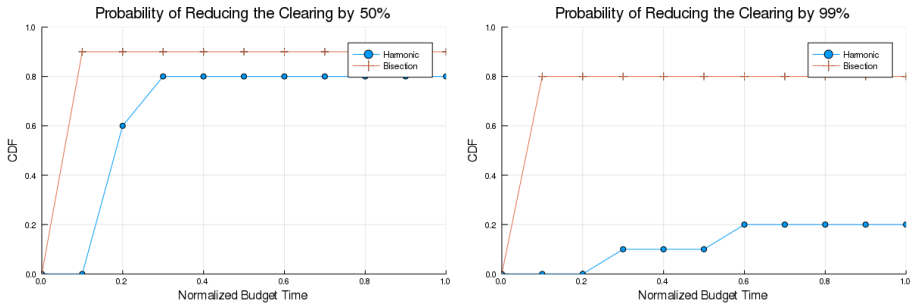

Fig. 5.2: Data profile showing the impact on performance of the management of the parameter ε_k for the stochastic instance.

We finish our computational analysis with showing that it is worth to invest some effort in the choice of the smoothing parameter ε_k in Step 3 of Algorithm 7. The strategy described in Section 3.3 is inner/outer iteration scheme. In the inner step we minimize the smooth approximation and in the outer step we measure the quality of the approximation and calibrate ε_k. In this case k is the number of outer iterations. For the strategy $\varepsilon_k = 1/k$ to make practical sense, the meaning of k is different. In this case, it is the number of times the objective function oracle of the smooth approximation is called by Ipopt. Due to these differences, we compare both strategies using time budgets in a data profile.

More precisely, for a given instance of WEP, we run both algorithms recording the price iterates of the smooth approximations and the time when the iterate is generated. After the runs finish, we compute separately the true clearing for the price iterates. This procedure is repeated for ten random initial prices for each instance. We then make data profiles [MW09; DM02] reporting the probability of the best results until a certain time delivering a percentage decrease of the initial clearing. The instances are the ones from [Sca73] reported above, and the stochastic instance is from [DJW17], also used above. Those results are reported in Figure 5.2. Note that again the $1/k$ strategy is not as accurate, and also that the bisection strategy for ε_k finds a good solution early in the process.

5.3 Inducing decomposition across scenarios

For simplicity, we consider the regularization parameter $\mu > 0$ fixed during this section, and all feasible sets in consideration uniformly bounded. The previous developments assume that the agents solve a parameterized convex problem with no special structure. In this section, we show how a two-stage stochastic structure can be used to obtain decomposition across scenarios. Computational implementation of this additional decomposition is a technically more involved topic, which we do not pursue here. But the proof of convergence can be done with the tools already developed in this paper.

The agents' problems are given by

$$\begin{cases} \min\limits_{x^0,\ldots,x^{|\mathcal{S}|}} & f_a^0(x^0,p) + \frac{1}{|\mathcal{S}|}\sum_{s\in\mathcal{S}} f_a^s(x^s,p) & \\ \text{s.t.} & B_a^i(p)x^i + C_a^i(p)x^0 = b_a^i(p) & \text{for } i = 0,\ldots,|\mathcal{S}|, \\ & g_a^i(x^i,p) + h_a^i(x^0,p) \leq 0 & \text{for } i = 0,\ldots,|\mathcal{S}|. \end{cases} \tag{21}$$

In particular, if the problem above is the stochastic WEP, the inequalities are only $z \geq 0$ and $x^s \geq 0$ for $s \in \{0,\ldots,|\mathcal{S}|\}$. In spite of some abuse of notation, the first stage variables of the current two-stage problem would be $x^0 := (z, x_w^0)$, where x_w^0 is the first stage consumption of the stochastic WEP. In other words, the activities are first stage decisions for the stochastic WEP. One has to map carefully the constraints of the general problem to that of the stochastic WEP. The objective function would be given by

$$f_a^s(x^s,p) = -\texttt{CES}_a^s(x^s) \quad \text{for all } s \in \{0,\ldots,|\mathcal{S}|\}.$$

Now, there is one price for each scenario/stage configuration. This implies that $\mathit{\Pi}$ has the form

$$\mathit{\Pi} = \mathit{\Delta}^{1+|\mathcal{S}|}, \quad \text{where} \quad \mathit{\Delta} = \{\tau > 0 : \sum_j \tau_j = 1\}.$$

To write the optimal value reformulation it is useful to define the first stage feasible set

$$X_a^0(p) := \{x^0 \in \mathbb{R}^{n_a} : \quad B_a^0(p)x^0 = b_a^0(p), \quad g_a^0(x^0,p) \leq 0\}.$$

The last problem, when written using the usual value function reformulation becomes

$$x_a^0(p) := \arg\min_{x^0}\{f_a^0(x^0,p) + \frac{1}{|\mathcal{S}|}\sum_{s\in\mathcal{S}} Q_a^s(x^0,p) : x^0 \in X_a^0(p)\}, \tag{22}$$

where $x_a^s(x^0,p)$ and $Q_a^s(x^0,p)$ are the unique solution and value function of the scenario subproblem:

$$\min_{x^s}\{f_a^s(x^s,p) : \quad B_a^s(p) + C_a^s(p)x^0 = b_a^s(p), \quad g_a^s(x^s,p) + h_a^s(x^0,p) \leq 0\}. \tag{23}$$

Note that $Q_a^s(x^0,p)$ is convex in x^0 for a fixed p, because: (i) x^0 influences the right-hand side of the equality constraints in a linear manner, and (ii) the function $g_a^s(x^s,p) + h_a^s(x^0,p)$ is jointly convex in x^s and x^0 (since (21) is convex for all p).

The idea is to parameterize the second stage decisions of the agent's problem, denoted by $x_a^s(x^0, p)$, as functions of both the first stage decision x^0 and the price vector $p \in \Pi$. As usual, to get a smooth approximation to this function we define, for $s = 1, \ldots, S$, the Tikhonov-regularized log-barrier penalty for the problem (23) by

$$\phi_a^{s,\mu}(x^s, p, x^0) := -\sum_{i=1}^{m_a^s} \log\{-g_{ai}^s(x^s, p) - h_{ai}^s(x^0, p)\} + \frac{\mu}{2}\|x^s\|^2.$$

If the scenario subproblems have only the inequalities $x^s \geq 0$, then

$$\phi_a^{s,\mu}(x^s, p, x^0) := -\sum_{i=1}^{m_a^s} \log\{x_i^s\} + \frac{\mu}{2}\|x^s\|^2.$$

Then, for any $\varepsilon > 0$, the smoothed approximation of $x_a^s(x^0, p)$ and the associated smoothing for the value function of the scenario subproblems are given by

$$x_a^s(\varepsilon, x^0, p) := \arg\min_{x^s} \{f_a^s(x^s, p) + \varepsilon\phi_a^{s,\mu}(x^s, p, x^0) : \quad B_a^s(p) + C_a^s(p)x^0 = b_a^s(p)\} \tag{24}$$

and

$$Q_a^s(\varepsilon, x^0, p) := f_a^s(x_a^s(\varepsilon, x^0, p), p).$$

Although $Q_a^s(x^0, p)$ is convex in x^0 for a fixed p, the smoothed value function $Q_a^s(\varepsilon, x^0, p)$ is only "approximately" convex [BSS20, Lemma 3]. The approximation of $Q_a^s(x^0, p)$ which is guaranteed to be convex in x^0 is given by

$$P_a^s(\varepsilon, x^0, p) := Q_a^s(\varepsilon, x^0, p) + \varepsilon\phi_a^{s,\mu}(x_a^s(\varepsilon, x^0, p), p, x^0).$$

By [BSS20, Lemma 3], we know that $\varepsilon\phi_a^{s,\mu}(x_a^s(\varepsilon, x^0, p), p, x^0) \to 0$ in a controlled manner when $\varepsilon \searrow 0$.

Because of the stated blanket assumptions, problems (24) satisfy the Slater condition for all p and x^0, the rows of $B_a^s(p)$ are linearly independent, and problem (24) is convex. Therefore, if $\mu > 0$, it follows that $x_a^s(\varepsilon, x^0, p)$ is well-defined and smooth, even if $x_a^s(x^0, p)$ is not. The same holds for $P_a^s(\varepsilon, x^0, p)$. Note now that one smooth approximation (not necessarily from above) for the objective function of master problem (22) is given by

$$x_a^0(\varepsilon, p) := \arg\min_{x^0} \{f_a^0(x^0, p) + \frac{1}{|\mathcal{S}|}\sum_{s\in\mathcal{S}} P_a^s(\varepsilon, x^0, p) : x^0 \in X_a^0(p)\}. \tag{25}$$

Note that in (25) we need to use $P_a^s(\varepsilon, x^0, p)$ instead of $Q_a^s(\varepsilon, x^0, p)$, so that problem (25) is smooth and convex for all p, our assumptions hold and the results apply.

The solution mapping $x_a^0(\varepsilon, p)$ is not guaranteed to be a smooth function of p. For this reason, we have to deal with the Tikhonov-regularized log-barrier for the first stage problem, given by

$$\phi_a^{0,\mu}(x^0, p) := -\sum_{i=1}^{m_a^0} \log\{-g_{ai}^0(x^0, p)\} + \frac{\mu}{2}\|x^0\|^2.$$

Again, when the first stage problem is linear and the only inequality constraints are $x^0 \geq 0$, we see that

$$\phi_a^{0,\mu}(x^0, p) := -\sum_{i=1}^{m_a^0} \log\{x_i^0\} + \frac{\mu}{2}\|x^0\|^2.$$

The smooth approximation $x_a^0(\delta, \varepsilon, p)$ for the first stage decision $x_a^0(\varepsilon, p)$ is given by

$$x_a^0(\delta, \varepsilon, p) := \arg\min_{x^0} \{f_a^0(x^0, p) + \frac{1}{|\mathcal{S}|}\sum_{s\in\mathcal{S}} P_a^s(\varepsilon, x^0, p) + \delta\phi_a^{0,\mu}(x^0, p) : B_a^0(p)x^0 = b_a^0(p)\}. \tag{26}$$

For $s = 1, \ldots, S$, the approximations of the second stage decisions are given by the composition

$$x_a^s(\varepsilon, p) := x_a^s(\varepsilon, x_a^0(\varepsilon, \varepsilon, p), p).$$

Note that we take $\delta = \varepsilon$ above for simplicity. The whole vector $x_a^\varepsilon(p)$ is thought of as the concatenation

$$x_a^\varepsilon(p) := (x_a^0(\varepsilon, p), x_a^1(\varepsilon, p), \ldots, x_a^S(\varepsilon, p)).$$

If $\mu > 0$ and the functions defining the agents' problems are sufficiently smooth, the regularized solution mappings $x_a^s(\varepsilon, p)$ above are smooth. The numerical issue is that to compute their derivatives based on formulas (14), we need higher-order derivatives of the problem's data.

Convergence of the joint agent-wise and scenario-wise decomposition is still based on proving that property (3) holds. For this purpose, denote by $Q_a^0(p)$ the objective function of problem (22), by $Q_a^0(\varepsilon, p)$ the objective function of problem (25), and by $Q_a^0(\delta, \varepsilon, p)$ the objective function of problem (26). Inequality (9) applied to the scenario subproblems yields, for $s = 1, \ldots, S$,

$$Q_a^s(p, x^0) \leq Q_a^s(\varepsilon, p, x^0) \leq Q_a^s(p, x^0) + \varepsilon m_a^s + \varepsilon \frac{\mu}{2} \|x_a^s(x^0, p)\|^2 \quad \text{for all } p \in p, x^0 \in X_a^0(p).$$

Then, because (11) is a blanket assumption if $\mu > 0$, we can use (10) to find $K > 0$ such that for all $p \in \Pi$ and $x^0 \in X_a^0(p)$ we have

$$f_a^0(x^0, p) + \frac{1}{|\mathcal{S}|} \sum_{s \in \mathcal{S}} Q_a^s(x^0, p) \leq f_a^0(x^0, p) + \frac{1}{|\mathcal{S}|} \sum_{s \in \mathcal{S}} Q_a^s(\varepsilon, x^0, p) \leq f_a^0(x^0, p) + \frac{1}{|\mathcal{S}|} \sum_{s \in \mathcal{S}} Q_a^s(x^0, p) + \varepsilon K. \tag{27}$$

Note that we used $Q_a^s(\varepsilon, x^0, p)$ in (27). To take into account $P_a^s(\varepsilon, x^0, p)$, we have (see [BSS20, Lemma 3]) to recall that for any $\kappa > 0$ it holds that $|Q_a^s(\varepsilon, x^0, p) - P_a^s(\varepsilon, x^0, p)| \leq \kappa$ for all $\varepsilon > 0$ small enough, uniformly on x^0 and p, since we are assuming uniformly bounded feasible sets. Then, for $\varepsilon = \varepsilon(\kappa) > 0$ small enough we have that

$$\begin{aligned} f_a^0(x^0, p) + \frac{1}{|\mathcal{S}|} \sum_{s \in \mathcal{S}} Q_a^s(x^0, p) - \kappa &\leq f_a^0(x^0, p) + \frac{1}{|\mathcal{S}|} \sum_{s \in \mathcal{S}} P_a^s(\varepsilon, x^0, p) \\ &\leq f_a^0(x^0, p) + \frac{1}{|\mathcal{S}|} \sum_{s \in \mathcal{S}} Q_a^s(x^0, p) + \varepsilon K + \kappa. \end{aligned}$$

Taking the infimum on the last inequality over $x^0 \in X_a^0(p)$, we obtain that

$$Q_a^0(p) - \kappa \leq Q_a^0(\varepsilon, p) \leq Q_a^0(p) + \varepsilon K + \kappa \quad \text{for all } p \in \Pi.$$

Applying inequality (9) and condition (11) to problem (26), and possibly modifying $K > 0$, we obtain that

$$Q_a^0(\varepsilon, p) \leq Q_a^0(\delta, \varepsilon, p) \leq Q_a^0(\varepsilon, p) + \delta K \quad \text{for all } p \in \Pi.$$

The last inequality combined with (5.3) gives

$$Q_a^0(p) - \kappa \leq Q_a^0(\delta, \varepsilon, p) \leq Q_a^0(p) + \varepsilon K + \delta K + \kappa \quad \text{for all } p \in \Pi.$$

We now take limits on the last inequality in ε and δ for a fixed κ. Then, we take let $\kappa \searrow 0$. It follows that

$$\lim_{\varepsilon, \delta \searrow 0, p' \in p, p' \to p} x_a^0(\delta, \varepsilon, p') = x_a^0(p) \quad \text{for all } p \in \Pi.$$

Therefore, we also have that

$$\lim_{\varepsilon \searrow 0, p' \in p, p' \to p} x_a^s(\varepsilon, p') = x_a^s(p) \quad \text{for all } p \in \Pi.$$

We conclude that property (3) holds, which ensures convergence of the presented agent-wise and scenario-wise decomposition scheme.

6 Concluding remarks

We finish commenting on some issues related to gradient consistency, decomposition, and risk.

Gradient consistency for the smoothing technique in question was proven in [BSS20] in the case when the value function is convex. For nonconvex value functions, the issue is considerably more involved and still open. But under second-order sufficient conditions for the subproblems, classical statements can be made, because the nonregularized solution mappings are smooth themselves. However, this is not our focus indeed and we do not assume second-order conditions or strict complementarity.

Augmented Lagrangian methods for usual optimization problems with general lower-level constraints are considered in [And+08], and generalizations of such methods for GNEPs in [KS16]. The augmented

Lagrangian method in [Sch12] solves essentially the same WEP as we deal with here, and solution times much better than in [DJW17] are reported. One issue with these methods though, is that decomposition is not possible because of the quadratic penalizations of the constraints. For instance, in [Sch12] experiments with up to thirty agents only are reported. With our method, subject to memory limitations and distributed computing capabilities, in principle we can solve problems with as many agents as required. The same holds for the number of scenarios. Decomposition techniques are important for this type of problems, because the number of variables for an equivalent formulation is the product of the number of agents, scenarios, and goods (thus, potentially huge).

Issues related to the existence of equilibrium prices for social welfare problems where agents solve risk-averse multistage stochastic optimization problems are considered in [PF18], and their relation to the classical first and second welfare theorems from economics are explored under some circumstances. This problem class is the same that we deal with here, the difference being that we aim to compute efficiently the best price possible according to the function F above, independently whether this is an equilibrium price or not. Following [PF18], we could consider a risk measure for the stochastic WEP, proceeding as in in [BSS20], with the decisions of the agents that go in (2) being risk-averse. Therefore, conceptually, there is no difference and the algorithm we present here remains essentially the same, with the same decomposition properties. For instance, if F stands for a measure of excess supply like in our numerical section, an equilibrium price p^* could be characterized by $F(p^*) := F(x_{\mathcal{A}}(p^*)) = 0$. On the other hand, if with risk measures and incomplete risk-markets it becomes impossible to perfectly balance supply and demand, the best p^* would be such that $F(p^*) > 0$. The non-existence of an equilibrium price is not an impediment to run a minimization procedure to find a price with a smaller associated objective function value (or with a smaller clearing) than what we currently have (the starting guess).

Acknowledgments The authors are grateful to the reviewers for valuable comments and remarks. Research of the second author is partly funded by CNPq Grant 306089/2019-0, CEPID CeMEAI, and PRONEX–Optimization. The third author is supported by CNPq Grant 303913/2019-3, by FAPERJ Grant E-26/202.540/2019, by PRONEX–Optimization and by the Russian Foundation for Basic Research grant 19-51-12003 NNIOa.

References

[And+08] R. Andreani, E. G. Birgin, J. M. Martínez, and M. L. Schuverdt. "On Augmented Lagrangian Methods with General Lower-Level Constraints". *SIAM Journal on Optimization* 18.4 (2008), pp. 1286–1309 (cit. on pp. 3, 18).

[ASP14] A. Alvarado, G. Scutari, and J.-S. Pang. "A New Decomposition Method for Multiuser DC-Programming and Its Applications". *IEEE Transactions on Signal Processing* 62.11 (2014), pp. 2984–2998 (cit. on p. 3).

[BH16] J. V. Burke and T. Hoheisel. "Epi-convergence Properties of Smoothing by Infimal Convolution". *Set-Valued and Variational Analysis* 25.1 (2016), pp. 1–23 (cit. on p. 4).

[BHK13] J. V. Burke, T. Hoheisel, and C. Kanzow. "Gradient Consistency for Integral-convolution Smoothing Functions". *Set-Valued and Variational Analysis* 21.2 (2013), pp. 359–376 (cit. on pp. 3, 4).

[BSS20] P. Borges, C. Sagastizábal, and M. Solodov. "A Regularized Smoothing Method for Fully Parameterized Convex Problems with Applications to Convex and Nonconvex Two-Stage Stochastic Programming". *Mathematical Programming, DOI:* `http://dx.doi.org/10.1007/s10107-020-01582-2` (2020) (cit. on pp. 2, 3, 5–8, 17–19).

[CF10] W. Chung and J. D. Fuller. "Subproblem Approximation in Dantzig-Wolfe Decomposition of Variational Inequality Models with an Application to a Multicommodity Economic Equilibrium Model". *Operations Research* 58.5 (2010), pp. 1318–1327 (cit. on p. 3).

[Che12] X. Chen. "Smoothing methods for nonsmooth, nonconvex minimization". *Mathematical Programming* 134.1 (2012), pp. 71–99 (cit. on pp. 2, 3, 7).

[DJW17] J. Deride, A. Jofré, and R. J.-B. Wets. "Solving Deterministic and Stochastic Equilibrium Problems via Augmented Walrasian". *Computational Economics* 53.1 (2017), pp. 315–342 (cit. on pp. 2, 3, 12, 14–16, 19).

[DM02] E. D. Dolan and J. J. Moré. "Benchmarking optimization software with performance profiles". *Mathematical Programming* 91.2 (2002), pp. 201–213 (cit. on p. 16).

[DS99] L. M. G. Drummond and B. F. Svaiter. "On Well Definedness of the Central Path". *Journal of Optimization Theory and Applications* 102.2 (1999), pp. 223–237 (cit. on p. 6).

[FC05] J. D. Fuller and W. Chung. "Dantzig-Wolfe Decomposition of Variational Inequalities". *Comput. Econ.* 25 (4 2005), pp. 303–326 (cit. on p. 3).

[FM99] M. C. Ferris and T. S. Munson. "Interfaces to PATH 3.0: Design, Implementation and Usage". In: *Computational Optimization*. Springer US, 1999, pp. 207–227 (cit. on pp. 3, 10).

[FPS11] F. Facchinei, V. Piccialli, and M. Sciandrone. "Decomposition Algorithms for Generalized Potential Games". *Comput. Optim. Appl.* 50.2 (2011), pp. 237–262 (cit. on p. 3).

[GF10] S. A. Gabriel and J. D. Fuller. "A Benders Decomposition Method for Solving Stochastic Complementarity Problems with an Application in Energy". *Computational Economics* 35.4 (2010), pp. 301–329 (cit. on p. 3).

[Guo+14] L. Guo, G.-H. Lin, J. J. Ye, and J. Zhang. "Sensitivity Analysis of the Value Function for Parametric Mathematical Programs with Equilibrium Constraints". *SIAM Journal on Optimization* 24.3 (2014), pp. 1206–1237 (cit. on p. 6).

[IS14] A. Izmailov and M. Solodov. *Newton-type methods for optimization and variational problems*. Springer Series in Operations Research and Financial Engineering. Springer, Cham, 2014 (cit. on pp. 4, 10).

[JJBW02] A. Jofré and R. J.-B. Wets. "Continuity Properties of Walras Equilibrium Points". *Annals of Operations Research* 114.1-3 (2002), pp. 229–243 (cit. on p. 2).

[KF19] Y. Kim and M. C. Ferris. "Solving equilibrium problems using extended mathematical programming". *Mathematical Programming Computation* 11.3 (2019), pp. 457–501 (cit. on p. 3).

[KFS18] D. Kourounis, A. Fuchs, and O. Schenk. "Toward the Next Generation of Multiperiod Optimal Power Flow Solvers". *IEEE Transactions on Power Systems* 33.4 (2018), pp. 4005–4014 (cit. on p. 12).

[KHF17] Y. Kim, O. Huber, and M. C. Ferris. "A structure-preserving pivotal method for affine variational inequalities". *Mathematical Programming* 168.1-2 (2017), pp. 93–121 (cit. on p. 3).

[KS12] A. A. Kulkarni and U. V. Shanbhag. "Revisiting Generalized Nash Games and Variational Inequalities". *Journal of Optimization Theory and Applications* (2012), pp. 1–12 (cit. on p. 3).

[KS16] C. Kanzow and D. Steck. "Augmented Lagrangian Methods for the Solution of Generalized Nash Equilibrium Problems". *SIAM Journal on Optimization* 26.4 (2016), pp. 2034–2058 (cit. on pp. 3, 18).

[LSS12] J. P. Luna, C. Sagastizábal, and M. Solodov. "A class of Dantzig–Wolfe type decomposition methods for variational inequality problems". *Mathematical Programming* 143.1-2 (2012), pp. 177–209 (cit. on p. 3).

[LSS13] J. P. Luna, C. Sagastizábal, and M. Solodov. "Complementarity and Game-Theoretical Models for Equilibria in Energy Markets: Deterministic and Risk-Averse Formulations" (2013), pp. 231–258 (cit. on p. 3).

[LSS16] J. P. Luna, C. Sagastizábal, and M. Solodov. "An approximation scheme for a class of risk-averse stochastic equilibrium problems". *Mathematical Programming* 157.2 (2016), pp. 451–481 (cit. on p. 3).

[MW09] J. J. Moré and S. M. Wild. "Benchmarking Derivative-Free Optimization Algorithms". *SIAM Journal on Optimization* 20.1 (2009), pp. 172–191 (cit. on p. 16).

[MZ98] R. D. Monteiro and F. Zhou. "On the Existence and Convergence of the Central Path for Convex Programming and Some Duality Results". *Computational Optimization and Applications* 10.1 (1998), pp. 51–77 (cit. on p. 6).

[PF18] A. Philpott and M. Ferris. "Dynamic Risked Equilibrium". *preprint* (2018) (cit. on p. 19).

[PFW16] A. Philpott, M. Ferris, and R. Wets. "Equilibrium, uncertainty and risk in hydro-thermal electricity systems". *Mathematical Programming* 157.2 (2016), pp. 483–513 (cit. on p. 1).

[RW09] T. Rockafellar and R. J.-B. Wets. *Variational Analysis*. Springer, 2009 (cit. on pp. 2–5).

[Sag12] C. Sagastizábal. "Divide to conquer: decomposition methods for energy optimization". *Mathematical Programming* 134.1 (2012), pp. 187–222 (cit. on p. 1).

[Sca73] H. E. Scarf. *The Computation of Economic Equilibria*. Yale University Press, 1973 (cit. on pp. 12, 13, 15, 16).

[Sch12] S. Schommer. "Computing equilibria in economies with incomplete markets, collateral and default penalties". *Annals of Operations Research* 206.1 (2012), pp. 367–383 (cit. on pp. 3, 15, 19).

[Scu+11] G. Scutari, D. P. Palomar, F. Facchinei, and J.-S. Pang. "Distributed dynamic pricing for MIMO interfering multiuser systems: A unified approach". *International Conference on NETwork Games, Control and Optimization* (2011) (cit. on p. 3).

[Scu+13] G. Scutari, F. Facchinei, P. Song, D. P. Palomar, and J.-S. Pang. “Decomposition by partial linearization in multiuser systems”. In: *2013 IEEE International Conference on Acoustics, Speech and Signal Processing*. IEEE, 2013 (cit. on p. 3).

[WB05] A. Wächter and L. T. Biegler. “On the implementation of an interior-point filter line-search algorithm for large-scale nonlinear programming”. *Mathematical Programming* 106.1 (2005), pp. 25–57 (cit. on pp. 2, 12).

[XWY14] M. Xu, S.-Y. Wu, and J. J. Ye. “Solving semi-infinite programs by smoothing projected gradient method”. *Computational Optimization and Applications* 59.3 (2014), pp. 591–616 (cit. on p. 3).

[XY13] M. Xu and J. J. Ye. “A smoothing augmented Lagrangian method for solving simple bilevel programs”. *Computational Optimization and Applications* 59.1-2 (2013), pp. 353–377 (cit. on p. 3).

[XYZ14] M. Xu, J. J. Ye, and L. Zhang. “Smoothing augmented Lagrangian method for nonsmooth constrained optimization problems”. *Journal of Global Optimization* 62.4 (2014), pp. 675–694 (cit. on p. 3).

[XYZ15] M. Xu, J. J. Ye, and L. Zhang. “Smoothing SQP Methods for Solving Degenerate Nonsmooth Constrained Optimization Problems with Applications to Bilevel Programs”. *SIAM Journal on Optimization* 25.3 (2015), pp. 1388–1410 (cit. on p. 3).