Interior-Point Methods in TensCalcTools

João P. Hespanha

April 3, 2020

Abstract

This document describes the optimization method used by TensCalcTools to solve constrained optimizations.

1 Constrained Optimization

Our goal is to find a vector $u^* \in \mathcal{U}$ that solves the optimization

$$f(u^*) = \min_{u \in \mathcal{U}} f(u), \tag{1}$$

with

$$\mathcal{U} := \{u \in \mathbb{R}^N : F(u) \geqslant 0,\ G(u) = 0\},$$

for given functions $f : \mathbb{R}^N \in \mathbb{R}$, $F : \mathbb{R}^N \to \mathbb{R}^M$, and $G : \mathbb{R}^N \to \mathbb{R}^K$.

1.1 Primal-dual method

The following duality-like result provides the motivation for a primal-dual-like method to solve the coupled minimizations in (1). It provides a set of conditions, involving an unconstrained optimization, that provide an approximation to the solution of (1).

Lemma 1 (Approximate equilibrium)**.** *Suppose that we have found* primal variables $u \in \mathbb{R}^N$ *and* dual variables $\lambda \in \mathbb{R}^M$, $\nu \in \mathbb{R}^K$ *that simultaneously satisfy all of the following conditions*[1]

$$L_f(u, \lambda, \nu) = \min_{\bar{u} \in \mathbb{R}^N} L_f(\bar{u}, \lambda, \nu), \tag{2a}$$

$$G(u) = \mathbf{0}_K, \tag{2b}$$

$$\lambda \geqslant \mathbf{0}_M, \qquad F(u) \geqslant \mathbf{0}_M, \tag{2c}$$

where

$$L_f(\bar{u}, \lambda, \nu) := f(\bar{u}) - \lambda \cdot F(\bar{u}) + \nu \cdot G(\bar{u}).$$

Then u approximately satisfies (1) *in the sense that*

$$f(u) \leqslant \epsilon_f + \min_{\bar{u} \in \mathcal{U}} f(\bar{u}), \qquad \epsilon_f := \lambda \cdot F(u). \tag{3}$$

□

[1] Given a vector $x \in \mathbb{R}^n$ and a scalar $a \in \mathbb{R}$, we denote by $x \geqslant a$ the entry-wise "greater than or equal to" comparison of each entry of x with a.

The following notation is used in (1)–(3) and below: Given an integer n, we denote by $\mathbf{0}_n$ and by $\mathbf{1}_n$ the n-vectors with all entries equal to 0 and 1, respectively. Given two vectors $x, y \in \mathbb{R}^n$ we denote by $x \geqslant y$ the entry-wise "greater than or equal to" comparison of the entries of x and y; and by $x \cdot y \in \mathbb{R}$, $x \odot y \in \mathbb{R}^n$, and $x \oslash y \in \mathbb{R}^n$ the inner product, entry-wise product, and entry-wise division of the two vectors, respectively.

Proof of Lemma 1. From the definitions of L_f, ϵ_f and using (2a), we conclude that

$$f(u) - \epsilon_f = \min_{\bar{u} \in \mathbb{R}^N} f(\bar{u}) - \lambda \cdot F(\bar{u}) + \nu \cdot G(\bar{u}).$$

Therefore

$$f(u) = \epsilon_f + \min_{u \in \mathbb{R}^N} f(u) - \lambda \cdot F(u) + \nu \cdot G(u) \leqslant \epsilon_f + \min_{u \in \mathcal{U}} f(u) - \lambda \cdot F(u) \leqslant \epsilon_f + \max_{\lambda \geqslant 0, \nu} \min_{u \in \mathcal{U}} f(u) - \lambda \cdot F(u),$$

where the first inequality is a consequence of the fact that $\mathcal{U} \subset \mathbb{R}^N$ and (2b), whereas the second inequality is a consequence of the fact that maximizing over $\lambda \geqslant 0$ will always lead to a larger value than what would be obtained for $\lambda \geqslant 0$, because of (2c). Because of (2b), we further conclude that

$$f(u) \leqslant \epsilon_f + \max_{\lambda \geqslant 0, \nu} \min_{u \in \mathcal{U}} f(u) - \lambda \cdot F(u) = \epsilon_f + \min_{\bar{u} \in \mathcal{U}} f(\bar{u}). \qquad \square$$

1.2 Interior-point primal-dual equilibria algorithm

The method proposed consists of using Newton iterations to solve a system of nonlinear equations on the primal variables $u \in \mathbb{R}^N$ and dual variables $\lambda \in \mathbb{R}^M$, $\nu \in \mathbb{R}^K$ introduced in Lemma 1. The specific system of equations consists of:

1. the first-order optimality conditions for the unconstrained minimizations in (2a)

$$\nabla_u L_f(u, \lambda, \nu) = \mathbf{0}_N, \tag{4}$$

   where $\nabla_u L_f$ denotes the gradient of L_f with respect to the variable u;

2. the equality conditions (2b); and

3. the equations

$$F(u) \odot \lambda = \mu \mathbf{1}_M, \tag{5}$$

   for some $\mu > 0$, which leads to

$$\epsilon_f := \lambda \cdot F(u) = M\mu.$$

Since our goal is to find primal variables u for which (3) holds with $\epsilon_f = 0$, we shall make the variable μ converge to zero as the Newton iterations progresses. This is done in the context of an interior-point method, meaning that all variables will be initialized so that the inequality constraints (2c) hold *strictly* and the progression along the Newton direction at each iteration will be selected so that these constraints are never violated.

The specific steps of the algorithm that follows are based on the primal-dual interior-point method for a single optimization, as described in [2]. To describe this algorithm, we re-write (4), (2b), and (5) as

$$\nabla_u L_f(u,\lambda,\nu) = \mathbf{0}_N, \qquad G(u) = \mathbf{0}_K, \qquad \lambda \odot F(u) = \mu \mathbf{1}_M, \tag{6a}$$

and (2c) as

$$\lambda \geqslant \mathbf{0}_M, \qquad F(u) \geqslant \mathbf{0}_M. \tag{6b}$$

Algorithm 1 (Primal-dual optimization).

Step 1. *Start with estimates u_0, λ_0, ν_0 that satisfy the inequalities $\lambda_0 > 0$, $F(u_0) > 0$ in (6b) and set $k = 0$. It is often a good idea to start with a value for u_0 that satisfies the equality constraint $G(u_0) = 0$, and*

$$\lambda_0 = \mu \mathbf{1}_M \oslash F(u_0),$$

which guarantees that we initially have $\lambda_0 \odot F(u_0) = \mu \mathbf{1}_M$.[2]

Step 2. *Linearize the equations in (6a) around a current estimate u_k, λ_k, ν_k, leading to*

$$\begin{bmatrix} \nabla_{uu} L_f(u_k,\lambda_k,\nu_k) & \nabla_{u\nu} L_f(u_k) & \nabla_{u\lambda} L_f(u_k) \\ \nabla_u G(u_k) & 0 & 0 \\ \operatorname{diag}(\lambda_k)\nabla_u F(u_k) & 0 & \operatorname{diag}[F(u_k)] \end{bmatrix} \begin{bmatrix} \Delta u \\ \Delta \nu \\ \Delta \lambda \end{bmatrix} = - \begin{bmatrix} \nabla_u L_f(u_k,\lambda_k,\nu_k) \\ G(u_k) \\ F(u_k) \odot \lambda_k - \mu \mathbf{1}_M \end{bmatrix}$$
$$\Leftrightarrow \quad \begin{bmatrix} \nabla_{uu} L_f(u_k,\lambda_k,\nu_k) & \nabla_u G(u_k)' & -\nabla_u F(u_k)' \\ \nabla_u G(u_k) & 0 & 0 \\ -\nabla_u F(u_k) & 0 & -\operatorname{diag}[F(u_k) \oslash \lambda_k] \end{bmatrix} \begin{bmatrix} \Delta u \\ \Delta \nu \\ \Delta \lambda \end{bmatrix} = - \begin{bmatrix} \nabla_u L_f(u_k,\lambda_k,\nu_k) \\ G(u_k) \\ -F(u_k) + \mu \mathbf{1}_M \oslash \lambda_k \end{bmatrix}, \tag{7}$$

where $\nabla_{uu} L_f$ denotes the Hessian matrix of L_f with respect to u. Since $F(u_k) > 0$ and $\lambda_k > 0$, we can solve this system of equations by first eliminating

$$\Delta\lambda = -\lambda_k - \operatorname{diag}[\lambda_k \oslash F(u_k)]\nabla_u F(u_k)\Delta u + \mu \mathbf{1}_M \oslash F(u_k), \tag{8a}$$

which leads to[3]

$$\begin{bmatrix} \nabla_{uu} L_f(u_k,\lambda_k,\nu_k) + \nabla_u F(u_k)' \operatorname{diag}[\lambda_k \oslash F(u_k)]\nabla_u F(u_k) & \nabla_u G(u_k)' \\ \nabla_u G(u_k) & 0 \end{bmatrix} \begin{bmatrix} \Delta u \\ \Delta \nu \end{bmatrix}$$

[2]João: Starting with $\nu_0 = 0$ seems to be okay. Note that the value of ν only affects the matrix in the left-hand side of (8c) when $G(u)$ is nonlinear.

[3]For numerical stability, one may solve instead

$$\begin{bmatrix} \nabla_{uu} L_f(u_k,\lambda_k,\nu_k) + \nabla_u F(u_k)' \operatorname{diag}[\lambda_k \oslash F(u_k)]\nabla_u F(u_k) + \epsilon I & \nabla_u G(u_k)' \\ \nabla_u G(u_k) & -\epsilon I \end{bmatrix} \begin{bmatrix} \Delta u \\ \Delta \nu \end{bmatrix}$$
$$= - \begin{bmatrix} \nabla_u f(u_k) + \nabla_u G(u_k)'\nu_k - \mu \nabla_u F(u_k)'(\mathbf{1}_M \oslash F(u_k)) \\ G(u_k) \end{bmatrix} \tag{8b}$$

for a small $\epsilon > 0$. The matrix in the left-hand side has the top submatrix positive definite and the bottom one negative definite and is therefore better conditioned (see [Saunders 1996, Sect 4.2,MattingleyBoyd]). Nevertheless, upon convergence we still get the right-hand side zero, which still gives us (6).

$$= -\begin{bmatrix} \nabla_u f(u_k) + \nabla_u G(u_k)' \nu_k - \mu \nabla_u F(u_k)' \big(\mathbf{1}_M \oslash F(u_k) \big) \\ G(u_k) \end{bmatrix}. \tag{8c}$$

Note however that starting by eliminating $\Delta\lambda$ may be undesirable when $\nabla_u F(u_k)$ has full rows, because in this case $\nabla_u F(u_k)' \operatorname{diag}[\lambda_k \oslash F(u_k)] \nabla_u F(u_k)$ will be a full matrix, which will "destroy" the computational advantages of any sparsity in $\nabla_{uu} L_f(u_k, \lambda_k, \nu_k)$. This is the case, e.g., when F has constraints that depend on every (or most) elements of u.

Step 3. *Find the* affine scaling direction $\left[\begin{smallmatrix} \Delta u_a' & \Delta \nu_a' & \Delta \lambda_a' \end{smallmatrix} \right]'$ *by solving* (8) *for $\mu = 0$:*

$$\begin{bmatrix} \nabla_{uu} L_f(u_k, \lambda_k, \nu_k) + \nabla_u F(u_k)' \operatorname{diag}[\lambda_k \oslash F(u_k)] \nabla_u F(u_k) & \nabla_u G(u_k)' \\ \nabla_u G(u_k) & 0 \end{bmatrix} \begin{bmatrix} \Delta u \\ \Delta \nu \end{bmatrix} = -\begin{bmatrix} \nabla_u f(u_k) + \nabla_u G(u_k)' \nu_k \\ G(u_k) \end{bmatrix}$$

$$\Delta\lambda = -\lambda_k - \operatorname{diag}[\lambda_k \oslash F(u_k)] \nabla_u F(u_k) \Delta u.$$

Step 4. *Select scalings so that the inequalities in* (6b) *would not be violated along the affine scaling direction:*

$$\alpha_a := \min\{\alpha_{\text{primal}}, \alpha_{\text{dual}}\},$$

where

$$\alpha_{\text{primal}} := \max\left\{ \alpha \in [0,1] : F(u_k + \alpha \Delta u_a) \geqslant 0 \right\}, \qquad \alpha_{\text{dual}} := \max\left\{ \alpha \in [0,1] : \lambda_k + \alpha \Delta \lambda_a \geqslant 0 \right\}$$

and define the following estimate for the "quality" of the affine scaling direction

$$\sigma := \left(\frac{F(u_k + \alpha_a \Delta u_a)'(\lambda_k + \alpha_a \Delta \lambda_a)}{F(u_k)' \lambda_k} \right)^{\delta},$$

where δ is a parameter typically selected equal to 2 or 3. Note that the numerator $F(u_k + \alpha_a \Delta u_a) \odot (\lambda_k + \alpha_a \Delta \lambda_a)$ is the value one would obtain for $\lambda \odot F(u)$ by moving purely along the affine scaling directions. A small value for σ thus indicates that a significant reduction in μ is possible.

Step 5. *Find the* search direction $\left[\begin{smallmatrix} \Delta u_s' & \Delta \nu_s' & \Delta \lambda_s' \end{smallmatrix} \right]'$ *by solving* (8) *for $\mu = \sigma \frac{F(u_k)' \lambda_k}{M}$*[4]*:*

$$\begin{bmatrix} \nabla_{uu} L_f(u_k, \lambda_k, \nu_k) & \nabla_u G(u_k)' & -\nabla_u F(u_k)' \\ \nabla_u G(u_k) & 0 & 0 \\ -\nabla_u F(u_k) & 0 & -\operatorname{diag}[F(u_k) \oslash \lambda_k] \end{bmatrix} \begin{bmatrix} \Delta u_s \\ \Delta \nu_s \\ \Delta \lambda_s \end{bmatrix} = -\begin{bmatrix} \nabla_u L_f(u_k, \lambda_k, \nu_k) \\ G(u_k) \\ -F(u_k) - (\nabla_u F(u_k) \Delta u_a) \odot \Delta \lambda_a \oslash \lambda_k + \mu \mathbf{1}_M \oslash \lambda_k \end{bmatrix},$$

where the (optional) blue term would come from a 2nd order expansion of the left-hand side of the last equality in (6a). [5] *Since $F(u_k) > 0$ and $\lambda_k > 0$, we can solve this system of equations by first eliminating*

$$\Delta\lambda_s = -\operatorname{diag}[\lambda_k \oslash F(u_k)] \nabla_u F(u_k) \Delta u_s - \lambda_k - (\nabla_u F(u_k) \Delta u_a) \odot \Delta \lambda_a \oslash F(u_k) + \mu \mathbf{1}_M \oslash F(u_k), \tag{9a}$$

[4] João: Check with code.

[5] João: Need reference or removal of the blue term.

which leads to

$$\begin{bmatrix} \nabla_{uu}L_f(u_k,\lambda_k,\nu_k)+\nabla_u F(u_k)'\operatorname{diag}[\lambda_k\oslash F(u_k)]\nabla_u F(u_k) & \nabla_u G(u_k)' \\ \nabla_u G(u_k) & 0 \end{bmatrix}\begin{bmatrix}\Delta u_s\\ \Delta\nu_s\end{bmatrix}$$
$$= -\begin{bmatrix} \nabla_u f(u_k)+\nabla_u G(u_k)'\nu_k+\nabla_u F(u_k)'((\nabla_u F(u_k)\Delta u_a)\odot\Delta\lambda_a\oslash F(u_k)) - \mu\nabla_u F(u_k)'\big(\mathbf{1}_M\oslash F(u_k)\big) \\ G(u_k)\end{bmatrix}. \tag{9b}$$

Step 6. *Update the estimates along the search direction so that the inequalities in (6b) hold strictly:*

$$u_{k+1}=u_k+\alpha_s\Delta u_s, \qquad \nu_{k+1}=\nu_k+\alpha_s\Delta\nu_s, \qquad \lambda_{k+1}=\lambda_k+\alpha_s\Delta\lambda_s$$

where

$$\alpha_s \coloneqq \min\{\alpha_{\text{primal}},\alpha_{\text{dual}}\},$$

and

$$\alpha_{\text{primal}} \coloneqq \max\Big\{\alpha\in[0,1]: F(u_k+\frac{\alpha}{.99}\Delta u_s)\geqslant 0\Big\}, \quad \alpha_{\text{dual}} \coloneqq \max\Big\{\alpha\in[0,1]: \lambda_k+\frac{\alpha}{.99}\Delta\lambda_s\geqslant 0\Big\}. \tag{10}$$

Step 7. *Repeat from 3 with an incremented value for k until*

$$\|\nabla_u L_f(u_k,\lambda_k,\nu_k)\|\leqslant\epsilon, \qquad \|G(u_k)\|\leqslant\epsilon_G, \qquad \lambda_k'F(u_k)\leqslant\epsilon_{\text{gap}}. \tag{11}$$

for sufficiently small tolerances ϵ, ϵ_G, ϵ_{gap}. □

When the function L_f that appear in the unconstrained minimizations in (2a) have a single stationary point that corresponds to their global minimum, termination of the Algorithm 1 guarantees that the assumptions of Lemma 1 hold [up to the tolerances in (11)] and we obtain the desired solution to (1).

The desired uniqueness of the stationary point holds, e.g., when the function $f(u)$ is convex in u, $F(u)$ is concave in u, and $G(u)$ is linear in u. However, in practice the Algorithm 1 can find solutions to (1) even when these convexity assumptions do not hold. For problems for which one cannot be sure whether the Algorithm 1 terminated at a global minimum of the unconstrained problem, one may run several instances of the algorithm with random initial conditions. Consistent results for the optimizations across multiple initializations will provide an indication that a global minimum has been found.

Remark 1 (Smoothness). Algorithm 1 requires all the functions f, F, G to be twice differentiable for the computation of the matrices that appear in (7). However, this does not preclude the use of this algorithm in many problems where these functions are not differentiable, because it is often possible to re-formulate non-smooth optimizations into smooth ones by appropriate transformation that often introduce additional optimization variables. Common examples of these transformations include the minimization of criteria involving ℓ_p norms, such as the "non-differentiable ℓ_1 optimization"

$$\min\big\{\|A_{m\times n}x-b\|_{\ell_1}+\cdots : x\in\mathbb{R}^n,\dots\big\}$$

which is equivalent to the following "smooth optimization"

$$\min\big\{v'\mathbf{1}_m+\cdots : x\in\mathbb{R}^n, v\in\mathbb{R}^m, -v\leqslant Ax-b\leqslant v,\dots\big\};$$

or the "non-differentiable ℓ_2 optimization"

$$\min\left\{\|A_{m\times n}x-b\|_{\ell_2}+\cdots : x\in\mathbb{R}^n,\dots\right\}$$

$$\min\left\{v+\cdots : x\in\mathbb{R}^n, v\geqslant 0, v^2\geqslant (Ax-b)'(Ax-b),\dots\right\}.$$

More examples of such transformations can be found, e.g., in [1].[6] $\square$

1.3 Sensitivity

Suppose now that the vector u of optimization variable can be decomposed as $u=(u_1,u_2)$, $u_1\in\mathbb{R}^{N_1}$, $u_2\in\mathbb{R}^{N_2}$, N_1+N_N and we want to determine how a perturbation in u_1 away from the optimal u_1^* would increase the value of (1), assuming that u_2 would adjust to still satisfy the constraints and minimize (1) for the possibly suboptimal value of u_1. Specifically, for a given value of u_1, let $u_2^\dagger(u_1)$ denote the value of u_2 that solves the optimization

$$f(u_1,u_2^\dagger(u_1))=\min_{(u_1,u_2)\in\mathcal{U}} f(u_1,u_2), \tag{12}$$

with

$$\mathcal{U}\coloneqq\{(u_1,u_2)\in\mathbb{R}^N : F(u_1,u_2)\geqslant 0,\ G(u_1,u_2)=0\}.$$

For a given minimum $u^*=(u_1^*,u_2^*)$ of (1), our goal is to determine

$$\frac{d^2 f(u_1,u_2^\dagger(u_1))}{du_1^2}\Big|_{u_1=u_1^*}.$$

Assuming that the solution (1) is the unique solution to the conditions of Lemma 1 with $\mu=0$:

$$\nabla_u L_f(u_1^*,u_2^*,\lambda^*,\nu^*)=\mathbf{0}_N,\qquad G(u_1^*,u_2^*)=\mathbf{0}_K,\qquad \lambda^*\odot F(u_1^*,u_2^*)=\mathbf{0}_M, \tag{13}$$

and the solution to (12) is also the unique solution to the corresponding conditions of Lemma 1 with $\mu=0$:

$$\nabla_{u_2} L_f(u_1,u_2^\dagger(u_1),\lambda^\dagger(u_1),\nu^\dagger(u_1))=\mathbf{0}_{N_2},\quad G(u_1,u_2^\dagger(u_1))=\mathbf{0}_K,\quad \lambda^\dagger\odot F(u_1,u_2^\dagger(u_1))=\mathbf{0}_M, \tag{14}$$

we must have

$$u_2^\dagger(u_1^*)=u_2^*,\qquad \lambda^\dagger(u_1)=\lambda^*,\qquad \nu^\dagger(u_1)=\nu^*.$$

Lemma 2. *The functions $u_2^\dagger(u_1)$, $\lambda^\dagger(u_1)$, $\nu^\dagger(u_1)$ defined implicitly by (14) satisfy:*

$$\begin{bmatrix}\nabla_{u_2u_2}L_f(u_1,u_2^\dagger,\lambda^\dagger,\nu^\dagger) & \nabla_{u_2}G(u_1,u_2^\dagger)' & -\nabla_{u_2}F(u_1,u_2^\dagger)'\\ \nabla_{u_2}G(u_1,u_2^\dagger) & 0 & 0\\ -\nabla_{u_2}F(u_1,u_2^\dagger) & 0 & -\operatorname{diag}[F(u_1,u_2^\dagger)\oslash\lambda^\dagger]\end{bmatrix}\begin{bmatrix}\frac{\partial u_2^\dagger}{\partial u_1}\\ \frac{\partial \nu^\dagger}{\partial u_1}\\ \frac{\partial \lambda^\dagger}{\partial u_1}\end{bmatrix}=-\begin{bmatrix}\nabla_{u_2u_1}L_f(u_1,u_2^\dagger,\lambda^\dagger,\nu^\dagger)\\ \nabla_{u_1}G(u_1,u_2^\dagger)\\ -\nabla_{u_1}F(u_1,u_2^\dagger)\end{bmatrix} \tag{15}$$

[6] João: We should dig for more references.

Moreover, for every value u_1 for which the set active inequality constraints does not change in a sufficiently small neighborhood of u_1, we have that

$$\frac{df(u_1, u_2^\dagger(u_1))}{du_1} = \nabla_{u_1} L_f(u_1, u_2^\dagger, \lambda^\dagger, \nu^\dagger) \tag{16}$$

$$\begin{aligned}\frac{d^2 f(u_1, u_2^\dagger(u_1))}{du_1^2} &= \nabla_{u_1 u_1} L_f(u_1, u_2^\dagger, \lambda^\dagger, \nu^\dagger) \\ &\quad + \begin{bmatrix} \nabla_{u_1 u_2} L_f(u_1, u_2^\dagger, \lambda^\dagger, \nu^\dagger) & \nabla_{u_1} G(u_1, u_2^\dagger)' & -\nabla_{u_1} F(u_1, u_2^\dagger) \end{bmatrix} \begin{bmatrix} \frac{\partial u_2^\dagger}{\partial u_1} \\ \frac{\partial \nu^\dagger}{\partial u_1} \\ \frac{\partial \lambda^\dagger}{\partial u_1} \end{bmatrix}.\end{aligned} \tag{17}$$

Equation 15 shows that all the partial derivatives $\frac{\partial u_2^\dagger}{\partial u_1}$, $\frac{\partial \nu^\dagger}{\partial u_1}$, $\frac{\partial \lambda^\dagger}{\partial u_1}$ can be obtained by solving a system of equations defined by a subset of the rows and columns of the matrix used in the computation of the Newton Step, which appears in (7).

Proof of Lemma 2. To prove 15, take derivatives of (14) with respect to u_1, to obtain

$$\begin{aligned}&\nabla_{u_2 u_1} L_f(u_1, u_2^\dagger, \lambda^\dagger, \nu^\dagger) + \nabla_{u_2 u_2} L_f(u_1, u_2^\dagger, \lambda^\dagger, \nu^\dagger) \frac{\partial u_2^\dagger}{\partial u_1} \\ &\qquad + \nabla_{u_2 \lambda} L_f(u_1, u_2^\dagger, \lambda^\dagger, \nu^\dagger) \frac{\partial \lambda^\dagger}{\partial u_1} + \nabla_{u_2 \nu} L_f(u_1, u_2^\dagger, \lambda^\dagger, \nu^\dagger) \frac{\partial \nu^\dagger}{\partial u_1} = \mathbf{0}_N,\end{aligned}$$

$$\nabla_{u_1} G(u_1, u_2^\dagger) + \nabla_{u_2} G(u_1, u_2^\dagger) \frac{\partial u_2^\dagger}{\partial u_1} = \mathbf{0}_K, \tag{18}$$

$$\operatorname{diag}[F(u_1, u_2^\dagger)] \frac{\partial \lambda^\dagger}{\partial u_1} + \operatorname{diag}[\lambda^\dagger] \nabla_{u_1} F(u_1, u_2^\dagger) + \operatorname{diag}[\lambda^\dagger] \nabla_{u_2} F(u_1, u_2^\dagger) \frac{\partial u_2^\dagger}{\partial u_1} = \mathbf{0}_M, \tag{19}$$

from which (15) follows by multiplying the bottom row by $\operatorname{diag}[\lambda^\dagger]^{-1}$.

To prove (16), we start by using the chain rule to compute

$$\frac{df(u_1, u_2^\dagger(u_1))}{du_1} = \nabla_{u_1} f(u_1, u_2^\dagger) + \nabla_{u_2} f(u_1, u_2^\dagger) \frac{\partial u_2^\dagger}{\partial u_1}. \tag{20}$$

To proceed, we note that the first equality in (14) is equivalent to

$$\nabla_{u_2} f(u_1, u_2^\dagger) - \lambda^{\dagger\prime} \nabla_{u_2} F(u_1, u_2^\dagger) + \nu^{\dagger\prime} \nabla_{u_2} G(u_1, u_2^\dagger) = \mathbf{0}_{N_2},$$

and therefore

$$\begin{aligned}\nabla_{u_2} f(u_1, u_2^\dagger) \frac{\partial u_2^\dagger}{\partial u_1} &= \lambda^{\dagger\prime} \nabla_{u_2} F(u_1, u_2^\dagger) \frac{\partial u_2^\dagger}{\partial u_1} - \nu^{\dagger\prime} \nabla_{u_2} G(u_1, u_2^\dagger) \frac{\partial u_2^\dagger}{\partial u_1} \\ &= -\lambda^{\dagger\prime} \Big(\operatorname{diag}[F(u_1, u_2^\dagger) \oslash \lambda^\dagger] \frac{\partial \lambda^\dagger}{\partial u_1} + \nabla_{u_1} F(u_1, u_2^\dagger) \Big) + \nu^\dagger \nabla_{u_1} G(u_1, u_2^\dagger) \\ &= -F(u_1, u_2^\dagger)' \frac{\partial \lambda^\dagger}{\partial u_1} - \lambda^{\dagger\prime} \nabla_{u_1} F(u_1, u_2^\dagger) + \nu^\dagger \nabla_{u_1} G(u_1, u_2^\dagger),\end{aligned}$$

where the second equality follows from (18)–(19). Using this in (20), we conclude that

$$\begin{aligned}\frac{df(u_1,u_2^\dagger(u_1))}{du_1} &= \nabla_{u_1} f(u_1,u_2^\dagger) - \lambda^{\dagger\prime}\nabla_{u_1}F(u_1,u_2^\dagger) + \nu^\dagger\nabla_{u_1}G(u_1,u_2^\dagger) - F(u_1,u_2^\dagger)'\frac{\partial\lambda^\dagger}{\partial u_1}\\ &= \nabla_{u_1} L_f(u_1,u_2^\dagger,\lambda^\dagger,\nu^\dagger) - F(u_1,u_2^\dagger)'\frac{\partial\lambda^\dagger}{\partial u_1}\\ &= \nabla_{u_1} L_f(u_1,u_2^\dagger,\lambda^\dagger,\nu^\dagger) - \sum_{i=1}^{M} F_i(u_1,u_2^\dagger)\frac{\partial\lambda_i^\dagger}{\partial u_1}.\end{aligned}$$

In case the set of active inequality constraints does not change areound u_1, each inequality $F_i(u_1,u_2^\dagger)$ either remains active (and therefore $F_i(u_1,u_2^\dagger)=0$) or inactive (and therefore $\lambda_i^\dagger(u_1)=0$. In either case all terms $F_i(u_1,u_2^\dagger)\frac{\partial\lambda_i^\dagger}{\partial u_1}$ are zero and (16) follows.

Finally, to obtain (17), we take second derivatives of (16) with respect to u_1, obtaining

$$\begin{aligned}\frac{d^2f(u_1,u_2^\dagger(u_1))}{du_1^2} &= \nabla_{u_1u_1}L_f(u_1,u_2^\dagger,\lambda^\dagger,\nu^\dagger) + \nabla_{u_1u_2}L_f(u_1,u_2^\dagger,\lambda^\dagger,\nu^\dagger)\frac{\partial u_2^\dagger}{\partial u_1}\\ &\quad + \nabla_{u_1\nu}L_f(u_1,u_2^\dagger,\lambda^\dagger,\nu^\dagger)\frac{\partial\nu^\dagger}{\partial u_1} + \nabla_{u_1\lambda}L_f(u_1,u_2^\dagger,\lambda^\dagger,\nu^\dagger)\frac{\partial\lambda^\dagger}{\partial u_1},\end{aligned}$$

from which (16) follows. ■

2 Interior-Point Method for Minimax Problems

Our goal is to find a pair $(u^*,d^*)\in\mathcal{U}[d^*]\times\mathcal{D}[u^*]$ that simultaneously solves the two coupled optimizations

$$f(u^*,d^*) = \min_{u\in\mathcal{U}[d^*]} f(u,d^*), \qquad g(u^*,d^*) = \min_{d\in\mathcal{D}[u^*]} g(u^*,d), \tag{21}$$

with

$$\begin{aligned}\mathcal{U}[d] &\coloneqq \{u\in\mathbb{R}^{N_u} : F_u(u,d)\geqslant 0,\ G_u(u,d)=0\},\\ \mathcal{D}[u] &\coloneqq \{d\in\mathbb{R}^{N_d} : F_d(u,d)\geqslant 0,\ G_d(u,d)=0\},\end{aligned}$$

for given functions $f:\mathbb{R}^{N_u}\times\mathbb{R}^{N_d}\in\mathbb{R}$, $g:\mathbb{R}^{N_u}\times\mathbb{R}^{N_d}\in\mathbb{R}$, $F_u:\mathbb{R}^{N_u}\times\mathbb{R}^{N_d}\to\mathbb{R}^{M_u}$, $F_d:\mathbb{R}^{N_u}\times\mathbb{R}^{N_d}\to\mathbb{R}^{M_d}$, $G_u:\mathbb{R}^{N_u}\times\mathbb{R}^{N_d}\to\mathbb{R}^{K_u}$, $G_d:\mathbb{R}^{N_u}\times\mathbb{R}^{N_d}\to\mathbb{R}^{K_d}$.

2.1 Primal-dual method

The following duality-like result provides the motivation for a primal-dual-like method to solve the coupled minimizations in (21). It provides a set of conditions, involving two unconstrained optimizations, that provide an approximation to the solution of (21).

Lemma 3 (Approximate equilibrium). *Suppose that we have found* primal variables $\hat{u}\in\mathbb{R}^{N_u}$, $\hat{d}\in\mathbb{R}^{N_d}$ *and* dual variables $\hat{\lambda}_{fu}\in\mathbb{R}^{M_u}$, $\hat{\lambda}_{gd}\in\mathbb{R}^{M_d}$, $\hat{\nu}_{fu}\in\mathbb{R}^{K_u}$, $\hat{\nu}_{gd}\in\mathbb{R}^{K_d}$ *that simultaneously satisfy all of the following conditions*

$$L_f(\hat{u},\hat{d},\hat{\lambda}_{fu},\hat{\nu}_{fu}) = \min_{u\in\mathbb{R}^{N_u}} L_f(u,\hat{d},\hat{\lambda}_{fu},\hat{\nu}_{fu}), \qquad L_g(\hat{u},\hat{d},\hat{\lambda}_{gd},\hat{\nu}_{gd}) = \min_{d\in\mathbb{R}^{N_d}} L_g(\hat{u},d,\hat{\lambda}_{gd},\hat{\nu}_{gd}) \tag{22a}$$

$$G_u(\hat{u}, \hat{d}) = 0, \qquad\qquad G_d(\hat{u}, \hat{d}) = 0, \tag{22b}$$

$$\hat{\lambda}_{fu} \geqslant 0, \qquad \hat{\lambda}_{gd} \geqslant 0, \qquad\qquad F_u(\hat{u}, \hat{d}) \geqslant 0, \qquad F_d(\hat{u}, \hat{d}) \geqslant 0, \tag{22c}$$

where

$$L_f(u, d, \lambda_{fu}, \nu_{fu}) \coloneqq f(u, d) - \lambda_{fu} \cdot F_u(u, d) + \nu_{fu} \cdot G_u(u, d),$$
$$L_g(u, d, \lambda_{gd}, \nu_{gd}) \coloneqq g(u, d) - \lambda_{gd} \cdot F_d(u, d) + \nu_{gd} \cdot G_d(u, d), \qquad \forall u, d, \lambda, \nu.$$

Then the pair $(\hat{u}, \hat{d})$ *approximately satisfies* (21) *in the sense that*

$$f(\hat{u}, \hat{d}) \leqslant \epsilon_f + \min_{u \in \mathcal{U}[\hat{d}]} f(u, \hat{d}), \qquad\qquad g(\hat{u}, \hat{d}) \leqslant \epsilon_g + \min_{d \in \mathcal{D}[\hat{u}]} g(\hat{u}, d), \tag{23}$$

with

$$\epsilon_f \coloneqq \hat{\lambda}_{fu} \cdot F_u(\hat{u}, \hat{d}), \qquad\qquad \epsilon_g \coloneqq \hat{\lambda}_{gd} \cdot F_d(\hat{u}, \hat{d}). \qquad \square$$

Proof of Lemma 3. The proof is a direct consequence of the following sequence of inequalities that start from the equalities in (22a):

$$\begin{aligned}
f(\hat{u}, \hat{d}) - \epsilon_f &= L_f(\hat{u}, \hat{d}, \hat{\lambda}_{fu}, \hat{\nu}_{fu}) - \hat{\nu}_{fu} \cdot G_u(\hat{u}, \hat{d}) = \min_{u \in \mathbb{R}^{N_u}} L_f(u, \hat{d}, \hat{\lambda}_{fu}, \hat{\nu}_{fu}) - 0 \\
&= \min_{u \in \mathbb{R}^{N_u}} f(u, \hat{d}) - \hat{\lambda}_{fu} \cdot F_u(u, \hat{d}) + \hat{\nu}_{fu} \cdot G_u(u, \hat{d}) \\
&\leqslant \max_{\lambda_{fu} \geqslant 0, \nu_{fu}} \min_{u \in \mathbb{R}^{N_u}} f(u, \hat{d}) - \lambda_{fu} \cdot F_u(u, \hat{d}) + \nu_{fu} \cdot G_u(u, \hat{d}) \\
&\leqslant \max_{\lambda_{fu} \geqslant 0, \nu_{fu}} \min_{u \in \mathcal{U}[\hat{d}]} f(u, \hat{d}) - \lambda_{fu} \cdot F_u(u, \hat{d}) + \nu_{fu} \cdot G_u(u, \hat{d}) \\
&= \min_{u \in \mathcal{U}[\hat{d}]} f(u, \hat{d}) \\
g(\hat{u}, \hat{d}) - \epsilon_g &= L_g(\hat{u}, \hat{d}, \hat{\lambda}_{gd}, \hat{\nu}_{gd}) - \hat{\nu}_{gd} \cdot G_d(\hat{u}, \hat{d}) = \min_{d \in \mathbb{R}^{N_d}} L_g(\hat{u}, d, \hat{\lambda}_{gd}, \hat{\nu}_{gd}) - 0 \\
&= \min_{d \in \mathbb{R}^{N_d}} g(\hat{u}, d) - \hat{\lambda}_{gd} \cdot F_d(\hat{u}, d) + \hat{\nu}_{gd} \cdot G_g(\hat{u}, d) \\
&\leqslant \max_{\lambda_{gd} \geqslant 0, \nu_{gd}} \min_{d \in \mathbb{R}^{N_d}} g(\hat{u}, d) - \lambda_{gd} \cdot F_d(\hat{u}, d) + \nu_{gd} \cdot G_g(\hat{u}, d) \\
&\leqslant \max_{\lambda_{gd} \geqslant 0, \nu_{gd}} \min_{d \in \mathcal{D}[\hat{u}]} g(\hat{u}, d) - \lambda_{gd} \cdot F_d(\hat{u}, d) + \nu_{gd} \cdot G_g(\hat{u}, d) \\
&= \min_{d \in \mathcal{D}[\hat{u}]} g(\hat{u}, d). \qquad \square
\end{aligned}$$

2.2 Interior-point primal-dual equilibria algorithm

The method proposed consists of using Newton iterations to solve a system of nonlinear equations on the primal variables $\hat{u} \in \mathbb{R}^{N_u}$, $\hat{d} \in \mathbb{R}^{N_d}$ and dual variables $\hat{\lambda}_{fu} \in \mathbb{R}^{M_u}$, $\hat{\lambda}_{gd} \in \mathbb{R}^{M_d}$, $\hat{\nu}_{fu} \in \mathbb{R}^{K_u}$, $\hat{\nu}_{gd} \in \mathbb{R}^{K_d}$ introduced in Lemma 3. The specific system of equations consists of:

1. the first-order optimality conditions for the unconstrained minimizations in (22a)[7]:

$$\nabla_u L_f(\hat{u}, \hat{d}, \hat{\lambda}_{fu}, \hat{\nu}_{fu}) = \mathbf{0}_{N_u}, \qquad\qquad \nabla_d L_g(\hat{u}, \hat{d}, \hat{\lambda}_{gd}, \hat{\nu}_{gd}) = \mathbf{0}_{N_d}; \tag{24}$$

[7]Given an integer M, we denote by $\mathbf{0}_M$ and by $\mathbf{1}_M$ the M-vectors with all entries equal to 0 and 1, respectively.

2. the equality conditions (22b); and

3. the equations[8]

$$F_u(\hat{u},\hat{d}) \odot \hat{\lambda}_{fu} = \mu \mathbf{1}_{M_u}, \qquad F_d(\hat{u},\hat{d}) \odot \hat{\lambda}_{gd} = \mu \mathbf{1}_{M_d}, \tag{25}$$

for some $\mu > 0$, which leads to

$$\epsilon_f = M_u \mu, \qquad \epsilon_g = M_d \mu.$$

Since our goal is to find primal variables $\hat{u}, \hat{d}$ for which (23) holds with $\epsilon_f = \epsilon_g = 0$, we shall make the variable μ converge to zero as the Newton iterations progress. This is done in the context of an interior-point method, meaning that all variables will be initialized so that the inequality constraints (22c) hold and the progression along the Newton direction at each iteration will be selected so that these constraints are never violated.

The specific steps of the algorithm that follows are inpired by the primal-dual interior-point method for a single optimization, as described in [2]. To describe this algorithm, we define

$$z \coloneqq \begin{bmatrix} \hat{u} \\ \hat{d} \end{bmatrix}, \qquad \lambda \coloneqq \begin{bmatrix} \hat{\lambda}_{fu} \\ \hat{\lambda}_{gd} \end{bmatrix}, \qquad \nu \coloneqq \begin{bmatrix} \hat{\nu}_{fu} \\ \hat{\nu}_{gd} \end{bmatrix}, \qquad G(z) \coloneqq \begin{bmatrix} G_u(\hat{u},\hat{d}) \\ G_d(\hat{u},\hat{d}) \end{bmatrix}, \qquad F(z) \coloneqq \begin{bmatrix} F_u(\hat{u},\hat{d}) \\ F_d(\hat{u},\hat{d}) \end{bmatrix},$$

which allow us to re-write (24), (22b), and (25) as

$$\nabla_u L_f(z,\lambda,\nu) = \mathbf{0}_{N_u}, \qquad \nabla_d L_g(z,\lambda,\nu) = \mathbf{0}_{N_d}, \qquad G(z) = \mathbf{0}_{K_u+K_d}, \qquad \lambda \odot F(z) = \mu \mathbf{1}_{M_u+M_d}, \tag{26a}$$

and (22c) as

$$\lambda \geqslant \mathbf{0}_{M_u+M_d}, \qquad F(z) \geqslant \mathbf{0}_{M_u+M_d}. \tag{26b}$$

Algorithm 2 (Primal-dual optimization).

Step 1. *Start with estimates z_0, λ_0, ν_0 that satisfy the inequalities $\lambda_0 \geqslant 0$, $F(z_0) \geqslant 0$ in (26b) and set $k = 0$.*

It is often a good idea to start with a value for z_0 that satisfies the equality constraint $G(z_0) = 0$, and

$$\lambda_0 = \mu \mathbf{1}_{M_u+M_d} \oslash F(z_0),$$

which guarantees that we initially have $\lambda_0 \odot F(u_0) = \mu \mathbf{1}_{M_u+M_d}$.[9]

Step 2. *Linearize the equations in (26a) around a current estimate z_k, λ_k, ν_k, leading to*

$$\begin{bmatrix} \nabla_{uz} L_f(z_k,\lambda_k,\nu_k) & \nabla_{u\nu} L_f(z_k) & \nabla_{u\lambda} L_f(z_k) \\ \nabla_{dz} L_g(z_k,\lambda_k,\nu_k) & \nabla_{d\nu} L_g(z_k) & \nabla_{d\lambda} L_g(z_k) \\ \nabla_z G(z_k) & 0 & 0 \\ -\nabla_z F(z_k) & 0 & -\operatorname{diag}[F(z_k) \oslash \lambda_k] \end{bmatrix} \begin{bmatrix} \Delta z \\ \Delta \nu \\ \Delta \lambda \end{bmatrix} = - \begin{bmatrix} \nabla_u L_f(z_k,\lambda_k,\nu_k) \\ \nabla_d L_g(z_k,\lambda_k,\nu_k) \\ G(z_k) \\ -F(z_k) + \mu \mathbf{1} \oslash \lambda_k \end{bmatrix}. \tag{27}$$

[8] Given two vectors $x, y \in \mathbb{R}^n$ we denote by $x \odot y \in \mathbb{R}^n$ and by $x \oslash y \in \mathbb{R}^n$ the entry-wise product and division of the two vectors, respectively.

[9] João: Starting with $\nu_0 = 0$ seems to be okay. Note that the value of ν only affects the matrix in the left-hand side of (28b) when $G(u)$ is nonlinear.

Since the vectors $F(z_k), \lambda_k$ have positive entries, we can solve this system of equations by first eliminating

$$\nabla_z F(z_k)\Delta z + \operatorname{diag}[F(z_k) \oslash \lambda_k]\Delta\lambda = -F(z_k) + \mu \mathbf{1} \oslash \lambda_k \quad \Leftrightarrow$$
$$\Delta\lambda = -\lambda_k - \operatorname{diag}[\lambda_k \oslash F(z_k)]\nabla_z F(z_k)\Delta z + \mu \mathbf{1} \oslash F(z_k) \tag{28a}$$

which leads to

$$\begin{bmatrix} \nabla_{uz} L_f(z_k, \lambda_k, \nu_k) & \nabla_{u\nu} L_f(z_k) \\ \nabla_{dz} L_g(z_k, \lambda_k, \nu_k) & \nabla_{d\nu} L_g(z_k) \\ \nabla_z G(z_k) & 0 \end{bmatrix} \begin{bmatrix} \Delta z \\ \Delta \nu \end{bmatrix} = - \begin{bmatrix} \nabla_u L_f(z_k, \lambda_k, \nu_k) + \nabla_{u\lambda} L_f(z_k)\Delta\lambda \\ \nabla_d L_g(z_k, \lambda_k, \nu_k) + \nabla_{d\lambda} L_g(z_k)\Delta\lambda \\ G(z_k) \end{bmatrix} \quad \Leftrightarrow$$
$$\begin{bmatrix} \nabla_{uz} L_f(z_k, \lambda_k, \nu_k) - \nabla_{u\lambda} L_f(z_k) \operatorname{diag}[\lambda_k \oslash F(z_k)]\nabla_z F(z_k) & \nabla_{u\nu} L_f(z_k) \\ \nabla_{dz} L_g(z_k, \lambda_k, \nu_k) - \nabla_{d\lambda} L_g(z_k) \operatorname{diag}[\lambda_k \oslash F(z_k)]\nabla_z F(z_k) & \nabla_{d\nu} L_g(z_k) \\ \nabla_z G(z_k) & 0 \end{bmatrix} \begin{bmatrix} \Delta z \\ \Delta \nu \end{bmatrix}$$
$$= - \begin{bmatrix} \nabla_u L_f(z_k, \lambda_k, \nu_k) - \nabla_{u\lambda} L_f(z_k)\lambda_k + \mu \nabla_{u\lambda} L_f(z_k)\big(\mathbf{1} \oslash F(z_k)\big) \\ \nabla_d L_g(z_k, \lambda_k, \nu_k) - \nabla_{d\lambda} L_g(z_k)\lambda_k + \mu \nabla_{d\lambda} L_g(z_k)\big(\mathbf{1} \oslash F(z_k)\big) \\ G(z_k) \end{bmatrix} \quad \Leftrightarrow$$
$$\begin{bmatrix} \nabla_{uz} L_f(z_k, \lambda_k, \nu_k) - \nabla_{u\lambda} L_f(z_k) \operatorname{diag}[\lambda_k \oslash F(z_k)]\nabla_z F(z_k) & \nabla_{u\nu} L_f(z_k) \\ \nabla_{dz} L_g(z_k, \lambda_k, \nu_k) - \nabla_{d\lambda} L_g(z_k) \operatorname{diag}[\lambda_k \oslash F(z_k)]\nabla_z F(z_k) & \nabla_{d\nu} L_g(z_k) \\ \nabla_z G(z_k) & 0 \end{bmatrix} \begin{bmatrix} \Delta z \\ \Delta \nu \end{bmatrix}$$
$$= - \begin{bmatrix} \nabla_u f(z_k) + \nabla_u\big((\nu_{fu})_k G_u(z_k)\big) + \mu \nabla_{u\lambda} L_f(z_k)\big(\mathbf{1} \oslash F(z_k)\big) \\ \nabla_d g(z_k) + \nabla_d\big((\nu_{gd})_k G_d(z_k)\big) + \mu \nabla_{d\lambda} L_g(z_k)\big(\mathbf{1} \oslash F(z_k)\big) \\ G(z_k) \end{bmatrix} \tag{28b}$$

Step 3. *Find the* affine scaling direction $\left[\begin{smallmatrix} \Delta z_a' & \Delta \nu_a' & \Delta\lambda_a' \end{smallmatrix}\right]'$ *by solving* (28) *for* $\mu = 0$*:*

$$\begin{bmatrix} \nabla_{uz} L_f(z_k, \lambda_k, \nu_k) - \nabla_{u\lambda} L_f(z_k) \operatorname{diag}[\lambda_k \oslash F(z_k)]\nabla_z F(z_k) & \nabla_{u\nu} L_f(z_k) \\ \nabla_{dz} L_g(z_k, \lambda_k, \nu_k) - \nabla_{d\lambda} L_g(z_k) \operatorname{diag}[\lambda_k \oslash F(z_k)]\nabla_z F(z_k) & \nabla_{d\nu} L_g(z_k) \\ \nabla_z G(z_k) & 0 \end{bmatrix} \begin{bmatrix} \Delta z_a \\ \Delta \nu_a \end{bmatrix}$$
$$= - \begin{bmatrix} \nabla_u f(z_k) + \nabla_u\big((\nu_{fu})_k G_u(z_k)\big) \\ \nabla_d g(z_k) + \nabla_d\big((\nu_{gd})_k G_d(z_k)\big) \\ G(z_k) \end{bmatrix}$$
$$\Delta\lambda_a = -\lambda_k - \operatorname{diag}[\lambda_k \oslash F(z_k)]\nabla_z F(z_k)\Delta z_a.$$

Step 4. *Select scalings so that the inequalities in* (26b) *would not be violated along the affine scaling direction:*

$$\alpha_a \coloneqq \min\{\alpha_{\text{primal}}, \alpha_{\text{dual}}\},$$

where

$$\alpha_{\text{primal}} \coloneqq \max\left\{\alpha \in [0, 1] : F(z_k + \alpha\Delta z_a) \geqslant 0\right\}, \qquad \alpha_{\text{dual}} \coloneqq \max\left\{\alpha \in [0, 1] : \lambda_k + \alpha\Delta\lambda_a \geqslant 0\right\} \tag{29}$$

and define the following estimate for the "quality" of the affine scaling direction

$$\sigma \coloneqq \left(\frac{F(z_k + \alpha_{\text{primal}}\Delta z_a)'(\lambda_k + \alpha_{\text{dual}}\Delta\lambda_a)}{F(z_k)'\lambda_k}\right)^{\delta},$$

where δ is a parameter typically selected equal to 2 or 3. Note that the numerator $F(z_k+\alpha_a\Delta z_a)'(\lambda_k+\alpha_a\Delta\lambda_a)$ is the value one would obtain for $\lambda' F(z)$ by moving purely along the affine scaling directions. A small value for σ thus indicates that a significant reduction in μ is possible.

Step 5. *Find the* search direction $\left[\,\Delta z_s' \; \Delta \nu_s' \; \Delta\lambda_s'\,\right]'$ *by solving (28) for* $\mu = \sigma \frac{F(z_k)\odot\lambda_k}{M_u+M_d}$*:*

$$\begin{bmatrix} \nabla_{uz} L_f(z_k,\lambda_k,\nu_k) & \nabla_{u\nu} L_f(z_k) & \nabla_{u\lambda} L_f(z_k) \\ \nabla_{dz} L_g(z_k,\lambda_k,\nu_k) & \nabla_{d\nu} L_g(z_k) & \nabla_{d\lambda} L_g(z_k) \\ \nabla_z G(z_k) & 0 & 0 \\ -\nabla_z F(z_k) & 0 & -\operatorname{diag}[F(z_k)\oslash\lambda_k] \end{bmatrix} \begin{bmatrix} \Delta z_s \\ \Delta \nu_s \\ \Delta\lambda_s \end{bmatrix} = -\begin{bmatrix} \nabla_u L_f(z_k,\lambda_k,\nu_k) \\ \nabla_d L_g(z_k,\lambda_k,\nu_k) \\ G(z_k) \\ -F(z_k) - \color{blue}{(\nabla_z F(z_k)\Delta z_a)\odot\Delta\lambda_a\oslash\lambda_k} + \mu\mathbf{1}\oslash\lambda_k \end{bmatrix},$$

where the (optional) blue term would come from a 2nd order expansion of the left-hand side of the last equality in (26a). [10] *Since the vectors $F(z_k),\lambda_k$ have positive entries, we can solve this system of equations by first eliminating*

$$\begin{aligned} \nabla_z F(z_k)\Delta z_s + \operatorname{diag}[F(z_k)\oslash\lambda_k]\Delta\lambda_s = -F(z_k) - \color{blue}{(\nabla_z F(z_k)\Delta z_a)\odot\Delta\lambda_a\oslash\lambda_k} + \mu\mathbf{1}\oslash\lambda_k \quad \Leftrightarrow \\ \Delta\lambda_s = -\lambda_k - \color{blue}{(\nabla_z F(z_k)\Delta z_a)\odot\Delta\lambda_a\oslash F(z_k)} - \operatorname{diag}[\lambda_k\oslash F(z_k)]\nabla_z F(z_k)\Delta z_s + \mu\mathbf{1}\oslash F(z_k) \end{aligned}$$

which leads to

$$\begin{bmatrix} \nabla_{uz} L_f(z_k,\lambda_k,\nu_k) & \nabla_{u\nu} L_f(z_k) \\ \nabla_{dz} L_g(z_k,\lambda_k,\nu_k) & \nabla_{d\nu} L_g(z_k) \\ \nabla_z G(z_k) & 0 \end{bmatrix} \begin{bmatrix} \Delta z_s \\ \Delta\nu_s \end{bmatrix} = -\begin{bmatrix} \nabla_u L_f(z_k,\lambda_k,\nu_k) + \nabla_{u\lambda} L_f(z_k)\Delta\lambda_s \\ \nabla_d L_g(z_k,\lambda_k,\nu_k) + \nabla_{d\lambda} L_g(z_k)\Delta\lambda_s \\ G(z_k) \end{bmatrix} \quad \Leftrightarrow$$

$$\begin{bmatrix} \nabla_{uz} L_f(z_k,\lambda_k,\nu_k) - \nabla_{u\lambda} L_f(z_k)\operatorname{diag}[\lambda_k\oslash F(z_k)]\nabla_z F(z_k) & \nabla_{u\nu} L_f(z_k) \\ \nabla_{dz} L_g(z_k,\lambda_k,\nu_k) - \nabla_{d\lambda} L_g(z_k)\operatorname{diag}[\lambda_k\oslash F(z_k)]\nabla_z F(z_k) & \nabla_{d\nu} L_g(z_k) \\ \nabla_z G(z_k) & 0 \end{bmatrix} \begin{bmatrix} \Delta z_s \\ \Delta\nu_s \end{bmatrix}$$
$$= -\begin{bmatrix} \nabla_u f(z_k) + \nabla_u\big((\nu_{fu})_k G_u(z_k)\big) - \color{blue}{\nabla_{u\lambda} L_f(z_k)\big((\nabla_z F(z_k)\Delta z_a)\odot\Delta\lambda_a\oslash F(z_k)\big)} + \mu\nabla_{u\lambda} L_f(z_k)\big(\mathbf{1}\oslash F(z_k)\big) \\ \nabla_d g(z_k) + \nabla_d\big((\nu_{gd})_k G_d(z_k)\big) - \color{blue}{\nabla_{d\lambda} L_g(z_k)\big((\nabla_z F(z_k)\Delta z_a)\odot\Delta\lambda_a\oslash F(z_k)\big)} + \mu\nabla_{d\lambda} L_g(z_k)\big(\mathbf{1}\oslash F(z_k)\big) \\ G(z_k) \end{bmatrix}.$$

Step 6. *Update the estimates along the search direction so that the inequalities in (26b) hold strictly:*

$$z_{k+1} = z_k + \alpha_s\Delta z_s, \qquad \nu_{k+1} = \nu_k + \alpha_s\Delta\nu_s, \qquad \lambda_{k+1} = \lambda_k + \alpha_s\Delta\lambda_s$$

where

$$\alpha_s \coloneqq \min\{\alpha_{\text{primal}},\alpha_{\text{dual}}\},$$

and

$$\alpha_{\text{primal}} \coloneqq \max\Big\{\alpha\in[0,1] : F(z_k + \frac{\alpha}{.99}\Delta z_s) \geqslant 0\Big\}, \quad \alpha_{\text{dual}} \coloneqq \max\Big\{\alpha\in[0,1] : \lambda_k + \frac{\alpha}{.99}\Delta\lambda_s \geqslant 0\Big\}. \tag{30}$$

[10] João: Need reference or removal of the blue term.

Step 7. *Repeat from 3 with an incremented value for k until*

$$\|\nabla_u L_f(z_k, \lambda_k, \nu_k)\| \leqslant \epsilon_u, \qquad \|\nabla_d L_g(z_k, \lambda_k, \nu_k)\| \leqslant \epsilon_d, \qquad \|G(z_k)\| \leqslant \epsilon_G, \qquad \lambda_k' F(z_k) \leqslant \epsilon_{\text{gap}}. \tag{31}$$

for sufficiently small tolerances ϵ_u, ϵ_d, ϵ_G, ϵ_{gap}. □

When the functions L_f and L_g that appear in the unconstrained minimizations in (22a) have a single stationary point that corresponds to their global minimum, termination of the Algorithm 2 guarantees that the assumptions of Lemma 3 hold [up to the tolerances in (31)] and we obtain the desired solution to (21).

The desired uniqueness of the stationary point holds, e.g., when the function $f(u, d)$ is convex in u, $g(u, d)$ is convex in d, $F_u(u, d)$ is concave in u, $F_d(u, d)$ is concave in d, and $G_u(u, d)$ is linear in u, and $G_d(u, d)$ is linear in d. However, in practice the Algorithm 2 can find solutions to (21) even when these convexity assumptions do not hold. For problems for which one cannot be sure whether the Algorithm 2 terminated at a global minimum of the unconstrained problem, one may run several instances of the algorithm with random initial conditions. Consistent results for the optimizations across multiple initializations will provide an indication that a global minimum has been found.

Remark 2 (Smoothness). Algorithm 2 requires all the functions f, g, F_u, F_d, G_u, G_d to be twice differentiable for the computation of the matrices that appear in (27). However, this does not preclude the use of this algorithm in many problems where these functions are not differentiable because it is often possible to re-formulate non-smooth optimizations into smooth ones by appropriate transformations that often introduce additional optimization variables. Common examples of these transformations include the minimization of criteria involving ℓ_p norms, such as the "non-differentiable ℓ_1 optimization"

$$\min\left\{\|A_{m\times n}x - b\|_{\ell_1} + \cdots : x \in \mathbb{R}^n, \dots\right\}$$

which is equivalent to the following "smooth optimization"

$$\min\left\{v'\mathbf{1}_m + \cdots : x \in \mathbb{R}^n, v \in \mathbb{R}^m, -v \leqslant Ax - b \leqslant v, \dots\right\};$$

or the "non-differentiable ℓ_2 optimization"

$$\min\left\{\|A_{m\times n}x - b\|_{\ell_2} + \cdots : x \in \mathbb{R}^n, \dots\right\}$$

$$\min\left\{v + \cdots : x \in \mathbb{R}^n, v \geqslant 0, v^2 \geqslant (Ax - b)'(Ax - b), \dots\right\}.$$

More examples of such transformations can be found, e.g., in [1]. □

3 Interior-Point Method for Minimax Problems with a Common Latent Variables

Like in Section 2, our goal is to find a pair $(u^*, d^*) \in \mathcal{U}[d^*] \times \mathcal{D}[u^*]$ that simultaneously solves the two coupled optimizations

$$\bar{f}(u^*, d^*) = \min_{u \in \bar{\mathcal{U}}[d^*]} \bar{f}(u, d^*), \qquad \bar{g}(u^*, d^*) = \min_{d \in \bar{\mathcal{D}}[u^*]} \bar{g}(u^*, d), \tag{32}$$

with

$$\bar{\mathcal{U}}[d] := \{u \in \mathbb{R}^{N_u} : \bar{F}_u(u, d) \geqslant 0,\ \bar{G}_u(u, d) = 0\}, \tag{33a}$$

$$\bar{\mathcal{D}}[u] \coloneqq \{d \in \mathbb{R}^{N_d} : \bar{F}_d(u,d) \geqslant 0,\ \bar{G}_d(u,d) = 0\}, \tag{33b}$$

for given functions $\bar{f} : \mathbb{R}^{N_u} \times \mathbb{R}^{N_d} \in \mathbb{R}$, $\bar{g} : \mathbb{R}^{N_u} \times \mathbb{R}^{N_d} \in \mathbb{R}$, $\bar{F}_u : \mathbb{R}^{N_u} \times \mathbb{R}^{N_d} \to \mathbb{R}^{M_u}$, $\bar{F}_d : \mathbb{R}^{N_u} \times \mathbb{R}^{N_d} \to \mathbb{R}^{M_d}$, $\bar{G}_u : \mathbb{R}^{N_u} \times \mathbb{R}^{N_d} \to \mathbb{R}^{K_u}$,q $\bar{G}_d : \mathbb{R}^{N_u} \times \mathbb{R}^{N_d} \to \mathbb{R}^{K_d}$. However, we are now interested in cases where these functions can be expressed in terms of common *latent variables*. Specifically, these functions can be expressed as

$$\begin{aligned}
\bar{f}(u,d) &= f\big(u,d,\chi(u,d)\big), & \bar{g}(u,d) &= g\big(u,d,\chi(u,d)\big),\\
\bar{F}_u(u,d) &= F_u\big(u,d,\chi(u,d)\big), & \bar{G}_u(u,d) &= G_u\big(u,d,\chi(u,d)\big),\\
\bar{F}_d(u,d) &= F_d\big(u,d,\chi(u,d)\big), & \bar{G}_d(u,d) &= G_d\big(u,d,\chi(u,d)\big),
\end{aligned}$$

$\forall u \in \mathbb{R}^{N_u}, d \in \mathbb{R}^{N_d}$, for a function $\chi : \mathbb{R}^{N_u} \times \mathbb{R}^{N_d} \to \mathbb{R}^{N_x}$ whose value $\chi(u,d)$ is defined implicitly by a function[11] $H : \mathbb{R}^{N_u} \times \mathbb{R}^{N_d} \times \mathbb{R}^{N_x} \to \mathbb{R}^{K_x}$ and an equation of the form

$$H(u,d,x) = 0. \tag{34}$$

The function H is assumed to be such that (34) has a unique solution x for every $u \in \mathbb{R}^{N_u}$, $d \in \mathbb{R}^{N_d}$. The followin corollary of Lemma 3 is useful in situations where it is difficult (or impossible) to find an explicit form for χ.

3.1 Primal-dual method

The following duality-like result provides the motivation for a primal-dual-like method to solve the coupled minimizations in (32). It provides a set of conditions, involving two unconstrained optimizations, that provide an approximation to the solution of (32). It improves upon a direct application of Lemma 3 to (32), in the implictely defined function χ does not appear in the conditions.

Corollary 1 (Approximate equilibrium). *Consider the coupled optimizations in (32) and assume that for every $u \in \mathbb{R}^{N_u}$, $d \in \mathbb{R}^{N_d}$, the equation (34) has a unique solution x. Suppose that we have found* primal variables $\hat{u} \in \mathbb{R}^{N_u}, \hat{d} \in \mathbb{R}^{N_d}, \hat{x} \in \mathbb{R}^{N_x}$ *and* dual variables $\hat{\nu}_{fu} \in \mathbb{R}^{K_u}, \hat{\nu}_{fx} \in \mathbb{R}^{K_x}, \hat{\nu}_{gd} \in \mathbb{R}^{K_d}, \hat{\nu}_{gx} \in \mathbb{R}^{K_x}$ *that simultaneously satisfy all of the following conditions*

$$G_u(\hat{u},\hat{d},\hat{x}) = 0, \qquad G_d(\hat{u},\hat{d},\hat{x}) = 0, \qquad H(\hat{u},\hat{d},\hat{x}) = 0, \tag{35a}$$

$$\hat{\lambda}_{fu} \geqslant 0, \qquad \hat{\lambda}_{gd} \geqslant 0, \qquad F_u(\hat{u},\hat{d},\hat{x}) \geqslant 0, \qquad F_d(\hat{u},\hat{d},\hat{x}) \geqslant 0, \tag{35b}$$

$$L_f(\hat{u},\hat{d},\hat{x},\hat{\lambda}_{fu},\hat{\nu}_{fu},\hat{\nu}_{fx}) = \min_{u\in\mathbb{R}^{N_u}, x\in\mathbb{R}^{N_x}} L_f(u,\hat{d},x,\hat{\lambda}_{fu},\hat{\nu}_{fu},\hat{\nu}_{fx}), \tag{35c}$$

$$L_g(\hat{u},\hat{d},\hat{x},\hat{\lambda}_{gd},\hat{\nu}_{gd},\hat{\nu}_{gx}) = \min_{d\in\mathbb{R}^{N_d}, x\in\mathbb{R}^{N_x}} L_g(\hat{u},d,x,\hat{\lambda}_{gd},\hat{\nu}_{gd},\hat{\nu}_{gx}) \tag{35d}$$

where

$$\begin{aligned}
L_f(u,d,x,\lambda_{fu},\nu_{fu},\nu_{fx}) &\coloneqq f(u,d,x) - \lambda_{fu}\cdot F_u(u,d,x) + \nu_{fu}\cdot G_u(u,d,x) + \nu_{fx}\cdot H(u,d,x),\\
L_g(u,d,x,\lambda_{gd},\nu_{gd},\nu_{gx}) &\coloneqq g(u,d,x) - \lambda_{gd}\cdot F_d(u,d,x) + \nu_{gd}\cdot G_d(u,d,x) + \nu_{gx}\cdot H(u,d,x).
\end{aligned}$$

Then $(\hat{u},\hat{d})$ approximately satisfy (32) in the sense that

$$\bar{f}(\hat{u},\hat{d}) \leqslant \epsilon_f + \min_{u\in\bar{\mathcal{U}}[\hat{d}]} \bar{f}(u,\hat{d}), \qquad \bar{g}(\hat{u},\hat{d}) \leqslant \epsilon_g + \min_{d\in\bar{\mathcal{D}}[\hat{u}]} \bar{g}(\hat{u},d), \tag{36}$$

[11] João: To eventually get square system of equations, we should have $K_x = N_x$.

with

$$\epsilon_f := \hat{\lambda}_{fu} \cdot F_u(\hat{u}, \hat{d}, \hat{x}), \qquad \epsilon_g := \hat{\lambda}_{gd} \cdot F_d(\hat{u}, \hat{d}, \hat{x}). \qquad \square$$

Note that while Corollary 1 utilizes a single primal (latent) variable $\hat{x}$, it requires two dual variables $\hat{\nu}_{fx}, \hat{\nu}_{gx} \in \mathbb{R}^{K_x}$ associated with the equality constraint $H(\hat{u}, \hat{d}, \hat{x}) = 0$.

Proof of Corollary 1. Since the equation in (34) has a unique solution in x, the optimizations in (32), can be re-written as

$$\bar{f}(u^*, d^*) = \min_{(u,x)\in\mathcal{U}[d^*]} f(u, d^*, x), \qquad \bar{g}(u^*, d^*) = \min_{(d,z)\in\mathcal{D}[u^*]} g(u^*, d, z), \tag{37}$$

with

$$\mathcal{U}[d] := \{(u, x) \in \mathbb{R}^{N_u} \times \mathbb{R}^{N_x} : F_u(u, d, x) \geqslant 0,\ G_u(u, d, x) = 0,\ H(u, d, x) = 0\}, \tag{38}$$

$$\mathcal{D}[u] := \{(d, z) \in \mathbb{R}^{N_d} \times \mathbb{R}^{N_x} : F_d(u, d, z) \geqslant 0,\ G_d(u, d, z) = 0,\ H(u, d, z) = 0\}, \tag{39}$$

which is again of the form considered in Section 2, but for optimization variables (u, x) and (d, z) in higher dimensional spaces.

Applying Lemma 3 to the new formulation in (37), we conclude that if we find *primal variables* $\hat{u} \in \mathbb{R}^N, \hat{x} \in \mathbb{R}^N, \hat{d} \in \mathbb{R}^{N_d}, \hat{z} \in \mathbb{R}^{N_x}$ and *dual variables* $\hat{\lambda}_{fu} \in \mathbb{R}^{M_u}, \hat{\lambda}_{gd} \in \mathbb{R}^{M_d}$, $\hat{\nu}_{fu} \in \mathbb{R}^{K_u}, \hat{\nu}_{fx} \in \mathbb{R}^{K_x}, \hat{\nu}_{gd} \in \mathbb{R}^{K_d}, \hat{\nu}_{gx} \in \mathbb{R}^{K_x}$ that simultaneously satisfy all of the following conditions

$$G_u(\hat{u}, \hat{d}, \hat{x}) = 0, \qquad H(\hat{u}, \hat{d}, \hat{x}) = 0, \qquad G_d(\hat{u}, \hat{d}, \hat{z}) = 0, \qquad H(\hat{u}, \hat{d}, \hat{z}) = 0, \tag{40a}$$

$$\hat{\lambda}_{fu} \geqslant 0, \qquad \hat{\lambda}_{gd} \geqslant 0, \qquad F_u(\hat{u}, \hat{d}, \hat{x}) \geqslant 0, \qquad F_d(\hat{u}, \hat{d}, \hat{z}) \geqslant 0, \tag{40b}$$

$$L_f(\hat{u}, \hat{x}, \hat{d}, \hat{z}, \hat{\lambda}_{fu}, \hat{\nu}_{fu}, \hat{\nu}_{fx}) = \min_{u\in\mathbb{R}^{N_u}, x\in\mathbb{R}^{N_x}} L_f(u, x, \hat{d}, \hat{z}, \hat{\lambda}_{fu}, \hat{\nu}_{fu}, \hat{\nu}_{fx}), \tag{40c}$$

$$L_g(\hat{u}, \hat{x}, \hat{d}, \hat{z}, \hat{\lambda}_{gd}, \hat{\nu}_{gd}, \hat{\nu}_{gx}) = \min_{d\in\mathbb{R}^{N_d}, z\in\mathbb{R}^{N_x}} L_g(\hat{u}, \hat{x}, d, z, \hat{\lambda}_{gd}, \hat{\nu}_{gd}, \hat{\nu}_{gx}) \tag{40d}$$

where

$$L_f(u, x, d, z, \lambda_{fu}, \nu_{fu}, \nu_{fx}) := f(u, d, x) - \lambda_{fu} \cdot F_u(u, d, x) + \nu_{fu} \cdot G_u(u, d, x) + \nu_{fx} \cdot H(u, d, x),$$
$$L_g(u, x, d, z, \lambda_{gd}, \nu_{gd}, \nu_{gx}) := g(u, d, z) - \lambda_{gd} \cdot F_d(u, d, z) + \nu_{gd} \cdot G_d(u, d, z) + \nu_{gx} \cdot H(u, d, z);$$

then

$$\bar{f}(\hat{u}, \hat{d}) = f(\hat{u}, \hat{d}, \hat{x}) \leqslant \epsilon_f + \min_{(u,x)\in\mathcal{U}[\hat{d}]} f(u, \hat{d}, x) = \epsilon_f + \min_{u\in\bar{\mathcal{U}}[\hat{d}]} \bar{f}(u, \hat{d}),$$
$$\bar{g}(\hat{u}, \hat{d}) = g(\hat{u}, \hat{d}, \hat{z}) \leqslant \epsilon_g + \min_{(d,z)\in\mathcal{D}[\hat{u}]} g(\hat{u}, d, z) = \epsilon_g + \min_{d\in\bar{\mathcal{D}}[\hat{u}]} \bar{g}(\hat{u}, d)$$

with

$$\epsilon_f := \hat{\lambda}_{fu} \cdot F_u(\hat{u}, \hat{d}, \hat{x}), \qquad \epsilon_g := \hat{\lambda}_{gd} \cdot F_d(\hat{u}, \hat{d}, \hat{z}). \qquad \square$$

The result follows from this, together with the observation that $\hat{x} = \hat{z}$ because the equations $H(\hat{u}, \hat{d}, \hat{x}) = 0$ and $H(\hat{u}, \hat{d}, \hat{z}) = 0$ in (40a) must have exactly the same solution $\hat{x} = \hat{z}$. ■

3.2 Interior-point primal-dual equilibria algorithm

The method proposed consists of using Newton iterations to solve a system of nonlinear equations on the primal variables $\hat{u} \in \mathbb{R}^{N_u}, \hat{d} \in \mathbb{R}^{N_d}, \hat{x} \in \mathbb{R}^{N_x}$ and dual variables $\hat{\lambda}_{fu} \in \mathbb{R}^{M_u}, \hat{\lambda}_{gd} \in \mathbb{R}^{M_d}$, $\hat{\nu}_{fu} \in \mathbb{R}^{K_u}, \hat{\nu}_{gd} \in \mathbb{R}^{K_d}, \hat{\nu}_{fx}, \hat{\nu}_{gx} \in \mathbb{R}^{K_x}$ introduced in Corollary 1. The specific system of equations consists of:

1. the first-order optimality conditions for the unconstrained minimizations in (35c)–(35d)[12]:

$$\nabla_u L_f(\hat{u}, \hat{d}, \hat{x}, \hat{\lambda}_{fu}, \hat{\nu}_{fu}, \hat{\nu}_{fx}) = \mathbf{0}_{N_u}, \qquad \nabla_x L_f(\hat{u}, \hat{d}, \hat{x}, \hat{\lambda}_{fu}, \hat{\nu}_{fu}, \hat{\nu}_{fx}) = \mathbf{0}_{N_u}, \tag{41a}$$

$$\nabla_d L_g(\hat{u}, \hat{d}, \hat{x}, \hat{\lambda}_{gd}, \hat{\nu}_{gd}, \hat{\nu}_{gx}) = \mathbf{0}_{N_d}, \qquad \nabla_x L_g(\hat{u}, \hat{d}, \hat{x}, \hat{\lambda}_{gd}, \hat{\nu}_{gd}, \hat{\nu}_{gx}) = \mathbf{0}_{N_d}; \tag{41b}$$

2. the equality conditions (35a); and

3. the equations[13]

$$F_u(\hat{u}, \hat{d}, \hat{x}) \odot \hat{\lambda}_{fu} = \mu \mathbf{1}_{M_u}, \qquad F_d(\hat{u}, \hat{d}, \hat{x}) \odot \hat{\lambda}_{gd} = \mu \mathbf{1}_{M_d}, \tag{42}$$

   for some $\mu > 0$, which leads to

$$\epsilon_f = M_u \mu, \qquad \epsilon_g = M_d \mu.$$

Since our goal is to find primal variables $\hat{u}, \hat{d}, \hat{x}$ for which (36) holds with $\epsilon_f = \epsilon_g = 0$, we shall make the variable μ converge to zero as the Newton iterations progress. This is done in the context of an interior-point method, meaning that all variables will be initialized so that the inequality constraints (40b) hold and the progression along the Newton direction at each iteration will be selected so that these constraints are never violated.

The specific steps of the algorithm that follows are inspired by the primal-dual interior-point method for a single optimization, as described in [2]. To describe this algorithm, we define

$$z := \begin{bmatrix} \hat{u} \\ \hat{d} \\ \hat{x} \end{bmatrix}, \quad \lambda := \begin{bmatrix} \hat{\lambda}_{fu} \\ \hat{\lambda}_{gd} \end{bmatrix}, \quad \nu := \begin{bmatrix} \hat{\nu}_{fu} \\ \hat{\nu}_{fx} \\ \hat{\nu}_{gd} \\ \hat{\nu}_{gx} \end{bmatrix}, \quad G(z) := \begin{bmatrix} G_u(\hat{u}, \hat{d}, \hat{x}) \\ G_d(\hat{u}, \hat{d}, \hat{x}) \\ H(\hat{u}, \hat{d}, \hat{x}) \end{bmatrix}, \quad F(z) := \begin{bmatrix} F_u(\hat{u}, \hat{d}) \\ F_d(\hat{u}, \hat{d}) \end{bmatrix},$$

which allow us to re-write (41), (35a), and (42) as

$$\nabla_u L_f(z, \lambda, \nu) = \mathbf{0}_{N_u}, \qquad \nabla_x L_f(z, \lambda, \nu) = \mathbf{0}_{N_x}, \tag{43a}$$

$$\nabla_d L_g(z, \lambda, \nu) = \mathbf{0}_{N_d}, \qquad \nabla_x L_g(z, \lambda, \nu) = \mathbf{0}_{N_x}, \tag{43b}$$

$$G(z) = \mathbf{0}_{K_u+K_d+K_x}, \qquad \lambda \odot F(z) = \mu \mathbf{1}_{M_u+M_d}, \tag{43c}$$

and (35b) as

$$\lambda \geqslant \mathbf{0}_{M_u+M_d}, \qquad F(z) \geqslant \mathbf{0}_{M_u+M_d}. \tag{43d}$$

Algorithm 3 (Primal-dual optimization with common latent variable)**.**

---

[12] Given an integer M, we denote by $\mathbf{0}_M$ and by $\mathbf{1}_M$ the M-vectors with all entries equal to 0 and 1, respectively.

[13] Given two vectors $x, y \in \mathbb{R}^n$ we denote by $x \odot y \in \mathbb{R}^n$ and by $x \oslash y \in \mathbb{R}^n$ the entry-wise product and division of the two vectors, respectively.

Step 1. *Start with estimates z_0, λ_0, ν_0 that satisfy the inequalities $\lambda_0 \geqslant 0$, $F(z_0) \geqslant 0$ in (43d) and set $k = 0$.*

It is often a good idea to start with a value for z_0 that satisfies the equality constraint $G(z_0) = 0$, and

$$\lambda_0 = \mu \mathbf{1}_{M_u+M_d} \oslash F(z_0),$$

which guarantees that we initially have $\lambda_0 \odot F(u_0) = \mu \mathbf{1}_{M_u+M_d}$.[14]

Step 2. *Linearize the equations in (43a)–(43c) around a current estimate z_k, λ_k, ν_k, leading to*

$$\begin{bmatrix} \nabla_{uz} L_f(z_k,\lambda_k,\nu_k) & \nabla_{u\nu} L_f(z_k) & \nabla_{u\lambda} L_f(z_k) \\ \nabla_{xz} L_f(z_k,\lambda_k,\nu_k) & \nabla_{x\nu} L_f(z_k) & \nabla_{x\lambda} L_f(z_k) \\ \nabla_{dz} L_g(z_k,\lambda_k,\nu_k) & \nabla_{d\nu} L_g(z_k) & \nabla_{d\lambda} L_g(z_k) \\ \nabla_{xz} L_g(z_k,\lambda_k,\nu_k) & \nabla_{x\nu} L_g(z_k) & \nabla_{x\lambda} L_g(z_k) \\ \nabla_z G(z_k) & 0 & 0 \\ -\nabla_z F(z_k) & 0 & -\operatorname{diag}[F(z_k) \oslash \lambda_k] \end{bmatrix} \begin{bmatrix} \Delta z \\ \Delta \nu \\ \Delta \lambda \end{bmatrix} = - \begin{bmatrix} \nabla_u L_f(z_k,\lambda_k,\nu_k) \\ \nabla_x L_f(z_k,\lambda_k,\nu_k) \\ \nabla_d L_g(z_k,\lambda_k,\nu_k) \\ \nabla_x L_g(z_k,\lambda_k,\nu_k) \\ G(z_k) \\ -F(z_k) + \mu \mathbf{1} \oslash \lambda_k \end{bmatrix}.$$

Since the vectors $F(z_k), \lambda_k$ have positive entries, we can solve this system of equations by first eliminating

$$\begin{aligned} \nabla_z F(z_k)\Delta z + \operatorname{diag}[F(z_k) \oslash \lambda_k]\Delta\lambda = -F(z_k) + \mu \mathbf{1} \oslash \lambda_k \quad &\Leftrightarrow \\ \Delta\lambda = -\lambda_k - \operatorname{diag}[\lambda_k \oslash F(z_k)]\nabla_z F(z_k)\Delta z + \mu \mathbf{1} \oslash F(z_k) \end{aligned}$$

which leads to

$$\begin{bmatrix} \nabla_{uz} L_f(z_k,\lambda_k,\nu_k) & \nabla_{u\nu} L_f(z_k) \\ \nabla_{xz} L_f(z_k,\lambda_k,\nu_k) & \nabla_{x\nu} L_f(z_k) \\ \nabla_{dz} L_g(z_k,\lambda_k,\nu_k) & \nabla_{d\nu} L_g(z_k) \\ \nabla_{xz} L_g(z_k,\lambda_k,\nu_k) & \nabla_{x\nu} L_g(z_k) \\ \nabla_z G(z_k) & 0 \end{bmatrix} \begin{bmatrix} \Delta z \\ \Delta \nu \end{bmatrix} = - \begin{bmatrix} \nabla_u L_f(z_k,\lambda_k,\nu_k) + \nabla_{u\lambda} L_f(z_k)\Delta\lambda \\ \nabla_x L_f(z_k,\lambda_k,\nu_k) + \nabla_{x\lambda} L_f(z_k)\Delta\lambda \\ \nabla_d L_g(z_k,\lambda_k,\nu_k) + \nabla_{d\lambda} L_g(z_k)\Delta\lambda \\ \nabla_x L_g(z_k,\lambda_k,\nu_k) + \nabla_{x\lambda} L_g(z_k)\Delta\lambda \\ G(z_k) \end{bmatrix} \quad \Leftrightarrow$$

$$\begin{bmatrix} \nabla_{uz} L_f(z_k,\lambda_k,\nu_k) - \nabla_{u\lambda} L_f(z_k) \operatorname{diag}[\lambda_k \oslash F(z_k)]\nabla_z F(z_k) & \nabla_{u\nu} L_f(z_k) \\ \nabla_{xz} L_f(z_k,\lambda_k,\nu_k) - \nabla_{x\lambda} L_f(z_k) \operatorname{diag}[\lambda_k \oslash F(z_k)]\nabla_z F(z_k) & \nabla_{x\nu} L_f(z_k) \\ \nabla_{dz} L_g(z_k,\lambda_k,\nu_k) - \nabla_{d\lambda} L_g(z_k) \operatorname{diag}[\lambda_k \oslash F(z_k)]\nabla_z F(z_k) & \nabla_{d\nu} L_g(z_k) \\ \nabla_{xz} L_g(z_k,\lambda_k,\nu_k) - \nabla_{x\lambda} L_g(z_k) \operatorname{diag}[\lambda_k \oslash F(z_k)]\nabla_z F(z_k) & \nabla_{x\nu} L_g(z_k) \\ \nabla_z G(z_k) & 0 \end{bmatrix} \begin{bmatrix} \Delta z \\ \Delta \nu \end{bmatrix}$$

$$= - \begin{bmatrix} \nabla_u L_f(z_k,\lambda_k,\nu_k) - \nabla_{u\lambda} L_f(z_k)\lambda_k + \mu \nabla_{u\lambda} L_f(z_k)\big(\mathbf{1} \oslash F(z_k)\big) \\ \nabla_x L_f(z_k,\lambda_k,\nu_k) - \nabla_{x\lambda} L_f(z_k)\lambda_k + \mu \nabla_{x\lambda} L_f(z_k)\big(\mathbf{1} \oslash F(z_k)\big) \\ \nabla_d L_g(z_k,\lambda_k,\nu_k) - \nabla_{d\lambda} L_g(z_k)\lambda_k + \mu \nabla_{d\lambda} L_g(z_k)\big(\mathbf{1} \oslash F(z_k)\big) \\ \nabla_x L_g(z_k,\lambda_k,\nu_k) - \nabla_{x\lambda} L_g(z_k)\lambda_k + \mu \nabla_{x\lambda} L_g(z_k)\big(\mathbf{1} \oslash F(z_k)\big) \\ G(z_k) \end{bmatrix} \quad \Leftrightarrow$$

$$\begin{bmatrix} \nabla_{uz} L_f(z_k,\lambda_k,\nu_k) - \nabla_{u\lambda} L_f(z_k) \operatorname{diag}[\lambda_k \oslash F(z_k)]\nabla_z F(z_k) & \nabla_{u\nu} L_f(z_k) \\ \nabla_{xz} L_f(z_k,\lambda_k,\nu_k) - \nabla_{x\lambda} L_f(z_k) \operatorname{diag}[\lambda_k \oslash F(z_k)]\nabla_z F(z_k) & \nabla_{x\nu} L_f(z_k) \\ \nabla_{dz} L_g(z_k,\lambda_k,\nu_k) - \nabla_{d\lambda} L_g(z_k) \operatorname{diag}[\lambda_k \oslash F(z_k)]\nabla_z F(z_k) & \nabla_{d\nu} L_g(z_k) \\ \nabla_{xz} L_g(z_k,\lambda_k,\nu_k) - \nabla_{x\lambda} L_g(z_k) \operatorname{diag}[\lambda_k \oslash F(z_k)]\nabla_z F(z_k) & \nabla_{x\nu} L_g(z_k) \\ \nabla_z G(z_k) & 0 \end{bmatrix} \begin{bmatrix} \Delta z \\ \Delta \nu \end{bmatrix}$$

[14] João: Starting with $\nu_0 = 0$ seems to be okay. Note that the value of ν only affects the matrix in the left-hand side of (28b) when $G(u)$ is nonlinear.

$$= -\begin{bmatrix} \nabla_u\big(f(z_k) + (\nu_{fu})_k G_u(z_k) + (\nu_{fx})_k H(z_k)\big) + \mu\nabla_{u\lambda}L_f(z_k)\big(\mathbf{1} \oslash F(z_k)\big) \\ \nabla_x\big(f(z_k) + (\nu_{fu})_k G_u(z_k) + (\nu_{fx})_k H(z_k)\big) + \mu\nabla_{x\lambda}L_f(z_k)\big(\mathbf{1} \oslash F(z_k)\big) \\ \nabla_d\big(g(z_k) + (\nu_{gd})_k G_d(z_k) + (\nu_{gx})_k H(z_k)\big) + \mu\nabla_{d\lambda}L_g(z_k)\big(\mathbf{1} \oslash F(z_k)\big) \\ \nabla_x\big(g(z_k) + (\nu_{gd})_k G_d(z_k) + (\nu_{gx})_k H(z_k)\big) + \mu\nabla_{x\lambda}L_g(z_k)\big(\mathbf{1} \oslash F(z_k)\big) \\ G(z_k) \end{bmatrix} \quad \Leftrightarrow$$

$$\begin{bmatrix} \nabla_{uz}L_f(z_k,\lambda_k,\nu_k) - \nabla_{u\lambda}L_f(z_k)\,\mathrm{diag}[\lambda_k \oslash F(z_k)]\nabla_z F(z_k) & \nabla_u G_u(z)' & \nabla_u H(z)' & 0 & 0 \\ \nabla_{xz}L_f(z_k,\lambda_k,\nu_k) - \nabla_{x\lambda}L_f(z_k)\,\mathrm{diag}[\lambda_k \oslash F(z_k)]\nabla_z F(z_k) & \nabla_x G_u(z)' & \nabla_x H(z)' & 0 & 0 \\ \nabla_{dz}L_g(z_k,\lambda_k,\nu_k) - \nabla_{d\lambda}L_g(z_k)\,\mathrm{diag}[\lambda_k \oslash F(z_k)]\nabla_z F(z_k) & 0 & 0 & \nabla_d G_d(z)' & \nabla_d H(z)' \\ \nabla_{xz}L_g(z_k,\lambda_k,\nu_k) - \nabla_{x\lambda}L_g(z_k)\,\mathrm{diag}[\lambda_k \oslash F(z_k)]\nabla_z F(z_k) & 0 & 0 & \nabla_x G_d(z)' & \nabla_x H(z)' \\ \nabla_z G_u(z_k) & 0 & 0 & 0 & 0 \\ \nabla_z G_d(z_k) & 0 & 0 & 0 & 0 \\ \nabla_z H(z_k) & 0 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} \Delta z \\ \Delta \nu \end{bmatrix}$$

$$= -\begin{bmatrix} \nabla_u\big(f(z_k) + (\nu_{fu})_k G_u(z_k) + (\nu_{fx})_k H(z_k)\big) + \mu\nabla_{u\lambda}L_f(z_k)\big(\mathbf{1} \oslash F(z_k)\big) \\ \nabla_x\big(f(z_k) + (\nu_{fu})_k G_u(z_k) + (\nu_{fx})_k H(z_k)\big) + \mu\nabla_{x\lambda}L_f(z_k)\big(\mathbf{1} \oslash F(z_k)\big) \\ \nabla_d\big(g(z_k) + (\nu_{gd})_k G_d(z_k) + (\nu_{gx})_k H(z_k)\big) + \mu\nabla_{d\lambda}L_g(z_k)\big(\mathbf{1} \oslash F(z_k)\big) \\ \nabla_x\big(g(z_k) + (\nu_{gd})_k G_d(z_k) + (\nu_{gx})_k H(z_k)\big) + \mu\nabla_{x\lambda}L_g(z_k)\big(\mathbf{1} \oslash F(z_k)\big) \\ G_u(z_k) \\ G_d(z_k) \\ H(z_k) \end{bmatrix}$$

References

[1] M. Grant and S. Boyd. Graph implementations for nonsmooth convex programs. In V. Blondel, S. Boyd, and H. Kimura, editors, *Recent Advances in Learning and Control*, volume 371 of *Lecture Notes in Control and Information Sciences*, pages 95–110. Springer Berlin / Heidelberg, 2008. (cited in p. 6, 13)

[2] L. Vandenberghe. The CVXOPT linear and quadratic cone program solvers. Technical report, Univ. California, Los Angeles, 2010. URL http://cvxopt.org/documentation. (cited in p. 3, 10, 16)