A Sequential Convex Programming Approach to Solving Quadratic Programs and Optimal Control Problems With Linear Complementarity Constraints

Jonas Hall[ID], Armin Nurkanović[ID], Florian Messerer[ID], and Moritz Diehl[ID]

Abstract—**Mathematical programs with complementarity constraints are notoriously difficult to solve due to their nonconvexity and lack of constraint qualifications in every feasible point. This letter focuses on the subclass of quadratic programs with linear complementarity constraints. A novel approach to solving a penalty reformulation using sequential convex programming and a homotopy on the penalty parameter is introduced. Linearizing the necessarily nonconvex penalty function yields convex quadratic subproblems, which have a constant Hessian matrix throughout all iterates. This allows solution computation with a single KKT matrix factorization. Furthermore, a globalization scheme is introduced in which the underlying merit function is minimized analytically, and guarantee of descent is provided at each iterate. The algorithmic features and possible computational speedups are illustrated in a numerical experiment.**

Index Terms—**Mathematical programming, quadratic programming, algorithm design and analysis, optimal control.**

I. INTRODUCTION

LINEAR Complementarity Quadratic Programs (LCQP) are quadratic programs with additional complementarity constraints. The complementarity conditions consist of inequality constraints, imposing nonnegativity of the complementary pairs, and a bi-linear equality constraint imposing orthogonality. In order to formalize this, consider an n-dimensional input space with n_C complementarity constraints. Let $L, R \in \mathbb{R}^{n_C \times n}$ be the linear input transformations selecting the complementarity pairs. Then a general LCQP can be written as

$$\text{LCQP:} \quad \underset{x \in \mathbb{R}^n}{\text{minimize}} \quad \frac{1}{2}x^\top Qx + g^\top x \tag{1a}$$

$$\text{subject to} \quad 0 \leq Ax - b, \tag{1b}$$

$$0 \leq Lx \perp Rx \geq 0, \tag{1c}$$

where $0 \prec Q \in \mathbb{R}^{n \times n}, g \in \mathbb{R}^n, A \in \mathbb{R}^{n_A \times n}$, and $b \in \mathbb{R}^{n_A}$. The complementarity constraint (1c) is a compact notation for

$$0 \leq Lx \perp Rx \geq 0 \iff \begin{cases} 0 \leq Lx \\ 0 \leq Rx \\ 0 = x^\top L^\top Rx. \end{cases} \tag{2}$$

An illustrative example is depicted in Figure 1. These problems are particularly difficult to solve due to their nonconvexity and nonsmoothness of the feasible set. Moreover, standard constraint qualifications such as the Linear Independence Constraint Qualification (LICQ) or the weaker Mangasarian-Fromovitz constraint qualification are violated at every feasible point [1, Proposition 1.1]. Thus, it is very difficulty to numerically solve (1) directly, as the multipliers are unbounded and the constraint Jacobian matrices are degenerate. Generalizations of (1) with nonlinear functions are known as Mathematical Programs with Complementarity Constraints (MPCC). These problems have received a lot of attention and many solution strategies have been proposed, many of which are included in the survey [2].

One popular approach to reformulate an MPCC into a less degenerate Non-Linear Program (NLP) is to remove the bi-linear term from the constraint and penalize its violation in the objective. Convergence and solution equivalence of the two approaches have been studied for example in [4]. This approach is further adapted to an interior point strategy in [5]. Similar penalty reformulations can be found in [6], and a more generic form in [7]. Instead of penalizing the bi-linear term, authors have suggested to replace (1c) with nonlinear complementarity functions, which vanish exactly on the complementarity set. A popular example is the Fisher-Burmeister function [8] and adaptations of it [9]. MPCCs

merit function. The according step length is controlled by an analytical globalization scheme, which minimizes the merit function in every step.

This letter is structured as follows: Section II provides the required concepts, including the exactness of the underlying penalty approach. In Section III, the novel approach to solving LCQPs using an SCP technique is presented. Further, a globalization scheme is demonstrated by analytically minimizing an exact merit function. A comparison to three state-of-the-art solution variants is provided in Section IV by solving an illustrative Optimal Control Problem (OCP). Section V concludes this letter and highlights further algorithmic improvements.

II. PENALTY REFORMULATION

This section briefly addresses the theory of stationarity for MPCCs and the underlying penalty reformulation with its convergence properties. Consider the general LCQP (1). Denote by $\mathcal{L}(x)$ and $\mathcal{R}(x)$ the sets of active constraints among those of $Lx \geq 0$ and $Rx \geq 0$, respectively. Further, let $\mathcal{W}(x) = \mathcal{L}(x) \cap \mathcal{R}(x)$ denote the set of weakly active complementarity pairs. In contrast, let $\bar{\mathcal{L}} = \mathcal{L} \setminus \mathcal{W}$ and $\bar{\mathcal{R}} = \mathcal{R} \setminus \mathcal{W}$ refer to the strongly active complementarity pairs, respectively.

Definition 1: A feasible point x of LCQP (1) is called strongly stationary, if there exist dual variables $y = (y_A, y_L, y_R) \in \mathbb{R}^{n_A} \times \mathbb{R}^{n_C} \times \mathbb{R}^{n_C}$ satisfying

$$Qx + g - A^\top y_A - L^\top y_L - R^\top y_R = 0, \tag{3a}$$

$$\min(Ax - b, y_A) = 0, \tag{3b}$$

$$y_{L_i} = 0, \quad i \in \bar{\mathcal{R}}(x), \tag{3c}$$

$$y_{R_i} = 0, \quad i \in \bar{\mathcal{L}}(x), \tag{3d}$$

$$y_{L_i}, y_{R_i} \geq 0, \quad i \in \mathcal{W}(x). \tag{3e}$$

For more details on stationarity of MPCCs, including other stationarity types, refer to [21, Section 2].

A popular approach to solving LCQPs are penalty reformulations, and the here presented algorithm is based on a technique as discussed for example in [4, Section 1]. Consider the penalty function

$$\varphi(x) = x^\top L^\top R x = \frac{1}{2} x^\top (L^\top R + R^\top L) x = \frac{1}{2} x^\top C x, \tag{4}$$

where $C \in \mathbb{R}^{n \times n}$ is the symmetrization of the complementarity product.

Remark 1: The matrix C is usually indefinite. This is necessarily the case if L and R consist of pairwise orthogonal rows, e.g., if each row selects a distinct optimization variable. For example, the curvature of the complementarity product $x_1^\top x_2$ at the origin towards $(1, 1)$ is positive, whereas it is negative towards $(-1, 1)$.

The penalty reformulation is obtained by replacing the bilinear complementarity constraint from (1c) with the penalty function (4) in the objective. The resulting QP reads as

$$\rho\text{LCQP:} \quad \underset{x \in \mathbb{R}^n}{\text{minimize}} \quad \frac{1}{2} x^\top Q x + g^\top x + \rho \cdot \varphi(x) \tag{5a}$$

$$\text{subject to} \quad 0 \leq Ax - b, \tag{5b}$$

$$0 \leq Lx, \tag{5c}$$

$$0 \leq Rx, \tag{5d}$$

where $\rho > 0$ is the respective penalty parameter. Throughout this letter all solutions of ρLCQP are assumed to satisfy LICQ.
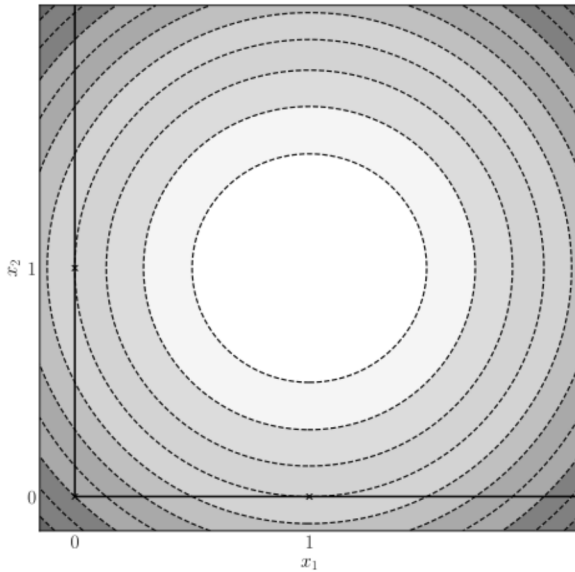

Fig. 1. An illustrative LCQP with $Q = 2\mathbb{I}_2$, $g = (-2, -2)^\top$, $L = (1, 0)$, and $R = (0, 1)$, as originally presented in [3]. The feasible set is depicted by the solid black line. This example contains two strongly stationary points located at (0, 1) and (1, 0) (which are local minima), and one weaker (Clarke-)stationary point at the origin (which is a local maximum).

have also been solved as nonlinear programs by replacing the complementarity product with an inequality constraint [10], or with (in)equality regularization schemes [4]. All these approaches lead to solving NLPs where constraint qualifications are satisfied. Usually a sequence of relaxed problems must be solved to recover a solution of the initial MPCC [4], [5]. Their limits in applications to direct optimal control have been shown in [11].

MPCCs appear in a wide range of applications in engineering, economics and science. An extensive list of applications is presented in [12]. Typical applications in mechanics appear in the context of friction problems or with impacts of rigid bodies [13]. Many of the MPCC examples appear as LCQPs, e.g., in [14] or [15].

Other authors have focused on this subclass before. An analysis of C-stationary points using homotopy approaches is provided in [16]. They also appear in the form of subproblems, when nonlinear programs with linear complementarity constraints are solved with a sequential quadratic programming approach [17]. LCQPs are also addressed as Mixed-Integer Quadratic Programs (MIQP), for example in a combination with the Benders scheme in [18] or using branch-and-bound techniques [19]. Similar to the approach presented here, the latter exploits linear algebra structures by reusing a factorized matrix. MIQP methods can be advantageous in some cases, in particular due to their ability of finding global solutions. However, their relaxations are weak which might result in large search trees and they are thus limited to formulations without too many integer variables [20].

The contribution of this letter is the introduction of a novel algorithm for solving LCQPs. The solution strategy is based on an existing penalty reformulation, which is solved with a Sequential Convex Programming (SCP) approach. Each iterate within the convex programming loop is shown to reduce the

Under this assumption a convergence property of the penalized approach is captured in the following theorem, which represents a special case of the theorem proven by Ralph and Wright in [4, Section 5].

Theorem 1: Let ρLCQP (5) satisfy LICQ at $x^* \in \mathbb{R}^n$, then the following statements hold:

  (i) If x^* is a strongly stationary point of the LCQP (1), then there exists a finite $\widetilde{\rho}$, such that x^* is a KKT point of (5) for all $\rho > \widetilde{\rho}$.
  (ii) If x^* is a KKT point of (5) and $\varphi(x^*) = 0$, then x^* is a strongly stationary point of the LCQP.

III. ALGORITHMIC DEVELOPMENT

The above motivates finding stationary points of the LCQP by solving (5) for a penalty large enough to ensure satisfaction of the complementarity constraints. In this section an approach to finding solutions via a penalty homotopy together with a sequential convex programming approach is introduced. Finally, an analytical globalization scheme with a guarantee of descent is provided.

A. Penalty Homotopy

For a given penalty parameter $\rho_k > 0$ the respective penalty reformulation (5) is solved. Subsequently, the penalty parameter $\rho_{k+1} = \beta\rho_k$ is updated for a fixed $\beta > 1$. This procedure is repeated until complementarity is satisfied. Theorem 1 ensures that if the LCQP has a strongly stationary point, then this point must also be a KKT point of the penalized reformulation with respect to a finite penalty parameter $\tilde{\rho}$. One could consider simply solving (5) for a very large penalty in the hope of instantly satisfying complementarity. However, these penalty reformulations often become ill-conditioned for large penalty parameters as the nonconvex part becomes dominant. Due to this nonconvexity, the original LCQP may contain many local solutions. Solving the penalized subproblem for a very small penalty parameter leads to a solution close to the global minimum of (1) without the bi-linear complementarity constraint, whereas a solution with respect to a large penalty parameter favors complementarity satisfaction. This motivates a homotopy on the penalty parameter with the aim of finding a good local solution [5]. Yet, this is only a heuristic and there is no guarantee of finding the global minimizer as the original NLP (1) is nonconvex. Further, a homotopy often avoids convergence to spurious solutions, as for example shown for OCPs with discontinuous systems [11]. The sequence of these penalized subproblems (5) is denoted as the outer loop.

B. The Sequential Convex Programming Approach

Each outer loop problem is solved using sequential convex programming, resulting in an inner loop. Let k and j denote the outer and inner loop indices, respectively. The very first inner loop is initialized with the initial guess, whereas all consecutive inner loops are initialized with the previous iterate. The penalty function is approximated at each iterate x_{kj} using its first-order Taylor expansion

$$\varphi(x) \approx \varphi(x_{kj}) + (x - x_{kj})^\top \nabla\varphi(x_{kj})$$
$$= \left(\varphi(x_{kj}) - x_{kj}^\top C x_{kj}\right) + x^\top C x_{kj}.$$

Now let $d_{kj} = C x_{kj}$, and note that $x^\top d_{kj}$ is the only term dependent on x. The penalty function is replaced by $d_{kj}^\top x$ resulting in the convex inner loop subproblem

$$\underset{x\in\mathbb{R}^n}{\text{minimize}} \quad \frac{1}{2}x^\top Q x + \left(g + \rho_k d_{kj}\right)^\top x \tag{6a}$$

$$\text{subject to} \quad 0 \leq Ax - b, \tag{6b}$$

$$0 \leq Lx, \tag{6c}$$

$$0 \leq Rx. \tag{6d}$$

Denote the unique minimizer of the inner loop subproblem by x_{kj}^* and the according step by $p_{kj} = x_{kj}^* - x_{kj}$. Given this inner solution an optimal step length α_{kj} is obtained from the globalization scheme described in Section III-C. Finally, the step update $x_{k,j+1} = x_{kj} + \alpha_{kj}p_{kj}$ is performed. The inner loop is terminated once a KKT point of the respective outer loop problem (5) is found. The following lemma relates the minimizers of the inner loop problems with the KKT points of the outer loop. A proof for a more general case can be found in [22, Lemma 4.1].

Lemma 1: Let (x_{kj}, ρ_k) be a feasible iterate of (6). Then the respective inner loop minimizer x_{kj}^* agrees with x_{kj} iff x_{kj} is a KKT point of the outer loop problem (5) with respect to ρ_k.

The algorithm is terminated once a complementarity satisfying KKT point of (5) is found. Theorem 1 indicates that the solution must be a strongly stationary point of the original LCQP, under the assumption of exact complementarity satisfaction.

There are two reasons why it can be attractive to replace the full penalty function by its linear approximation. First, convex subproblems are obtained at the cost of the additional inner loop. While the original formulation becomes more and more indefinite as the penalty parameter grows, convexity of the inner loop subproblem is always ensured, as the Hessian matrix remains to be Q in every subproblem. This also induces the second advantage: the Hessian and constraint matrices remain constant over all (inner and outer) iterates. Consequently, the KKT matrix factorization can be reused, and each inner loop problem can be solved efficiently, e.g., by making use of the hot-starting technique employed in active-set solvers such as qpOASES [23]. With the computation of factorizations being a significant expense, this advantage can outweigh the cost of inner loop iterations, as demonstrated in Section IV.

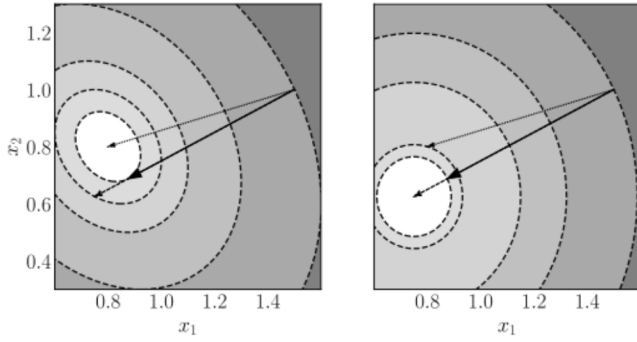### C. Optimal Step Length Globalization

Consider the merit function

$$\psi(x, \rho) = \frac{1}{2}x^\top (Q + \rho C)x + g^\top x, \tag{7}$$

which coincides with the outer loop objective function. On the other hand, the inner loop objective function provides the strictly convex quadratic model

$$\vartheta_{kj}(x) = \frac{1}{2}x^\top Q x + (g + \rho_k d_{kj})^\top x. \tag{8}$$

An analytical globalization scheme by is introduced by solving

$$\underset{\alpha\in[0,1]}{\text{minimize}} \quad \psi(x_{kj} + \alpha x_{kj}^*, \rho). \tag{9}$$

Fig. 2. The left plot shows the level lines of the NLP objective (5a), and the right plot depicts the level lines of the convex QP subproblem objective (6a). Both plots include a total of three vectors. Out of the parallel vectors, the dashed one shows $p_{kj} = x_{kj}^* - x_{kj}$, and the solid one is $x_{kj} + \alpha_{kj} p_{kj}$. The dotted vector indicates the stationary point of the outer loop problem (5). Note that the chosen step length minimizes (5a) on the path $x_{kj} + \alpha p_{kj}$.

Fig. 3. Illustration of the merit function (10) for $\delta_{kj} > 0$ (dashed), $\delta_{kj} < 0$ (dotted), and the quadratic model $\delta_{kj} = 0$ (solid) along the path $x_{kj} + \alpha p_{kj}$ for $\alpha \in [0, 1]$. Note that all functions must have equal descent at x_{kj}, as the linearization of the merit function and quadratic model agree at this point (dotted line). Further, note that the quadratic model is minimized for $\alpha = 1$, as it coincides with the unique minimizer of the inner loop subproblem.

This concept is visualized in Figure 2. Evaluating the objective function in (9) yields the quadratic polynomial in α

$$\psi(x_{kj} + \alpha p_{kj}, \rho_k) = \frac{1}{2}\alpha^2 q_{kj} + \alpha \ell_{kj} + \psi(x_{kj}, \rho_k), \quad (10)$$

where

$$\gamma_{kj} = p_{kj}^\top Q p_{kj}, \quad (11a)$$

$$\delta_{kj} = p_{kj}^\top \rho_k C p_{kj}, \quad (11b)$$

$$q_{kj} = \gamma_{kj} + \delta_{kj}, \quad (11c)$$

$$\ell_{kj} = x_{kj}^\top (Q + \rho_k C) p_{kj} + g^\top p_{kj}. \quad (11d)$$

There are two different cases to be considered for solving (9), both related to the sign of δ_{kj}. Both cases are handled individually and their geometric meanings are discussed.

First consider the sign of the linear component. Observe that ℓ_{kj} represents the directional derivative of the merit function along p_{kj}, i.e.,

$$\nabla \psi(x_{kj}, \rho_k)^\top p_{kj} = ((Q + \rho_k C) x_{kj} + g)^\top p_{kj} = \ell_{kj}. \quad (12)$$

This provides descent along p_{kj} at $\psi(x_{kj}, \rho_k)$ given $\ell_{kj} < 0$. The following guarantee of descent is supplied:

Lemma 2 (Direction of Descent): Given a feasible point x_{kj} of (5) and inner loop iterate $p_{kj} = x_{kj}^* - x_{kj}$, the merit function at x_{kj} is nonincreasing towards p_{kj}, i.e.,

$$\nabla \psi(x_{kj}, \rho_k)^\top (x_{kj}^* - x_{kj}) \leq 0. \quad (13)$$

Further, if x_{kj} is not a stationary point of (5) with respect to ρ_k, then
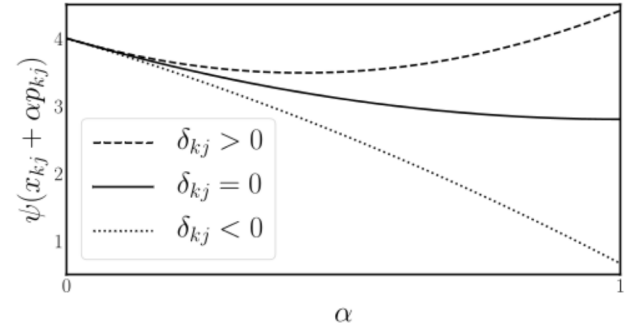
$$\nabla \psi(x_{kj}, \rho_k)^\top (x_{kj}^* - x_{kj}) < 0. \quad (14)$$

Proof: Since x_{kj}^* is the global minimum of the inner loop optimization problem, the following relation holds

$$\vartheta_{kj}(x_{kj}^*) \leq \vartheta_{kj}(x), \quad (15)$$

where x is any feasible point of (5). Since ϑ_{kj} is convex and differentiable it holds for any $a, b \in \mathbb{R}^n$ that

$$\nabla \vartheta_{kj}(a)^\top (b - a) \leq \vartheta_{kj}(b) - \vartheta_{kj}(a). \quad (16)$$

This property provides descent for the quadratic model

$$\nabla \vartheta_{kj}(x_{kj})^\top p_{kj} \leq \vartheta_{kj}(x_{kj}^*) - \vartheta_{kj}(x_{kj}) \leq 0, \quad (17)$$

Note that this inequality is strict if $x_{kj} \neq x_{kj}^*$. Recall that $C x_{kj} = d_{kj}$, then the equation

$$\nabla \psi(x_{kj}, \rho_k)^\top p_{kj} = (Q x_{kj} + \rho_k C x_{kj} + g)^\top p_{kj} \quad (18a)$$

$$= \nabla \vartheta_{kj}(x_{kj})^\top p_{kj} \quad (18b)$$

shows that the directional derivatives of the merit function and quadratic model at x_{kj} towards p_{kj} agree. Inequality (13) follows immediately.

Assume that x_{kj} is not outer loop stationary. Then Lemma 1 yields $x_{kj}^* \neq x_{kj}$. As captured before, the inequality (17) becomes strict, and again Equation (18) shows the statement. ∎

Now consider the sign of δ_{kj}, which determines whether the curvature of the merit function along p_{kj} is more or less positive than the curvature of the quadratic model. The positive case $\delta_{kj} > 0$ indicates that the merit function along p_{kj} has a stronger positive curvature, and thus is minimized before the full step. The optimal step length in this case is given by

$$\alpha_{kj} = \frac{-\ell_{kj}}{q_{kj}}. \quad (19)$$

Note that $q_{kj} > \delta_{kj} > 0$. Assuming that x_{kj} is not already a stationary point, a strictly positive step length is obtained. The nonpositive case $\delta_{kj} \leq 0$ leads to $\alpha_{kj} = 1$, as the curvature of the merit function is less positive than the quadratic model. Both scenarios are visualized in Figure 3. Though this strategy provides the best available step length at each iterate, there does not yet exist a statement for sufficient decrease, which would ensure convergence of the inner loop. However, in practice this has not caused any complications.

D. Pseudo Code

This section provides a pseudo code description of the presented approach, which summarizes all features described before. For simplicity, the inner loop index j is dropped. The algorithm requires the input of an initial guess `x0` $\in \mathbb{R}^n$ and initial penalty parameter `rho` > 0, as well as a penalty

---

Algorithm 1: Pseudo Code of the Approach

Input: `x0, rho, beta`
Output: Stationary point `(xk, yk)` of LCQP (1)

1 # Create QP solver and factorize KKT matrix
2 `qp(Q, g, A, lb, ub, lbA, ubA);`

3 # Initialize solver with zero penalty QP
4 `(xk, yk) = qp.solve();`

5 # Outer loop (penalty update loop)
6 **while** true **do**

7　　# Inner loop (approximate penalty function)
8　　**while** `Stat(xk, yk, rho) > tol` **do**

9　　　　# Update objective's linear component
10　　　`qp.update_g(g + rho*dk);`

11　　　　# Step computation (solve (6))
12　　　`(xn, yk) = qp.solve();`

13　　　　# Get optimal step length according to (III-C)
14　　　`alpha = StepLength(xk, xn, rho);`

15　　　　# Perform step
16　　　`xk = xk + alpha*(xn - xk);`

17　　# Terminate or increase penalty parameter
18　　**if** `Phi(xk) < tol` **then**
19　　　`return (xk, yk);`

20　　`rho = beta*rho;`

---

update factor `beta > 1`. The output of this pseudo code is a stationary point of the LCQP (1). The objective's Hessian is denoted by `Q` and its linear term by `g`. The matrix `A` contains all linear constraints, i.e., it includes the rows of A, L and R. During the initialization of the QP solver, the KKT matrix is assumed to be factorized. Line 6 begins the outer loop, where termination is checked and the penalty value is updated. The inner loop begins in Line 8, in which the subproblem (6) is solved and step updates are performed. The variable `xn` represents the inner loop minimizer x_{kj}^*. The inner loop is terminated if the outer loop stationarity, denoted by `Stat(xk, yk, rho)`, drops below a certain threshold `tol`. Similarly, the outer loop is terminated once the complementarity value, denoted by `Phi(xk)`, falls below this tolerance.

IV. A NUMERICAL EXAMPLE

This section briefly discusses a numerical benchmark by solving the illustrative OCP [15, Section 2]

$$\underset{x_0 \in \mathbb{R}, x(\cdot)}{\text{minimize}} \int_0^2 x(t)^2 \mathrm{d}t + (x(2) - 5/3)^2 \tag{20a}$$

$$\text{subject to } x(0) = x_0, \tag{20b}$$

$$\dot{x}(t) \in 2 - \mathrm{sgn}(x(t)), \quad t \in [0, 2]. \tag{20c}$$

The effective degree of freedom in this optimization problem is the initial value x_0. Though the constraint (20c) is a discontinuous ODE, it has a unique solution given by a piecewise linear function with slope 3 for $x(t) < 0$ and slope 1 for $x(t) > 0$. The ODE describes a Filippov differential inclusion,

TABLE I
AVERAGE VALUES OVER ALL EXPERIMENTS

	complementarity	distance to analytical solution
LCQP	**6.8e−17**	**0.018**
LCQP Schur	2.3e−16	**0.018**
IPOPT Penalty	1.4e−04	0.072
IPOPT Smoothed	1.6e+04	0.078
IPOPT Relaxed	6.0e+03	0.61

which can be reformulated into a dynamic complementarity system [11]. This method introduces three algebraic variables $y(\cdot), \lambda^-(\cdot), \lambda^+(\cdot)$, which describe the switch in the ODE, the negative part of x and the positive part of x, respectively. However, $\lambda^+ = x + \lambda^-$ can be eliminated. This OCP is discretized using implicit Euler in order to obtain the LCQP

$$\underset{\substack{x_0, \ldots, x_N \in \mathbb{R} \\ y_0, \ldots, y_{N-1} \in \mathbb{R} \\ \lambda_0^-, \ldots, \lambda_{N-1}^- \in \mathbb{R}}}{\text{minimize}} \sum_{k=0}^{N-1} E_k(x_k) + E_N(x_N)$$

$$\text{subject to } x_{k-1} + h(3 - 2y_k) = x_k, \quad 1 \le k < N,$$

$$0 \le x_k + \lambda_k^- \perp 1 - y_k \ge 0, \quad 1 \le k < N,$$

$$0 \le \lambda_k^- \perp y_k \ge 0, \quad 1 \le k < N,$$

where E_k for $0 \le k < N$ represents the quadrature formula of the implicit Euler discretization, and E_N the terminal cost. The discretized problem represents an LCQP (after a small regularization on the algebraic variables).

In the following, five different solution variants are compared, two of which are based on a MATLAB implementation of the presented algorithm. These two methods differ only in the used linear solver within the QP subproblem solver qpOASES [24]: one uses the default dense solver, whereas the other utilizes the Schur complement method for which the sparse solver MA57 [25] is required. These methods are denoted by LCQP and LCQP Schur, respectively. The remaining three methods are all solved with IPOPT [26] through the CasADi interface [27]: one method, denoted by IPOPT Pen, solves the exact same outer loop problems as the LCQP methods, and the other two strategies solve a homotopy of (in)equality regularization schemes. The regularization schemes replace the complementarity product with $x^\top L^\top R x \le \sigma$ for the relaxed method, and $x^\top L^\top R x = \sigma$ for the smoothed method, both for some $\sigma > 0$ (see [11] for details). These methods are denoted by IPOPT Smoothed and IPOPT Relaxed. The source code of this benchmark is available at https://github.com/hallfjonas/IVOCP.

Table I provides the average complementarity satisfaction together with the average absolute distance to the analytical solution, showing that the proposed algorithm has the highest quality in both aspects. In fact, complementarity is satisfied up to machine precision, which is favoured by having an active-set solver on the subproblem level. Solutions computed with the IPOPT penalty method achieve significantly less precision (due to its conflicting barrier penalty), and the regularization schemes only achieve a low complementarity satisfaction naturally. Figure 4 shows the average CPU times of this experiment. The introduced method outperforms all other approaches
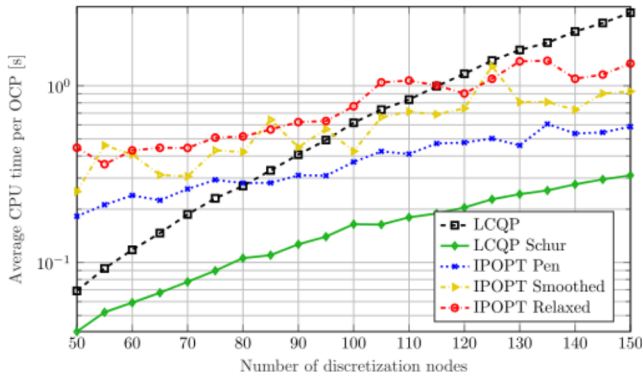
Fig. 4. Plotting the average CPU time required for each method and discretization size to solve 100 differently initialized LCQPs.

in the first few discretizations. This originates from the fact that the factorization of the KKT matrix is reused, whereas IPOPT is required to recompute it after each penalty update. As the experiments gain size, IPOPT performs better in terms of CPU time than the introduced algorithm, which is due to its exploitation of sparsity structures. However, if a solver like MA57 is available, the LCQP Schur method is able to outperform IPOPT in all experiments. Both regularized methods are unable to compete against the penalty approaches for moderately sized formulations.

V. Conclusion and Future Work

This letter presented a novel SCP approach to solving LCQPs. A computationally cheap globalization strategy with the guarantee of merit function descent at each iterate was introduced. Its applicability and promising performance was demonstrated by solving an initial value optimal control problem. A comparison against state-of-the-art solution variants solved by a high performance NLP-solver showed that the algorithm is able to compete in all of the three categories: solution CPU time, complementarity satisfaction, and solution quality. Future work aims at providing an open-source software package to reliably solve LCQPs. The presented algorithm will be implemented with multiple QP solvers on the subsolver level (e.g., qpOASES [24] and OSQP [28]). Further, the option to solely solve the outer loop homotopy with an adequate solver could be provided. On the theoretical side, future work consists of providing a proof of global convergence regarding both inner and outer loops. Additionally, the presented algorithm could be utilized on a subsolver level for solving nonlinear MPCCs opening up applicability to a wider range of problems.

References

[1] J. J. Ye, D. L. Zhu, and Q. J. Zhu, "Exact penalization and necessary optimality conditions for generalized bilevel programming problems," *SIAM J. Optim.*, vol. 7, no. 2, pp. 481–507, 1997.

[2] Y. Kim, S. Leyffer, and T. Munson, "MPEC methods for bilevel optimization problems," in *Bilevel Optimization: Advances and Next Challenges*. Cham, Switzerland: Springer Int., 2020, pp. 335–360.

[3] H. Scheel and S. Scholtes, "Mathematical programs with complementarity constraints: Stationarity, optimality, and sensitivity," *Math. Oper. Res.*, vol. 25, no. 1, pp. 1–22, 2000.

[4] D. Ralph and S. J. Wright, "Some properties of regularization and penalization schemes for MPECs," *Optim. Methods Softw.*, vol. 19, no. 5, pp. 527–556, 2004.

[5] S. Leyffer, G. López-Calva, and J. Nocedal, "Interior methods for mathematical programs with complementarity constraints," *SIAM J. Optim.*, vol. 17, no. 1, pp. 52–77, 2006.

[6] L. Abdallah, T. Migot, and M. Haddou, "Solving mathematical programs with complementarity constraints with a penalization approach," in *Optimization of Complex Systems: Theory, Models, Algorithms and Applications*. Cham, Switzerland: Springer Int., 2020, pp. 228–237.

[7] M. Anitescu, "On using the elastic mode in nonlinear programming approaches to mathematical programs with complementarity constraints," *SIAM J. Optim.*, vol. 15, no. 4, pp. 1203–1236, 2005.

[8] A. Fischer, "An NCP–function and its use for the solution of complementarity problems," in *Recent Advances in Nonsmooth Optimization*. Singapore: World Sci., 1995, pp. 88–105.

[9] B. Chen, X. Chen, and C. Kanzow, "A penalized Fischer-Burmeister NCP-function," *Math. Program.*, vol. 88, no. 1, pp. 211–216, 2000.

[10] R. Fletcher and S. Leyffer, "Solving mathematical programs with complementarity constraints as nonlinear programs," *Optim. Methods Softw.*, vol. 19, no. 1, pp. 15–40, 2004.

[11] A. Nurkanović, S. Albrecht, and M. Diehl, "Limits of MPCC formulations in direct optimal control with nonsmooth differential equations," in *Proc. Eur. Control Conf. (ECC)*, St. Petersburg, Russia, 2020, pp. 2015–2020.

[12] M. C. Ferris and J.-S. Pang, "Engineering and economic applications of complementarity problems," *SIAM Rev.*, vol. 39, no. 4, pp. 669–713, 1997.

[13] D. E. Stewart, *Dynamics with Inequalities*. Philadelphia, PA, USA: Soc. Ind. Appl. Math., 2011.

[14] D. E. Stewart, "A numerical method for friction problems with multiple contacts," *J. Aust. Math. Soc. B, Appl. Math.*, vol. 37, no. 3, pp. 288–308, 1996.

[15] D. E. Stewart and M. Anitescu, "Optimal control of systems with discontinuous differential equations," *Numerische Mathematik*, vol. 114, no. 4, pp. 653–695, 2010.

[16] D. Ralph and O. Stein, "The C-index: A new stability concept for quadratic programs with complementarity constraints," *Math. Oper. Res.*, vol. 36, no. 3, pp. 504–526, 2011.

[17] M. Fukushima, Z.-Q. Luo, and J.-S. Pang, "A globally convergent sequential quadratic programming algorithm for mathematical programs with linear complementarity constraints," *Comput. Optim. Appl.*, vol. 10, no. 1, pp. 5–34, 1998.

[18] L. Bai, J. E. Mitchell, and J.-S. Pang, "On convex quadratic programs with linear complementarity constraints," *Comput. Optim. Appl.*, vol. 54, no. 3, pp. 517–554, 2013.

[19] B. Stellato, V. V. Naik, A. Bemporad, P. Goulart, and S. Boyd, "Embedded mixed-integer quadratic optimization using the OSQP solver," in *Proc. Eur. Control Conf. (ECC)*, Limassol, Cyprus, 2018, pp. 1536–1541.

[20] J.-S. Pang, "Three modeling paradigms in mathematical programming," *Math. Program.*, vol. 125, no. 2, pp. 297–323, 2010.

[21] L. Guo, G.-H. Lin, and J. Y. Jane, "Solving mathematical programs with equilibrium constraints," *J. Optim. Theory Appl.*, vol. 166, no. 1, pp. 234–256, 2015.

[22] F. Messerer, K. Baumgärtner, and M. Diehl. (2021). *Survey of Sequential Convex Programming and Generalized Gauss–Newton Methods*. [Online]. Available: www.optimization-online.org/DB_HTML/2020/04/7733.html

[23] H. J. Ferreau, H. G. Bock, and M. Diehl, "An online active set strategy to overcome the limitations of explicit MPC," *Int. J. Robust Nonlinear Control*, vol. 18, no. 8, pp. 816–830, 2008.

[24] H. J. Ferreau, C. Kirches, A. Potschka, H. G. Bock, and M. Diehl, "qpOASES: A parametric active-set algorithm for quadratic programming," *Math. Program. Comput.*, vol. 6, no. 4, pp. 327–363, 2014.

[25] I. S. Duff, "MA57—A code for the solution of sparse symmetric definite and indefinite systems," *ACM Trans. Math. Softw.*, vol. 30, pp. 118–144, Jun. 2004.

[26] A. Wächter and L. T. Biegler, "On the implementation of an interior-point filter line-search algorithm for large-scale nonlinear programming," *Math. Program.*, vol. 106, no. 1, pp. 25–57, 2006.

[27] J. A. E. Andersson, J. Gillis, G. Horn, J. B. Rawlings, and M. Diehl, "CasADi: A software framework for nonlinear optimization and optimal control," *Math. Program. Comput.*, vol. 11, pp. 1–36, Jul. 2018.

[28] B. Stellato, G. Banjac, P. Goulart, A. Bemporad, and S. Boyd, "OSQP: An operator splitting solver for quadratic programs," *Math. Program. Comput.*, vol. 12, no. 4, pp. 637–672, 2020.