A Direct Method for Solving Integral Penalty Transcriptions of Optimal Control Problems

Martin P. Neuenhofen[1] and Eric C. Kerrigan[2]

Abstract— **We present a numerical method for the minimization of objectives that are augmented with large quadratic penalties of overdetermined inconsistent equality constraints. Such objectives arise from quadratic integral penalty methods for the direct transcription of equality constrained optimal control problems. The Augmented Lagrangian Method (ALM) has a number of advantages over the Quadratic Penalty Method (QPM) for solving this class of problems. However, if the equality constraints of the discretization are inconsistent, then ALM might not converge to a point that minimizes the unconstrained bias of the objective and penalty term. Therefore, in this paper we explore a modification of ALM that fits our purpose. Numerical experiments demonstrate that the modified ALM can minimize certain quadratic penalty-augmented functions faster than QPM, whereas the unmodified ALM converges to a minimizer of a significantly different problem.**

I. Motivation in the Optimal Control Context

The method of choice for the numerical solution of optimal control problems is direct transcription. Typical direct transcriptions methods use orthogonal collocation [2]. It is known that the latter can struggle with singular arc and high-index differential algebraic equalities (DAEs); the former arising in the example

$$\min_{y,u} \quad J = \int_0^{\frac{\pi}{2}} \left(y(t)^2 + \cos(t)u(t) \right) \, dt, \tag{OCP}$$
$$\text{s.t.} \quad y(0) = 0, \quad \dot{y}(t) = \frac{1}{2}y(t)^2 + u(t),$$

which has the analytic solution $y^\star(t) = \frac{\sin(t)}{\cos(t)-2}$ and $J^\star \approx -0.2569969625$.

Quadratic integral penalty methods [1], [4], [6] are an alternative to collocation methods, where the squared path equality constraint residual is integrated and added as a penalty into the objective. In [6] the authors present such a method with a proof of convergence under mild assumptions, including convergence for singular arcs and high-index DAEs. This is verified in [6] in comparison to collocation methods via numerical experiments.

[1]Martin P. Neuenhofen is with the Deparment of Electrical & Electronic Engineering, Imperial College London, SW7 2AZ London, UK, e-mail: m.neuenhofen19@imperial.ac.uk `www.MartinNeuenhofen.de`

[2]Eric C. Kerrigan is with the Deparment of Electrical & Electronic Engineering and Department of Aeronautics, Imperial College London, SW7 2AZ London, UK, e-mail: e.kerrigan@imperial.ac.uk `www.imperial.ac.uk/people/e.kerrigan`

Before proceeding, we guide the reader through the solution of (OCP) via the quadratic integral penalty method: let y, u be represented with continuous piecewise linear finite element functions y_h, u_h on a uniform mesh of $N \in \mathbb{N}$ elements, $h := \frac{\pi}{2N}$; represented with $\mathbf{x} := [y_h(h), \ldots, y_h(Nh), u_h(0), \ldots, u_h(Nh)]^\mathsf{T} \in \mathbb{R}^n$, $n := 2N + 1$. $y_h(0) = 0$ is fixed and removed from $\mathbf{x}$. We minimize

$$\min_{\mathbf{x} \in \mathbb{R}^{2N+1}} \quad \Phi_\omega(\mathbf{x}) := \int_0^{\frac{\pi}{2}} \left(y_h(t)^2 + \cos(t)u_h(t) \right) \, dt$$
$$+ \frac{1}{2\omega} \int_0^{\frac{\pi}{2}} \left\| -\dot{y}_h(t) + \frac{1}{2}y_h(t)^2 + u_h(t) \right\|_2^2 \, dt \, . \tag{1}$$

The integrals are evaluated with Gauss-Legendre quadrature of $q = 8$ points per element. Writing τ, α for abscissae and weights, $m := Nq$, and

$$f(\mathbf{x}) := \sum_{j=1}^{Nq} \alpha_j \left(y_h(\tau_j)^2 + \cos(\tau_j)u_h(\tau_j) \right) \tag{2a}$$

$$c(\mathbf{x}) := \begin{bmatrix} \vdots \\ \sqrt{\alpha_j} \left(-\dot{y}_h(\tau_j)^2 + \frac{1}{2}y_h(\tau_j)^2 + u_h(\tau_j) \right) \\ \vdots \end{bmatrix} \in \mathbb{R}^m \tag{2b}$$

allows us to express (1) as an *unconstrained quadratic penalty program*:

$$\min_{\mathbf{x} \in \mathbb{R}^n} \quad \Phi_\omega(\mathbf{x}) = f(\mathbf{x}) + \frac{1}{2\omega}\|c(\mathbf{x})\|_2^2 \, , \tag{UQPP}$$

$\omega \in \mathbb{R}_+ \setminus \{0\}$ controls the quadratic penalty and should be chosen on the order of approximation of the finite element space [4], [6].

A sometimes related problem is the *equality constrained program*:

$$\min_{\mathbf{x} \in \mathbb{R}^n} \quad f(\mathbf{x}) \quad \text{s.t.} \quad c(\mathbf{x}) = \mathbf{0} \in \mathbb{R}^m \, , \tag{ECP}$$

with Lagrangian $\mathcal{L}(\mathbf{x}, \boldsymbol{\lambda}) := f(\mathbf{x}) - \boldsymbol{\lambda}^\mathsf{T} c(\mathbf{x})$ and Lagrange multiplier $\boldsymbol{\lambda} \in \mathbb{R}^m$. The Karush-Kuhn Tucker optimality system of (UQPP) and (ECP) is

$$\nabla_{\mathbf{x}}\mathcal{L}(\mathbf{x}, \boldsymbol{\lambda}) = \mathbf{0}$$
$$c(\mathbf{x}) + \omega\boldsymbol{\lambda} = \mathbf{0} \tag{KKT}$$

where $\omega = 0$ for (ECP). The KKT multiplier for (UQPP) is a substitution trick such that $\nabla\mathcal{L}$ matches $\nabla\Phi_\omega$.

We recommend the use of the above penalty finite element method when numerically solving optimal control problems, because penalty methods have favourable convergence properties over collocation methods [6].

A difficulty that remains is with numerically solving (UQPP). This can be more challenging than solving a problem of the form (ECP). Below we describe important details.

A. Problems (UQPP) and (ECP) have different solutions

In our discretization (2) it holds that $m = Nq \equiv 8N$ and $n = 2N + 1 \ll m$, hence (ECP) is *overdetermined*. Clearly, problem (UQPP) cannot be overdetermined because it is unconstrained.

Considering problem (ECP) in the context of overdetermination poses the risk of *inconsistency*. For instance, suppose that in (OCP) we had added the constraint $y(\pi/2) = -\frac{1}{2}$. Then $c(\mathbf{x}) \neq \mathbf{0} \; \forall \mathbf{x} \in \mathbb{R}^n$ would hold, i.e. (ECP) would be infeasible due to inconsistent constraints c. In contrast, problem (UQPP) obviously possesses feasible points because it is unconstrained.

For our example, (ECP) is feasible, i.e. $\exists \mathbf{x} \in \mathbb{R}^n :$ $c(\mathbf{x}) = \mathbf{0}$. Namely, the equality is achieved (only) by $\mathbf{x} = \mathbf{0}$, regardless of the discretization parameter h. This is far away from $y^\star$, to which solutions of (UQPP) converge as $h, \omega \searrow +0$. In conclusion, solutions to (ECP) can differ significantly from solutions to (UQPP).

The Modified Augmented Lagrangian Method (MALM), discussed in Section II, converges to minimizers of (UQPP) instead of (ECP).

B. Solutions of (UQPP) depend on the value of ω

As experimentally verified in [4] and outlined in the analysis in [6], the discretization (1)–(2) converges when both $h, \omega \searrow +0$. That is, for fixed h, too large values of ω result in bad feasibility of the numerical optimal control solution, whereas too small values of ω result in feasible, yet far from optimal solutions.

Fig. 1 demonstrates this. Our discretization of (OCP) with $N = 40$ is solved $\forall \omega \in \{10^2, 10^{-1}, 10^{-4}\}$. The value ω determines the bias between minimization of $f(\mathbf{x})$ and $\|c(\mathbf{x})\|_2 \equiv \| - \dot{y}_h + \frac{1}{2} y_h + u_h \|_{L^2(0, \pi/2)}$; note that the latter is accurate due to the order of quadrature. For $\omega = 10^{-1}$ the solution achieves a good trade-off between feasibility and optimality on that coarse mesh.

In conclusion, the value of ω can have a significant influence on the solution of (UQPP) and it is hence important that (UQPP) be minimized for the specified value of ω.

The value ω appears within the dual update formula of MALM. This is an important feature, so that MALM can converge to minimizers of (UQPP) for the specific value of ω.

C. Direct minimization of (UQPP) is numerically inefficient

To the unprejudiced it appears natural to minimize the unconstrained objective Φ_ω using a numerical method
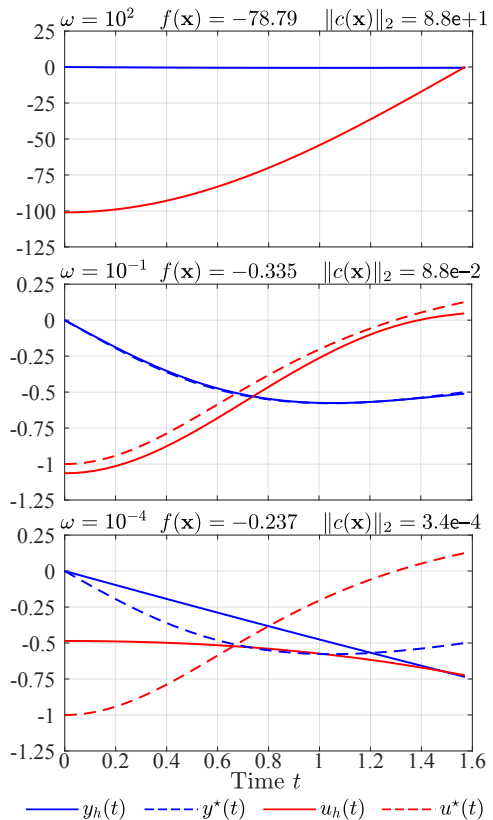

Fig. 1. Numerical solution to (OCP) for $N = 40$ and different values of ω.

for unconstrained minimization. However, (unless c is an affine function) this will result in very many iterations. This is so because the *nonlinear* penalties result in *curved* valleys through which iterative minimization methods make slow progress.

To demonstrate this inefficiency, consider the instance

$$f(\mathbf{x}) := -x_1 - x_2 \tag{3a}$$

$$c(\mathbf{x}) := \begin{bmatrix} (x_1 + \varepsilon)^2 + x_2^2 - 2 \\ (x_1 - \varepsilon)^2 + x_2^2 - 2 \end{bmatrix} \in \mathbb{R}^m \tag{3b}$$

with primal and dual initial guesses $\mathbf{x}_0 := \sqrt{2}[\cos(3\pi/8) \quad \sin(3\pi/8)]^\mathsf{T}$ and $\boldsymbol{\lambda}_0 := 0.4619[1 \quad 1]^\mathsf{T}$, for $\varepsilon = 0$. We discuss later with Table II that minimization of (UQPP) of (3) with a simple trust-region method in Algorithm 2 takes 334 iterations when $\omega = 10^{-6}$. This is inefficient when compared to MALM, which solves the same instance in only 16 iterations.

The Augmented Lagrangiam Method (ALM) uses a local minimization method (typically a quasi-Newton variant) for the primal variables, and then updates the duals. Since ALM eventually solves (ECP), the quasi-Newton system must solve (KKT) with $\omega = 0$. In contrast, MALM converges to minimizers of (UQPP), i.e. solves (KKT) with $\omega > 0$. This yields a dual regularization, which keeps the magnitude of $\|\boldsymbol{\lambda}\|$ bounded and improves the convergence of the Newton iteration.

D. Structure of the Paper

Section II derives the proposed modified ALM (MALM) for nonlinear functions f, c. Section III presents numerical experiments. This section also elaborates on the numerical difficulties of solving either (UQPP) or (ECP) and suitable values of ω for a given instance $\mathbf{x}_0, \boldsymbol{\lambda}_0, f, c$. The numerical experiments compare the efficiency in terms of computational cost and iteration count for the Quadratic Penalty Method (QPM), ALM, and MALM.

II. Derivation of the Modified Augmented Lagrangian Method

MALM is a solution method for (UQPP). MALM has been presented for minimizing convex quadratic penalty functions in [11], where f must satisfy certain convexity properties and c must be linear. Here, we derive MALM for nonlinear problems, in a stronger relation to its origins in ALM [5], [8].

We derive MALM for (UQPP) from ALM for (ECP). To apply ALM, we need an auxiliary problem of form (ECP) instead. Our approach to achieving this works by temporarily using an auxiliary variable $\boldsymbol{\xi} \in \mathbb{R}^m$. This variable will be eliminated later in the augmented optimality system.

A. Auxiliary Problem

Consider the following equivalent problem to (UQPP):

$$\min_{\hat{\mathbf{x}}:=(\mathbf{x}, \boldsymbol{\xi}) \in \mathbb{R}^{(n+m)}} \quad \hat{f}(\hat{\mathbf{x}}) := f(\mathbf{x}) + \frac{\omega}{2} \|\boldsymbol{\xi}\|_2^2 \quad (4a)$$

$$\text{subject to} \quad \hat{c}(\hat{\mathbf{x}}) := c(\mathbf{x}) + \omega \boldsymbol{\xi} = \mathbf{0} \,. \quad (4b)$$

Using Lagrange multipliers $\boldsymbol{\lambda} \in \mathbb{R}^m$, the optimality conditions of (4) are

$$\begin{bmatrix} \nabla f(\mathbf{x}) \\ \omega \boldsymbol{\xi} \end{bmatrix} - \begin{bmatrix} \nabla c(\mathbf{x}) \\ \omega \mathbf{I} \end{bmatrix} \boldsymbol{\lambda} = \mathbf{0} \quad (5a)$$

$$c(\mathbf{x}) + \omega \boldsymbol{\xi} = \mathbf{0} \,. \quad (5b)$$

B. Augmented Optimality System

Since (4) is of form (ECP), we can apply ALM [7, Alg. 17.3]. To this end, we augment (5) with an auxiliary vector $\mathbf{z} \in \mathbb{R}^m$ and a moderate penalty parameter $\rho > 0$:

$$\begin{bmatrix} \nabla f(\mathbf{x}) \\ \omega \boldsymbol{\xi} \end{bmatrix} - \begin{bmatrix} \nabla c(\mathbf{x}) \\ \omega \mathbf{I} \end{bmatrix} (\boldsymbol{\lambda} + \mathbf{z}) = \mathbf{0} \quad (6a)$$

$$c(\mathbf{x}) + \omega \boldsymbol{\xi} + \rho \mathbf{z} = \mathbf{0} \,. \quad (6b)$$

The intuition for doing so in ALM is similar to what we did in (KKT) for (UQPP), where the Lagrange multipliers were used as a substitute to ensure that the gradient of $\mathcal{L}$ matches that of Φ_ω. Likewise, here $\mathbf{z}$ works as a penalty substitute for (5b).

We could use (6) directly in order to form an ALM iteration. That iteration would consist of alternately solving the optimality system (6) for $(\mathbf{x}, \boldsymbol{\xi}, \mathbf{z})$ where $\boldsymbol{\lambda}$ is fixed and updating $\boldsymbol{\lambda} \leftarrow \boldsymbol{\lambda} + \mathbf{z}$, being equivalent to $\boldsymbol{\lambda} \leftarrow \boldsymbol{\lambda} - \frac{1}{\rho} (c(\mathbf{x}) + \omega \boldsymbol{\xi})$.

Algorithm 1 Modified Augmented Lagrangian Method

1: **procedure** MALM($f, c, \omega, \mathbf{x}_0, \boldsymbol{\lambda}_0, \mathsf{tol}$)
2: $\quad \rho \leftarrow \rho_0$
3: $\quad$ **for** $k = 1, 2, 3, \ldots, k_{\max}$ **do**
4: $\quad\quad$ Compute $\mathbf{x}_k$ by TRM($\Psi_k, \mathbf{x}_{k-1}, \mathsf{tol}$).

$$\mathbf{x}_k \leftarrow \operatorname*{argmin}_{\mathbf{x} \in \mathbb{R}^n} \Psi_k(\mathbf{x})$$

5: $\quad\quad$ Update $\boldsymbol{\lambda}_k \leftarrow \boldsymbol{\lambda}_{k-1} - \frac{1}{\omega+\rho} \left(c(\mathbf{x}_k) + \omega \boldsymbol{\lambda}_{k-1} \right)$
6: $\quad\quad$ **if** $\|c(\mathbf{x}_k) + \omega \boldsymbol{\lambda}_k\|_\infty \leq \mathsf{tol}$ **then**
7: $\quad\quad\quad$ **return** $\mathbf{x}_k, \boldsymbol{\lambda}_k$
8: $\quad\quad$ **else**
9: $\quad\quad\quad$ Decrease $\rho \leftarrow c_\rho \rho$ to promote convergence.
10: $\quad\quad$ **end if**
11: $\quad$ **end for**
12: **end procedure**

C. Elimination of the Auxiliary Vector

However, we propose to eliminate $\boldsymbol{\xi} = \boldsymbol{\lambda} + \mathbf{z}$ instead to obtain

$$\nabla f(\mathbf{x}) - \nabla c(\mathbf{x})(\boldsymbol{\lambda} + \mathbf{z}) = \mathbf{0} \quad (7a)$$

$$c(\mathbf{x}) + \omega \boldsymbol{\lambda} + (\omega + \rho)\mathbf{z} = \mathbf{0} \,. \quad (7b)$$

As in ALM, we solve (7) with an iteration of two alternating steps:

1) Keep the value of $\boldsymbol{\lambda}$ fixed, and solve (7) for $(\mathbf{x}, \mathbf{z})$.
2) Update $\boldsymbol{\lambda}$ as $\boldsymbol{\lambda} \leftarrow \boldsymbol{\lambda} + \mathbf{z}$.

Analogous to ALM, the first step can be realized by minimizing a suitable augmented Lagrangian function for $\mathbf{x}$ at fixed $\boldsymbol{\lambda}$, whereas in the second step $\mathbf{z}$ can be expressed in terms of $\mathbf{x}$ from (7b). Using this, the method can be expressed compactly in Algorithm 1, where

$$\Psi_k(\mathbf{x}) := \mathcal{L}(\mathbf{x}, \boldsymbol{\lambda}_{k-1}) + \frac{1}{2(\omega + \rho)} \|c(\mathbf{x}) + \omega \boldsymbol{\lambda}_{k-1}\|_2^2 \quad (8)$$

is the augmented Lagrangian function, with $\mathcal{L}(\mathbf{x}, \boldsymbol{\lambda}) = f(\mathbf{x}) - \boldsymbol{\lambda}^\mathsf{T} \cdot c(\mathbf{x})$ as in (KKT).

In our experiments we use $\mathsf{tol} = 10^{-8}, c_\rho = 0.1, \rho_0 = 0.1$. Care must be taken that the problem in line 4 is bounded below. To this end, practical methods use box constraints [3, eq. 3.2.2] or a trust-region [3, eq. 3.2.4].

In order to minimize (8), one can use any unconstrained minimization method. Here, for simplicity and reproducibility of the numerical experiments to follow, we use the simplified trust-region method in Algorithm 2. This method uses quasi-Newton directions $\mathbf{d}$ in line 7, where the shift is determined directly by a decrease condition instead of implicitly by a trust-region radius. If we had used a line search method instead, then a shift would have still been necessary due to non-convexity, which would interfere with the line search; hence why we opted against it.

As described in [7, eq. 17.21], the quasi-Newton direction for the quadratic penalty function can be computed

Algorithm 2 Simplified Trust-Region Method

1: **procedure** TRM($\varphi, \mathbf{x}_0, \mathsf{tol}$)
2:     $j \leftarrow 0$
3:     **while** $\|\nabla\varphi(\mathbf{x}_j)\|_\infty > \mathsf{tol}$ **do**
4:         $j \leftarrow j + 1, \qquad \sigma \leftarrow 10^{-11}$
5:         $\mathbf{H} \leftarrow \nabla^2\varphi(\mathbf{x}_{j-1}), \qquad \mathbf{g} \leftarrow \nabla\varphi(\mathbf{x}_{j-1})$
6:         **repeat**
7:           $\sigma \leftarrow 10\sigma, \qquad \mathbf{d} \leftarrow -(\mathbf{H} + \sigma\mathbf{I})^{-1}\mathbf{g}$
8:         **until** $\varphi(\mathbf{x}_{j-1} + \mathbf{d}) < \varphi(\mathbf{x}_{j-1})$
9:         $\mathbf{x}_j \leftarrow \mathbf{x}_{j-1} + \mathbf{d}$
10:    **end while**
11:    **return** $\mathbf{x}_j$
12: **end procedure**

Fig. 2. Level sets of components of c, and contours of f. The level sets of c_1 and c_2 are two circles, that almost fully overlap each other.

in a more numerically stable fashion from a 2×2 saddle-point linear equation system, by expressing the equations in terms of both $\mathbf{x}$ and $\mathbf{z}$. This has not been presented in the algorithms here for accessibility, but caused no issue in the numerical experiments with double precision and $\omega \geq 10^{-6}$.

D. Discussion

1) True Generalization of ALM: MALM is a true generalization of ALM, because it differs merely by the parameter ω. If $\omega = 0$ then MALM is identical to ALM. Both methods then enjoy the same convergence properties and approach the same limit point.

2) Benefit: MALM solves the penalty function Φ_ω in (UQPP) by minimizing a sequence of penalty functions Ψ_k. When does this make sense? By selecting $\rho \gg \omega$. Thereby, the penalty functions Ψ_k have less steep valleys and hence can often be minimized more efficiently in comparison to one minimization of Φ_ω. The numerical experiments in the next section verify this claim.

III. NUMERICAL EXPERIMENTS

We present two numerical test problems. The first is instructional, the second is an optimal control problem.

A. Circle Problem

1) Setting: This problem considers the instance (3) for various values of ε. Fig. 2 shows the geometry of the instance: Level sets of f and c are red and blue, respectively. The figure also shows two points

$$\mathbf{x}_A := [0 \quad \sqrt{2}]^\mathsf{T}, \qquad \mathbf{x}_B := [1 \quad 1]^\mathsf{T}$$

as the white and black star, respectively.

The instance can be interpreted in either of two ways:

(A) Either we meant c in a precise sense, meaning we wish to find a solution to $c(\mathbf{x}) = \mathbf{0}$ and, if non-unique, select the point that gives the smallest yield for f.

(B) Or we actually meant c in a rough sense, meaning we wish to minimize f subject to $\|\mathbf{x}\|_2^2 = 2 + \mathcal{O}(\varepsilon)$.

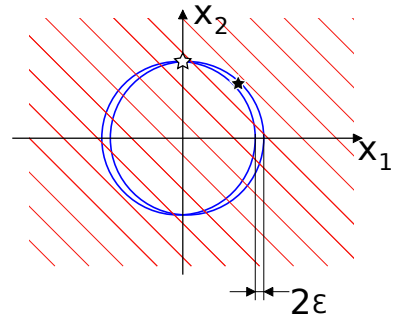Both problems are reasonable in their own right: For example, (A) makes sense when we have to solve a

complex equation system and want to find a desirable solution. On the other hand, (B) makes sense when our constraints suffer from errors, e.g. measurement errors or consistency errors, such as by discretization. For example, imagine a discretized optimal control problem, where c inherits consistency errors that have the size of ε.

Crucially, both solutions $\mathbf{x}_A, \mathbf{x}_B$ can be characterized sharply with a suitable problem statement. Obviously, $\mathbf{x}_A$ is the solution of (ECP). Less obvious, $\mathbf{x}_B$ can be computed as the solution of (UQPP) when choosing ω suitable w.r.t. ε. Here, a suitable choice is $\omega = \mathcal{O}(\varepsilon)$. To see this, notice that (KKT) admits a well-scaled solution $\|\mathbf{x}\|_2, \|\boldsymbol{\lambda}\|_2 = \mathcal{O}(1)$ and $\|c(\mathbf{x})\| = \mathcal{O}(\varepsilon)$ when this selection for ω is made.

Lastly, we stress that for this instance the solution $\mathbf{x}_A$ has an ill-conditioned KKT system with a dual solution $\|\boldsymbol{\lambda}\| = \mathcal{O}(1/\varepsilon^2)$, whereas $\mathbf{x}_B$ is well-behaved, i.e, its KKT equations are well-conditioned.

2) Computational Results: We solve the instance with MALM and QPM, for various values of ε, ω, including 0. We implement QPM by solving (UQPP) via Algorithm 2. Recall that MALM=ALM for $\omega = 0$ and that QPM is not applicable (n.a.) when $\omega = 0$.

Further, we investigate the limit points $\mathbf{x}_\infty$ (which are identical for both tested methods throughout all tests) for each ε, ω, by measuring the quantities

$$e_A := \|\mathbf{x}_\infty - \mathbf{x}_A\|_2, \qquad e_B := \|\mathbf{x}_\infty - \mathbf{x}_B\|_2.$$

Table I shows the quantities e_A, e_B for respective ε, ω. Dividing the table into a lower left and an upper right triangle, we see that solutions in the lower triangle rather converge to $\mathbf{x}_A$ while those on the diagonal and in the upper right converge to $\mathbf{x}_B$.

Table II shows the sum of the number of all inner iterations (i.e. iterations j of Algorithm 2) of QPM and MALM for respective ε, ω. We see a trend for each of the two methods: QPM converges in a few iterations when ω is moderate. However, when ε, ω both decrease, the iteration count blows up. The trend for MALM is different. MALM converges reliably for all ε, ω in the upper right triangle, including those where ε, ω are very

TABLE I

SOLUTION OF THE CIRCLE PROBLEM W.R.T. ε, ω. SMALLER VALUES MEAN CLOSER CONVERGENCE TO EITHER POINT. TABLE CELLS IN THE LOWER LEFT CONVERGE TO $\mathbf{x}_A$, CELLS IN THE UPPER RIGHT TO $\mathbf{x}_B$.

	e_A / e_B	ε				
		1.0e–1	1.0e–2	1.0e–4	1.0e–6	0e+0
ω	1.0e–1	6.5e–1 / 4.6e–1	1.1e+0 / 1.0e–2	1.1e+0 / 8.8e–3	1.1e+0 / 8.8e–3	1.1e+0 / 8.8e–3
	1.0e–2	1.2e–1 / 9.7e–1	1.0e+0 / 5.6e–2	1.1e+0 / 8.8e–4	1.1e+0 / 8.8e–4	1.1e+0 / 8.8e–4
	1.0e–4	3.7e–3 / 1.1e+0	1.1e–1 / 9.8e–1	1.1e+0 / 5.7e–4	1.1e+0 / 8.8e–6	1.1e+0 / 8.8e–6
	1.0e–6	3.5e–3 / 1.1e+0	1.2e–3 / 1.1e+0	1.0e+0 / 5.6e–2	1.1e+0 / 5.7e–6	1.1e+0 / 8.8e–8
	0e+0	3.5e–3 / 1.1e+0	3.5e–5 / 1.1e+0	3.5e–9 / 1.1e+0	3.5e–13 / 1.1e+0	1.1e+0 / 0e+0

TABLE II

NUMBER OF ITERATIONS FOR MALM/ALM AND QPM FOR THE CIRCLE PROBLEM W.R.T. ε, ω. FEWER ITERATIONS MEAN BETTER COMPUTATIONAL EFFICIENCY. MALM WITH $\omega = 0$ (I.E. ALM) SOMETIMES DOES NOT CONVERGE (N.C.) FOR THIS PROBLEM. QPM IS NOT APPLICABLE (N.A.) WHEN $\omega = 0$.

	#MALM / #QPM	ε				
		1.0e–1	1.0e–2	1.0e–4	1.0e–6	0e+0
ω	1.0e–1	22 / 6	19 / 9	18 / 9	16 / 9	16 / 9
	1.0e–2	25 / 9	26 / 12	19 / 13	18 / 13	16 / 13
	1.0e–4	25 / 10	83 / 30	28 / 77	21 / 77	16 / 77
	1.0e–6	26 / 10	83 / 35	97 / 311	30 / 334	16 / 334
	0e+0	26 / n. a.	83 / n. a.	n. c. / n. a.	n. c. / n. a.	16 / n. a.

TABLE III

SOLUTION OF THE OPTIMAL CONTROL PROBLEM W.R.T. N, ω. FOR A GIVEN MESH SIZE N, THE VALUE FOR ω IS SUITABLE WHEN IT YIELDS A GOOD BALANCE BETWEEN THE ORDERS OF MAGNITUDE FOR BOTH δJ (OPTIMALITY GAP) AND r (FEASIBILITY RESIDUAL). TABLE CELLS ON AND AROUND THE DIAGONAL TYPICALLY YIELD A GOOD BALANCE.

	δJ / r	N				
		16	64	256	1024	4096
ω	1.0e–1	-7.8e–2 / 9.2e–2	-7.9e–2 / 8.9e–2	-7.9e–2 / 8.9e–2	-7.9e–2 / 8.9e–2	-7.9e–2 / 8.9e–2
	2.5e–2	-1.5e–2 / 2.7e–2	-2.0e–2 / 2.3e–2	-2.0e–2 / 2.2e–2	-2.0e–2 / 2.2e–2	-2.0e–2 / 2.2e–2
	6.4e–3	7.6e–3 / 8.8e–3	-4.3e–3 / 8.4e–3	-5.0e–3 / 6.0e–3	-5.0e–3 / 5.7e–3	-5.0e–3 / 5.7e–3
	1.6e–3	1.7e–2 / 2.5e–3	3.3e–3 / 4.3e–3	-1.2e–3 / 2.3e–3	-1.3e–3 / 1.5e–3	-1.3e–3 / 1.4e–3
	4.0e–4	2.0e–2 / 6.5e–4	1.2e–2 / 1.7e–3	3.7e–4 / 1.6e–3	-3.1e–4 / 5.9e–4	-3.1e–4 / 3.7e–4
	0e+0	2.6e–1 / 0e+0	2.6e–1 / 0e+0	2.6e–1 / 0e+0	2.6e–1 / 0e+0	2.6e–1 / 0e+0

B. Optimal Control Problem

1) Setting: We solve the instance (2) for various values of N, ω for $\mathbf{x}_0 = \mathbf{0}, \boldsymbol{\lambda}_0 = \mathbf{0}$. Recall that the instance represents the discretization (1) with mesh size $h = \frac{\pi}{2N}$, which only converges to the analytical solution when $h, \omega \searrow +0$ together, as was discussed along Fig. 1.

As for the former experiment, we solve the instance with MALM and QPM, for various values of N, ω, including $\omega = 0$. We also investigate the limit points $\mathbf{x}_\infty$ (which are identical for both tested methods throughout all tests) for each N, ω, by measuring the quantities

$$\delta J := f(\mathbf{x}) - J^\star \equiv J(y_h, u_h) - J^\star$$
$$r := \|c(\mathbf{x})\|_2 \equiv \left\| -\dot{y}_h + \frac{1}{2}y_h^2 + u_h \right\|_{L^2(0,\pi/2)} .$$

Recall from the circle test problem that there are two interpretations (A) and (B) for the instance. As is clear from the context, we wish to compute a solution of kind (B). However, if we are (deliberately) uncareful in choosing ω suitable w.r.t. h then the iteration will converge to a solution of kind (A).

2) Computational Results: Table III shows the quantities $\delta J, r$ for respective N, ω. Dividing the table into a lower left and an upper right triangle, we see that solutions in the lower left achieve good feasibility but at the sacrifice of optimality, whereas solutions in the upper right are not sufficiently feasible. Only solutions on the diagonal strike a good balance between minimizing J and r, i.e. optimality and feasibility. Thus, when being limited by computation time to solve on a moderately sized mesh, then accordingly ω should not be chosen too small. Therefore, we must consider the table column by column. The table clearly shows that the best balance between feasibility and optimality is obtained in those table entries that live on the diagonal.

small. The last row shows ALM. ALM converges quickly when $\varepsilon = 0$, but its iteration count blows up for positive decreasing values of ε. In two instances ALM did not converge (n.c.) within 1000 iterations.

3) Interpretation of the Results: Table I confirms that, depending on the parameters ε, ω, we either solve for $\mathbf{x}_A$ or $\mathbf{x}_B$. Table II indicates that $\mathbf{x}_A$ cannot be computed numerically efficiently, as expected due to the almost linearly dependent constraints.

In the optimal control context we are interested in solving problems in the sense of (B), hence we now investigate the upper right triangle in Table II. We see that both methods converge for all of these instances. Yet, when ε, ω are small, but strictly positive, then MALM outperforms QPM. This is relevant because there are problems from optimal control discretizations with inconsistencies like ε from discretization errors. In these cases, we wish to drive $\omega, h \searrow +0$ to yield convergence of the discretization, which brings us into the lower right region of the table.

TABLE IV

Number of iterations for MALM/ALM and QPM for the Optimal Control Problem w.r.t. N, ω. Fewer iterations mean better computational efficiency. MALM with $\omega = 0$ (i.e. ALM) does not converge (n.c.) for this problem. QPM is not applicable (n.a.) when $\omega = 0$.

$\#$MALM $\#$QPM		N				
		16	64	256	1024	4096
ω	1.0e−1	16	16	14	12	10
		7	7	7	7	7
	2.5e−2	22	19	16	14	13
		7	12	10	10	10
	6.4e−3	23	20	19	18	14
		11	13	20	19	19
	1.6e−3	25	23	20	20	17
		12	9	42	67	101
	4.0e−4	28	27	23	23	20
		16	19	30	68	60
	0e+0	n. c.	n. c.	n. c.	n. c.	n. c.
		n. a.	n. a.	n. a.	n. a.	n. a.

Table IV shows the sum of the number of all inner iterations of MALM and QPM for respective N, ω. We see the same trend as for the circle problem: QPM converges in a few iterations when ω is moderate. In contrast, when ε, ω both decrease then its iteration count increases. In contrast, MALM converges reliably for all N, ω in the upper right triangle, including those where N is very large and ω very small. The last row shows that ALM does not convergence (n.c.) within 150 iterations for any mesh size.

C. Interpretation of the Results

For this test problem, Table III demonstrates that the numerical solution converges to the optimal control solution when $h, \omega \searrow +0$ together.

Table IV shows that QPM converges fast when N is small and ω is moderate. In contrast to this, for large N and small ω, MALM is clearly more efficient. However, Table III reveals that large N and small ω are a necessity for the numerical computation of accurate optimal control solutions.

Importantly, both experiments make clear that ALM is unsuitable for solving applications with inconsistent constraints: For the experiment depicted in Fig. 2, ALM converges to $\mathbf{x}_A$ (white star) whenever $\varepsilon > 0$. For optimal control problems, the magnitude of ε models consistency errors of discretizations, in which case $\mathbf{x}_B$ is the sought solution of a well-conditioned problem, whereas $\mathbf{x}_A$ is the unsought solution of an ill-conditioned problem. Table IV shows that ALM fails to converge for the discretized control problem because it attempts to seek $\mathbf{x}_A$, which is

numerically hard. Table III shows further that the exact minimizer is undesired here because $\omega = 0$ results in a bad balance between the goals of minimizing both r and δJ, illustrated in Fig. 1.

IV. Conclusions

We derived a modified version of ALM, called MALM. MALM outperforms QPM when minimizing unconstrained quadratic penalty programs (UQPP) when ω is very small, in a similar manner as ALM outperforms QPM when solving equality-constrained programs (ECP).

The efficiency of MALM for the minimization of quadratic penalty functions has been demonstrated with numerical experiments. These experiments show that there are problem instances where it is beneficial to solve a problem of type (UQPP) rather than (ECP), one important class arising from integral penalty discretizations of optimal control problems.

In this paper we have presented the method in isolated form for "approximately" equality constrained programs in the sense that $c(\mathbf{x}) \approx \mathbf{0}$. Future work could extend the approach to problems with both equality and inequality constraints. Extensions of ALM for inequality constraints have been proposed in [9], [10], which can form a basis for similar extensions of MALM.

References

[1] A. V. Balakrishnan. On a new computing technique in optimal control. *SIAM J. Control*, 6:149–173, 1968.

[2] J. T. Betts. *Practical Methods for Optimal Control and Estimation Using Nonlinear Programming.* Cambridge University Press, New York, NY, USA, 2nd edition, 2010.

[3] A. R. Conn, N. I. M. Gould, and Ph. L. Toint. *LANCELOT*, volume 17 of *Springer Series in Computational Mathematics.* Springer-Verlag, Berlin, 1992. A Fortran package for large-scale nonlinear optimization (release A).

[4] W. W. Hager. Multiplier methods for nonlinear optimal control. *SIAM J. Numer. Anal.*, 27(4):1061–1080, 1990.

[5] M. R. Hestenes. Multiplier and gradient methods. *J. Optim. Theory Appl.*, 4:303–320, 1969.

[6] M. P. Neuenhofen and E. C. Kerrigan. Dynamic optimization with convergence guarantees. *arXiv:1810.04059*, 2018.

[7] J. Nocedal and S. J. Wright. *Numerical optimization.* Springer Series in Operations Research and Financial Engineering. Springer, New York, second edition, 2006.

[8] M. J. D. Powell. A method for nonlinear constraints in minimization problems. In *Optimization (Sympos., Univ. Keele, Keele, 1968)*, pages 283–298. Academic Press, London, 1969.

[9] M. J. D. Powell. On search directions for minimization algorithms. *Math. Programming*, 4:193–201, 1973.

[10] R. T. Rockafellar. The multiplier method of Hestenes and Powell applied to convex programming. *J. Optim. Theory Appl.*, 12:555–562, 1973.

[11] M.H.B.M. Shariff and J.R. Dormand. A modified augmented Lagrangian method for a class of constrained problems. *Journal of Computational and Applied Mathematics*, 151(2):257 – 270, 2003.