Integration of Sequential Quadratic Programming and Domain Decomposition Methods for Nonlinear Optimal Control Problems

Matthias Heinkenschloss[1] and Denis Ridzal[2]

[1] Department of Computational and Applied Mathematics, MS-134, Rice University, 6100 Main Street, Houston, TX 77005-1892, USA. heinken@rice.edu
[2] Computational Mathematics and Algorithms, MS-1320, Sandia National Laboratories †, P.O. Box 5800, Albuquerque, NM 87185-1320, USA. dridzal@sandia.gov

Summary. We discuss the integration of a sequential quadratic programming (SQP) method with an optimization-level domain decomposition (DD) preconditioner for the solution of the quadratic optimization subproblems. The DD method is an extension of the well-known Neumann-Neumann method to the optimization context and is based on a decomposition of the first order system of optimality conditions. The SQP method uses a trust-region globalization and requires the solution of quadratic subproblems that are known to be convex, hence solving the first order system of optimality conditions associated with these subproblems is equivalent to solving these subproblems. In addition, our SQP method allows the inexact solution of these subproblems and adjusts the level of exactness with which these subproblems are solved based on the progress of the SQP method. The overall method is applied to a boundary control problem governed by a semilinear elliptic equation.

1 Introduction

Optimization algorithms for PDE constrained optimization problems which use second order derivative information require the solution of large-scale linear systems that involve linearizations of the governing PDE and its adjoint. Domain decomposition methods can be used to effectively solve these subproblems. In this paper we discuss the integration of a sequential quadratic programming (SQP) method with an optimization-level domain decomposition (DD) preconditioner for the quadratic optimization subproblems arising inside the SQP method.

† Sandia is a multiprogram laboratory operated by Sandia Corporation, a Lockheed Martin Company, for the United States Department of Energy's National Nuclear Security Administration under Contract DE-AC04-94-AL85000.

As an example problem we consider the following boundary control problem with states y and controls u.

$$\text{Minimize } \frac{1}{2}\int_{\Omega} l(y(x),x)dx + \frac{\alpha}{2}\int_{\partial\Omega_c} u^2(x)dx \tag{1a}$$

subject to

$$-\epsilon\Delta y(x) + g(y(x),x) = 0, \qquad x \in \Omega, \tag{1b}$$

$$y(x) = 0, \qquad x \in \partial\Omega \setminus \partial\Omega_c, \tag{1c}$$

$$\epsilon\frac{\partial}{\partial \mathbf{n}} y(x) = u(x), \qquad x \in \partial\Omega_c. \tag{1d}$$

Here $\alpha > 0$ is a given parameter. Because of page restrictions, we limit our presentation to semilinear elliptic optimal control problems in which the functions g, l and the problem data are such that the optimal control problem (1) has a solution $y \in H^1(\Omega)$, $u \in L^2(\partial\Omega_c)$. Furthermore, we assume that the state equation and the objective functional are twice Fréchet differentiable in $H^1(\Omega) \times L^2(\partial\Omega_c)$, that the linearized state equation has a unique solution in $H^1(\Omega)$ that depends continuously on the right hand side and boundary data, and that a second order sufficient optimality condition is satisfied at the solution. These assumptions are satisfied for the example problem considered in Section 4 as well as those discussed, e.g., in [8, 11, 10, 16]. To establish Fréchet differentiability and second order optimality conditions for other semilinear elliptic optimal control problems, however, a more involved setting and analysis is required. See, e.g., [17, 24]. Our approach can be adapted to many of those problems. We note that our approach can also be applied to the optimal control of incompressible Navier-Stokes equations. However, since these are systems of PDEs and because the compatibility conditions that are implied by the incompressibility condition require a careful treatment, the presentation of our approach for the optimal control of incompressible Navier-Stokes equations is too lengthy and will be given elsewhere.

In this work we use the optimization-level DDM introduced in [3, 13] for the solution of convex quadratic subproblems arising in the solution of (1). These optimization-level DDMs are extensions of the well known Neumann-Neumann methods (see, e.g., [20, 22, 23]) or the Robin-Robin methods for problems with advection (see, e.g., [1, 2]) from the PDE to the optimization context. In particular, all subproblem solves that arise in our DDM correspond to the solution of subdomain optimal control problems, which are essentially smaller copies of the original one. We note that our DDM is not the only optimization-level DDM. By 'optimization-level' we mean that the DDM is applied directly to the optimization problem, not individually to the state and adjoint PDEs. For example the DDM used in [18, 19] may be viewed as the optimization-level version of the restrictive additive Schwarz method discussed, e.g., in [6]. Heinkenschloss and Nguyen [12] analyze an optimization-level additive Schwarz method. Overall, however, the theoretical

properties of optimization-level DDMs are still relatively poorly understood. We also point out that many optimization-level DDMs, including ours and the ones in [18, 19] are obtained by applying DDM to the system of optimality conditions, the so-called KKT system. This is only possible if the system of optimality conditions is necessary and sufficient, i.e., if the optimization problem is convex. This restriction is not always made explicit enough and is typically important for nonlinear PDE constrained optimization problems.

SQP algorithms coupled with DDMs have been discussed in [4, 5, 18, 19].

Our SQP method builds on the works [15, 21]. There are important features that distinguish our SQP from those in [4, 5, 18, 19]. First, all quadratic subproblems that arise in our SQP method are known a-priori to be convex. This allows us to apply optimization-level DDMs to these subproblems, which are based on a decomposition of the first order optimality conditions, the so-called KKT conditions. Since our subproblems are convex, solving these optimality systems is equivalent to solving the quadratic optimization problems. Secondly, we allow the inexact solution of the large scale linear KKT systems that arise as subproblems inside the SQP algorithms, and provide a rigorous way to control the level of inexactness with which these systems have to be solved. The level of inexactness is coupled to the progress of the SQP algorithm, which enables us to apply coarse, more inexpensive solves away from the solution. Our DDM is used as a preconditioner for the large scale linear KKT systems that arise in the SQP algorithm. Other preconditioners could be used as well. In particular, it is possible to incorporate the DD Krylov-Schur preconditioner used by [4, 5] or (restricted) additive Schwarz preconditioners as used by [18, 19].

2 Optimal Control of Advection Diffusion Equations

We begin with a discussion of our DD approach for convex linear-quadratic optimal control problems governed by an advection diffusion equation. The example problem is given as follows.

$$\text{Minimize } \frac{1}{2}\int_{\Omega}(y(x)-\widehat{y}(x))^2dx+\frac{\alpha}{2}\int_{\partial\Omega_c}u^2(x)dx \tag{2a}$$

subject to

$$-\epsilon\Delta y(x)+\mathbf{a}(x)\cdot\nabla y(x)+r(x)y(x)=f(x), \qquad x\in\Omega, \tag{2b}$$

$$y(x)=0, \qquad x\in\partial\Omega\setminus\partial\Omega_c, \tag{2c}$$

$$\epsilon\frac{\partial}{\partial\mathbf{n}}y(x)=u(x), \qquad x\in\partial\Omega_c, \tag{2d}$$

where $\partial\Omega_c$ is the control boundary, $\mathbf{a}, f, g, r, \widehat{y}$ are given functions, $\epsilon, \alpha > 0$ are given scalars, and $\mathbf{n}$ denotes the outward unit normal. Our main interest is not in this particular optimal control problem. As we will see in more detail later,

our SQP method applied to (1) requires the repeated solution of convex linear-quadratic optimal control subproblems governed by linear elliptic PDEs. The governing PDEs in these linear-quadratic subproblems are of the form (2b-d), with $\mathbf{a}$, r, and f determined by the SQP algorithm. The objective function in these subproblems is slightly different from (2a) and is given by a quadratic model of the Lagrangian associated with (1). However, the problem structure of the SQP subproblems and that of (2) are close enough so that a study of (2) reveals how to deal with the subproblems arising in our SQP method for (1).

The system of necessary and sufficient optimality conditions for (2) is given by the adjoint equations

$$\begin{aligned} -\epsilon\Delta p(x) - \mathbf{a}(x)\cdot\nabla p(x) & \\ +(r(x) - \nabla\cdot\mathbf{a}(x))p(x) = -(y(x)-\hat{y}(x)), & \qquad x\in\Omega, \quad (3a)\\ p(x) = 0, & \qquad x\in\partial\Omega\setminus\partial\Omega_c, \quad (3b)\\ \epsilon\frac{\partial}{\partial\mathbf{n}}p(x) + \mathbf{a}(x)\cdot\mathbf{n}(x)\; p(x) = 0, & \qquad x\in\partial\Omega_c, \quad (3c) \end{aligned}$$

by the equation

$$p(x) = \alpha u(x) \qquad x\in\Omega_c, \tag{4}$$

and by the state equation (2b-d).

We apply DD to the system of optimality conditions (2b-d), (3), (4). For simplicity, we consider the two-subdomain case only. Everything can be extended to more than two subdomains following the discussions in [3, 13]. We decompose Ω into two subdomains Ω_1, Ω_2 with interface $\Gamma = \overline{\Omega_1}\cap\overline{\Omega_2}$. The outer unit normal for subdomain i is denoted by $\mathbf{n}_i$. By γ_Γ we denote the trace operator and we define $V_\Gamma = \{\gamma_\Gamma v \;:\; v\in H^1(\Omega),\; v = 0 \text{ on } \partial\Omega\setminus\partial\Omega_c\}$ We now split (2b-d), (3), (4) as follows. Given $y_\Gamma, p_\Gamma \in V_\Gamma$ and $i\in\{1,2\}$ we consider the system

$$\begin{aligned} -\epsilon\Delta y_i(x) + \mathbf{a}(x)\cdot\nabla y_i(x) + r(x)y_i(x) = f(x) & \qquad \text{in } \Omega_i, \quad (5a)\\ y_i(x) = 0 & \qquad \text{on } \partial\Omega_i\cap\partial\Omega\setminus\partial\Omega_c, \quad (5b)\\ \epsilon\frac{\partial}{\partial\mathbf{n}}y_i(x) = u_i(x), & \qquad \text{on } \partial\Omega_i\cap\partial\Omega_c, \quad (5c)\\ y_i(x) = y_\Gamma(x) & \qquad \text{on } \Gamma, \quad (5d) \end{aligned}$$

$$\begin{aligned} -\epsilon\Delta p_i(x) - \mathbf{a}(x)\cdot\nabla p_i(x) & \\ +(r(x) - \nabla\cdot\mathbf{a}(x))p_i(x) = -(y_i(x)-\hat{y}(x)) & \qquad \text{in } \Omega_i, \quad (5e)\\ p_i(x) = 0, & \qquad \text{on } \partial\Omega_i\cap\partial\Omega\setminus\partial\Omega_c, \quad (5f)\\ \epsilon\frac{\partial}{\partial\mathbf{n}}p_i(x) + \mathbf{a}(x)\cdot\mathbf{n}(x)\; p_i(x) = 0, & \qquad \text{on } \partial\Omega_i\cap\partial\Omega_c, \quad (5g)\\ p_i(x) = p_\Gamma(x) & \qquad \text{on } \Gamma, \quad (5h) \end{aligned}$$

$$\alpha u_i(x) - p_i(x) = 0 \qquad \text{on } \partial\Omega_c \cap \partial\Omega_i. \tag{5i}$$

The system (5) together with the interface conditions

$$\begin{aligned} \left(\epsilon\tfrac{\partial}{\partial \mathbf{n}_i} - \tfrac{1}{2}\mathbf{a}(x)\mathbf{n}_i\right) y_i(x) &= -\left(\epsilon\tfrac{\partial}{\partial \mathbf{n}_j} - \tfrac{1}{2}\mathbf{a}(x)\mathbf{n}_j\right) y_j(x) \; x \in \partial\Omega_i \cap \partial\Omega_j, \\ \left(\epsilon\tfrac{\partial}{\partial \mathbf{n}_i} + \tfrac{1}{2}\mathbf{a}(x)\mathbf{n}_i\right) p_i(x) &= -\left(\epsilon\tfrac{\partial}{\partial \mathbf{n}_j} + \tfrac{1}{2}\mathbf{a}(x)\mathbf{n}_j\right) p_j(x) \; x \in \partial\Omega_i \cap \partial\Omega_j, \end{aligned} \tag{6}$$

on Γ are equivalent to the original optimality system (2b-d), (3), (4).

It can be shown that for given $y_\Gamma, p_\Gamma \in V_\Gamma$ the system (5) has a unique solution (y_i, p_i, u_i). If we view (y_i, p_i, u_i), $i = 1, 2$, as a function of $y_\Gamma, p_\Gamma \in V_\Gamma$ defined through (5), then (6) becomes an equation in y_Γ, p_Γ. Since $y_\Gamma, p_\Gamma \in V_\Gamma$, $i = 1, 2$, depends on y_Γ, p_Γ in an affine linear way, (5), (6) can be written as

$$(S_1 + S_2) \begin{pmatrix} y_\Gamma \\ p_\Gamma \end{pmatrix} = r_1 + r_2, \tag{7}$$

where S_i, $i = 1, 2$, is applied to y_Γ, p_Γ by first solving (5) with $f = 0$ and then evaluating $\left(\epsilon\frac{\partial}{\partial \mathbf{n}_i} - \frac{1}{2}\mathbf{a}(x)\mathbf{n}_i\right) y_i(x)$, $\left(\epsilon\frac{\partial}{\partial \mathbf{n}_i} + \frac{1}{2}\mathbf{a}(x)\mathbf{n}_i\right) p_i(x)$. The right hand side is computed by solving (5) with $y_\Gamma = p_\Gamma = \mathbf{0}$ and then evaluating $\left(\epsilon\frac{\partial}{\partial \mathbf{n}_i} - \frac{1}{2}\mathbf{a}(x)\mathbf{n}_i\right) y_i(x)$, $\left(\epsilon\frac{\partial}{\partial \mathbf{n}_i} + \frac{1}{2}\mathbf{a}(x)\mathbf{n}_i\right) p_i(x)$.

One can show that (5) is the system of optimality conditions for a subdomain optimal control problem that is essentially a restriction of (2) to subdomain Ω_i, but with the additional interface boundary condition (5d) and with an additional interface normal derivative term in the objective that leads to (5h). See [3, 13].

One can also show that the subdomain operators S_i, $i = 1, 2$, are invertible and that

$$S_i^{-1} \begin{pmatrix} r_\Gamma^u \\ r_\Gamma^\lambda \end{pmatrix} = \begin{pmatrix} \gamma_\Gamma y_i \\ \gamma_\Gamma p_i \end{pmatrix},$$

where γ_Γ denotes the trace operator and where y_i, p_i are obtained by solving

$$-\epsilon\Delta y_i(x) + \mathbf{a}(x) \cdot \nabla y_i(x) + r(x) y_i(x) = 0 \qquad \text{in } \Omega_i, \tag{8a}$$

$$y_i(x) = 0 \qquad \text{on } \partial\Omega_i \cap \partial\Omega \setminus \partial\Omega_c, \tag{8b}$$

$$\epsilon\frac{\partial}{\partial \mathbf{n}} y_i(x) = u_i(x), \qquad \text{on } \partial\Omega_i \cap \partial\Omega_c, \tag{8c}$$

$$\epsilon\frac{\partial}{\partial \mathbf{n}_i} y_i(x) - \tfrac{1}{2}\mathbf{a}(x) \cdot \mathbf{n}_i y_i(x) = r_i^y(x) \qquad \text{on } \Gamma, \tag{8d}$$

$$
\begin{aligned}
&-\epsilon\Delta p_i(x) - \mathbf{a}(x)\cdot\nabla p_i(x) \\
&\quad +(r(x) - \nabla\cdot\mathbf{a}(x))p_i(x) = -(y_i(x) - \hat{y}(x)) && \text{in } \Omega_i, && \text{(8e)}\\
&\qquad p_i(x) = 0, && \text{on } \partial\Omega_i \cap \partial\Omega \setminus \partial\Omega_c, && \text{(8f)}\\
&\epsilon\frac{\partial}{\partial\mathbf{n}}p_i(x) + \mathbf{a}(x)\cdot\mathbf{n}(x)\; p_i(x) = 0, && \text{on } \partial\Omega_i \cap \partial\Omega_c, && \text{(8g)}\\
&\epsilon\frac{\partial}{\partial\mathbf{n}_i}p_i(x) + \tfrac{1}{2}\mathbf{a}(x)\cdot\mathbf{n}_i p_i(x) = r_i^p(x) && \text{on } \Gamma, && \text{(8h)}\\
&\qquad \alpha u_i(x) - p_i(x) = 0 && \text{on } \partial\Omega \cap \partial\Omega_i. && \text{(8i)}
\end{aligned}
$$

See [3, 13]. One can show that (8) is the system of optimality conditions for a subdomain optimal control problem that is essentially a restriction of (2) to subdomain Ω_i, but with the additional interface boundary condition (8d) and with an additional interface boundary term in the objective that involves $y_i r_\Gamma^\lambda$ which leads to (8h).

We solve (7) using a preconditioned Krylov subspace method such as GMRES or sQMR with preconditioner $S_1^{-1} + S_2^{-1}$. As we have mentioned earlier, everything can be extended to the case of many subdomains. See [3, 13]. One can show that the discrete versions of S_i are Schur complements. They are symmetric and highly indefinite. The number of positive and negative eigenvalues is proportional to the number of discretized states y_i on the interface. While the observed performance of these methods is comparable to that of Neumann-Neumann (Robin-Robin) methods for elliptic PDEs, there is no theoretical explanation for this observed behavior in the optimization case yet.

3 Inexact Trust-Region-SQP Method

Many nonlinear optimal control problems can abstractly be written as a nonlinear programming problem (NLP) in Hilbert space,

$$
\begin{aligned}
&\min\;\; f(x) && \text{(9a)}\\
&\text{s.t.}\;\; c(x) = 0, && \text{(9b)}
\end{aligned}
$$

where $f : \mathcal{X} \to \mathbb{R}$ and $c : \mathcal{X} \to \mathcal{Y}$ for some Hilbert spaces $\mathcal{X}$ and $\mathcal{Y}$. In our example problem (1) we have $x = (y, u)$, $\mathcal{X} = H^1(\Omega) \times (L^2(\partial\Omega_c))^2$ and $\mathcal{Y} = (H^1(\Omega))'$, where $'$ is used to denote the dual, and $c(x) = 0$ represents the weak formulation of the semilinear elliptic equations (1b-d). The corresponding Lagrangian functional $L : \mathcal{X} \times \mathcal{Y} \to \mathbb{R}$ is given by $L(x, \lambda) = f(x) + \langle\lambda, c(x)\rangle_{\mathcal{Y}}$. We use subscript x to denote Fréchet derivatives with respect to x. Given estimates x_k, λ_k for the solution of (9) and corresponding Lagrange multiplier, SQP methods approximately solve

$$
\begin{aligned}
&\min\;\; \frac{1}{2}\langle H_k s_k, s_k\rangle_{\mathcal{X}} + \langle\nabla_x L(x_k, \lambda_k), s_k\rangle_{\mathcal{X}} && \text{(10a)}\\
&\text{s.t.}\;\; c_x(x_k)s_k + c(x_k) = 0 && \text{(10b)}
\end{aligned}
$$

and use the solution s_k to obtain a better approximation of the solution of (9). In (9) H_k is the Hessian $\nabla_{xx}L(x_k, \lambda_k)$ of the Lagrangian or a replacement thereof, obtained, e.g., using a quasi-Newton method. If x_k is sufficiently close to the solution and if a second order sufficiency condition is satisfied at the solution, then $x_{k+1} = x_k + s_k$ can be used at the new iterate. To ensure global convergence and to deal with possible negative curvature of H_k when x_k is away from the solution, we add a trust-region constraint $\|s_k\|_{\mathcal{X}} \leq \Delta_k$ to (10), where $\Delta_k > 0$ is the trust-region radius, which is adapted by the optimization algorithm. To deal with the possible incompatibility of the trust-region constraint and (10b), we use a composite step algorithm (see [7, Ch. 15] for an overview). The trial step s_k is decomposed as $s_k = n_k + t_k$, where for a given parameter $\xi \in (0, 1)$, the so-called quasi-normal step n_k is an approximate solution of

$$\min \ \|c_x(x_k)n + c(x_k)\|_{\mathcal{Y}} \tag{11a}$$

$$\text{s.t.} \ \|n\|_{\mathcal{X}} \leq \xi\Delta_k, \tag{11b}$$

and the so-called tangential step t_k is an approximate solution of

$$\min \ \frac{1}{2}\langle H_k t, t\rangle_{\mathcal{X}} + \langle \nabla_x L(x_k, \lambda_k) + H_k n_k, t\rangle_{\mathcal{X}} \tag{12a}$$

$$\text{s.t.} \ c_x(x_k)t = 0 \tag{12b}$$

$$\|t\|_{\mathcal{X}} \leq \Delta_k - \|n_k\|_{\mathcal{X}}. \tag{12c}$$

Once the trial step $s_k = n_k + t_k$ is computed, an augmented Lagrangian merit function and a quadratic approximation of it are used to decide whether to accept the trial step, i.e, set $x_{k+1} = x_k + s_k$, or to reject it, i.e., set $x_{k+1} = x_k$, and how to update the trust-region radius. The rules are fairly easy to implement, but their precise description is lengthy. Because of page limitations, we refer to [15, 21] for the details and instead focus on the issue of linear system solves that relates to the use of DD methods.

One way to compute an approximate solution of the quasi-normal step subproblem (11) that is suitable for use within our SQP method is the so-called dog-leg approach, which requires the computation of the minimum norm solution of $\min \|c_x(x_k)n + c(x_k)\|_{\mathcal{Y}}$. The minimum norm solution can be computed by solving

$$\begin{pmatrix} I & c_x(x_k)^* \\ c_x(x_k) & 0 \end{pmatrix} \begin{pmatrix} n \\ y \end{pmatrix} = \begin{pmatrix} 0 \\ -c(x_k) \end{pmatrix} \tag{13}$$

for $y \in \mathcal{Y}$, $n \in \mathcal{X}$. The quasi-normal step is then computed as a linear combination of the minimum norm solution n and of $-c_x(x_k)^* c(x_k)$ or by a simple scaling of the minimum norm solution. For a detailed description of the quasi-normal step computation see, e.g., [7, Sec. 15.4.1.2], [21]. In our context it is important to note that the quasi-normal step computation requires the solution of (13), which in our example application (1) leads to a subproblem of the type (2). Note that (13) is the system of necessary and sufficient optimality conditions for the quadratic problem

$$\min \ \frac{1}{2}\|n\|_{\mathcal{X}}^2 \tag{14a}$$

$$\text{s.t.} \ \ c_x(x_k)n + c(x_k) = 0. \tag{14b}$$

With a bounded linear operator W_k whose range is the null space of $c_x(x_k)$, we can eliminate (12b). Various such operators exist. We use the orthogonal projection onto the null space. In this case $W_k = W_k^* = W_k^2 \in \mathcal{L}(\mathcal{X}, \mathcal{X})$ and $s = W_k w$ can be computed by solving the system

$$\begin{pmatrix} I & c_x(x_k)^* \\ c_x(x_k) & 0 \end{pmatrix} \begin{pmatrix} s \\ z \end{pmatrix} = \begin{pmatrix} w \\ 0 \end{pmatrix}. \tag{15}$$

Using this operator, (12) can be written equivalently as

$$\min \ \frac{1}{2}\langle W_k H_k W_k t, t\rangle_{\mathcal{X}} + \langle \nabla_x L(x_k, \lambda_k) + H_k n_k, W_k t\rangle_{\mathcal{X}} \tag{16a}$$

$$\text{s.t.} \ \ \|t\|_{\mathcal{X}} \leq \Delta_k - \|n_k\|_{\mathcal{X}}. \tag{16b}$$

An approximate solution of (16) that is suitable for use within our SQP method can be computed using the Steihaug-Toint modification of the conjugate gradient method (see, e.g., [7]). With W_k given by (15), the Steihaug-Toint CG method can be implemented in an elegant way that in each CG iteration requires the application of W_k. See, e.g., [9]. Note that each application of W_k requires the solution of (15), which is the system of necessary and sufficient optimality conditions for

$$\min \ \frac{1}{2}\|s\|_{\mathcal{X}}^2 - \langle w, s\rangle_{\mathcal{X}} \tag{17a}$$

$$\text{s.t.} \ \ c_x(x_k)s = 0. \tag{17b}$$

We remark that it is easy to apply a preconditioned Steihaug-Toint CG method by replacing I in (15) by $\widetilde{H}_k$, where $\widetilde{H}_k$ is a selfadjoint operator that is strictly positive on the null-space of $c_x(x_k)$ and approximates H_k. (One can even set $\widetilde{H}_k = H_k$, if it is strictly positive on the null-space of $c_x(x_k)$.) In this case $\|s\|_{\mathcal{X}}^2$ in (17) has to be replaced by $\langle \widetilde{H}_k s, s\rangle_{\mathcal{X}}$. The requirements on $\widetilde{H}_k$ guarantee that the modified quadratic program (17) remains convex.

We conclude by noting that each iteration of our trust-region SQP method requires the solution of systems of the type (13) and (15) or, equivalently, the solution of convex quadratic programs of the type (14) and (17). The solves are done iteratively. Consequently, the SQP algorithm needs to provide stopping tolerances to the linear system solvers. These stopping tolerances need to be chosen to guarantee convergence of the overall algorithm, but at the same time it is desirable to choose them as large as possible to make the solution of these subproblems as inexpensive as possible. A rigorous approach that accomplishes this is detailed in [14, 15, 21]. It is used to generate the numerical results shown in the following section.

4 Optimal Control of a Semilinear Elliptic Equation

Our example problem (1) is a special case of (9) and is solved using the trust-region SQP method with inexact linear system solves outlined in the previous section. Each iteration of our trust-region SQP method requires the iterative solution of convex quadratic programs of the type (14) and (17). For the example problem (1) these quadratic programs are essentially of the form (2), with $\mathbf{a}$, r, f given by the current state and control determined by the SQP algorithm. The objective function in these subproblems is slightly different from (2a), but the domain decomposition approach outlined in Section 2 can easily be applied to these subproblems. We remark that all quadratic programs arising in our trust-region SQP method are known to be convex. Hence our optimization-level domain decomposition approach which decomposes the system of first order optimality conditions can be safely applied.

For our numerical example, we solve

$$\text{minimize } \frac{1}{2}\int_{\Omega}(y-\widehat{y})^2\,dx+\frac{\alpha}{2}\int_{\partial\Omega}u^2\,ds \tag{18a}$$

subject to

$$-\Delta y+y^3-y=f \ \text{ in } \Omega, \qquad \frac{\partial y}{\partial n}=u \text{ on } \partial\Omega. \tag{18b}$$

See, e.g., [11, 24]. We use $\Omega=(0,1)^2$, $\alpha=1$, $\widehat{y}(x)=\cos(\pi x_1)\cos(\pi x_2)$, and $f(x)=\cos(\pi x_1)\cos(\pi x_2)(2\pi^2+\cos^2(\pi x_1)\cos^2(\pi x_2)-1)$.

The problem (18) is discretized using piecewise linear finite elements for states and controls. The domain Ω is subdivided into triangles by first subdividing it into squares of size $h\times h$ and then subdividing each square into two triangles. The domain Ω is subdivided into square subdomains of size $H\times H$.

Tables 1 and 2 show the behavior of our SQP method with a one-level and two-level optimization-level Neumann-Neumann DD preconditioner for varying mesh and subdomain sizes. The number of outer SQP iterations is constant over varying mesh sizes and subdomain sizes. This is not too surprising (although not yet proven for our class of SQP methods), since we use an SQP method with exact second order derivative information and there are known mesh independence results for many Newton-like methods.

Within each iteration of the SQP method, a KKT-type system has to be solved for the computation of a Lagrange multiplier estimate, to compute the quasi-normal step (cf., (14)), and within each iteration of the Steihaug-Toint CG algorithm used to compute the tangential step (cf., (16)). Tables 1 and 2 show only a mild increase in the number of calls to GMRES as the number of subdomains is increased or the mesh size is decreased.

A significant difference is seen in the average number of GMRES iterations used to solve a KKT-type system depending on whether a one-level Neumann-Neumann DD preconditioner is used or a two-level preconditioner. This is

Table 1. One-level preconditioner: Number of SQP iterations, number of calls to GMRES, the total number of GMRES iterations, and the average number of GMRES iterations per call.

$1/h$	64×64			128×128		
$1/H$	2×2	4×4	8×8	2×2	4×4	8×8
SQP iter's	5	5	5	5	5	5
GMRES calls	36	41	45	40	50	53
GMRES total	195	1313	4733	197	1719	5895
GMRES avg	5.4	32.0	105.2	4.9	34.4	111.2

Table 2. Two-level preconditioner: Number of SQP iterations, number of calls to GMRES, the total number of GMRES iterations, and the average number of GMRES iterations per call.

$1/h$	64×64			128×128		
$1/H$	2×2	4×4	8×8	2×2	4×4	8×8
SQP iter's	5	5	5	5	5	5
GMRES calls	36	41	44	40	49	50
GMRES total	96	348	393	132	515	574
GMRES avg	2.7	8.5	8.9	3.3	10.5	11.5

expected since the performance of the one-level Neumann-Neumann DD preconditioner deteriorates as the number of subdomains increases, whereas the performance of the two-level preconditioner is insensitive to the number of subdomains. For single PDEs, this is shown theoretically as well as numerically, see, e.g., [22, 23]. For the optimization case this has been observed numerically ([13]), but not yet proven theoretically.

Figure 1 shows the relative residual stopping tolerances required for GMRES during its calls within the Steihaug-Toint CG algorithm used to compute the tangential step (cf., (16)). Each box/star indicates one call to GMRES, each box indicates a new SQP iteration. This figure shows that our SQP algorithm adjusts the stopping tolerance and has the capability to coarsen the relative residual stopping tolerance. We note that the dynamic adjustment is particularly beneficial over using a fixed stopping tolerance when the preconditioner is less effective and many GMRES iterations have to be executed to achieve a lower tolerance.

Acknowledgement. This work was supported in part by NSF grants ACI-0121360 and DMS-0511624.

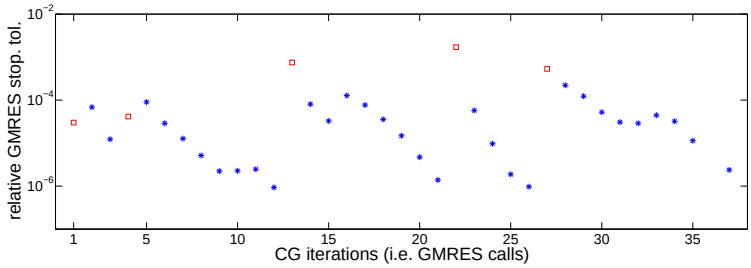

Fig. 1. Relative stopping tolerances for every call to GMRES within the Steihaug-Toint CG algorithm. One CG iteration corresponds to one GMRES call. The red square indicates the beginning of a new SQP iteration

References

[1] Y. Achdou and F. Nataf. A Robin-Robin preconditioner for an advection-diffusion problem. *C. R. Acad. Sci. Paris Sér. I Math.*, 325(11):1211–1216, 1997.

[2] Y. Achdou, P. Le Tallec, F. Nataf, and M. Vidrascu. A domain decomposition preconditioner for an advection-diffusion problem. *Comput. Methods Appl. Mech. Engrg.*, 184(2-4):145–170, 2000.

[3] R. A. Bartlett, M. Heinkenschloss, D. Ridzal, and B. van Bloemen Waanders. Domain decomposition methods for advection dominated linear–quadratic elliptic optimal control problems. *Comput. Methods Appl. Mech. Engrg.*, 195:6428–6447, 2006.

[4] G. Biros and O. Ghattas. Parallel Lagrange–Newton–Krylov–Schur methods for PDE–constrained optimization. part I: The Krylov–Schur solver. *SIAM J. Sci. Comput.*, 27(2):587–713, 2005.

[5] G. Biros and O. Ghattas. Parallel Lagrange–Newton–Krylov–Schur methods for PDE–constrained optimization. part II: The Lagrange–Newton solver and its application to optimal control of steady viscous flows. *SIAM J. Sci. Comput.*, 27(2):714–739, 2005.

[6] X.-C. Cai, M. Dryja, and M. Sarkis. Restricted additive Schwarz preconditioners with harmonic overlap for symmetric positive definite linear systems. *SIAM J. Numer. Anal.*, 41(4):1209–1231, 2003.

[7] A. R. Conn, N. I. M. Gould, and Ph. L. Toint. *Trust–Region Methods.* SIAM, Philadelphia, 2000.

[8] M. Delgado, J. A. Montero, and A. Suárez. Optimal control for the degenerate elliptic logistic equation. *Appl. Math. Optim.*, 45(3):325–345, 2002.

[9] N. I. M. Gould, M. E. Hribar, and J. Nocedal. On the solution of equality constrained quadratic programming problems arising in optimization. *SIAM J. Sci. Comput.*, 23(4):1376–1395, 2001.

[10] M. D. Gunzburger, L. S. Hou, and S. S. Ravindran. Analysis and approximation of optimal control problems for a simplified Ginzburg-Landau model of superconductivity. *Numer. Math.*, 77:243–268, 1997.

[11] M. D. Gunzburger, L. S. Hou, and T. P. Svobotny. Finite element approximations of an optimal control problem associated with the scalar Ginzburg-Landau equation. *Comput. Math. Appl.*, pages 123–131, 1991.

[12] M. Heinkenschloss and H. Nguyen. Domain decomposition preconditioners for linear–quadratic elliptic optimal control problems. Technical Report TR04–20, Dept. of Computational and Applied Mathematics, Rice U., 2004.

[13] M. Heinkenschloss and H. Nguyen. Neumann-Neumann domain decomposition preconditioners for linear–quadratic elliptic optimal control problems. *SIAM J. Sci. Comput.*, 28(3):1001–1028, 2006.

[14] M. Heinkenschloss and D. Ridzal. Solution of a class of quadratic programs arising in nonlinear programming using inexact linear system solves. Technical report, Dept. of Computational and Applied Math., Rice U., 2006.

[15] M. Heinkenschloss and L. N. Vicente. Analysis of inexact trust–region SQP algorithms. *SIAM J. Optim.*, 12:283–302, 2001.

[16] A. W. Leung. Positive solutions for systems of PDE and optimal control. *Nonlinear Anal.*, 47(2):1345–1356, 2001.

[17] X. Li and J Yong. *Optimal Control Theory for Infinite Dimensional Systems*. Systems & Control: Foundations & Applications. Birkhäuser-Verlag, Boston, Basel, Berlin, 1995.

[18] E. Prudencio, R. Byrd, and X.-C. Cai. Parallel full space SQP Lagrange-Newton-Krylov-Schwarz algorithms for PDE-constrained optimization problems. *SIAM J. Sci. Comput.*, 27(4):1305–1328, 2006.

[19] E. Prudencio and X.-C. Cai. Parallel multi-level Lagrange-Newton-Krylov-Schwarz algorithms with pollution removing for PDE-constrained optimization. Technical report, U. of Colorado at Boulder, Dept. of Computer Science, Boulder, Colorado, 2006.

[20] A. Quarteroni and A. Valli. *Domain Decomposition Methods for Partial Differential Equations*. Oxford University Press, Oxford, 1999.

[21] D. Ridzal. *Trust Region SQP Methods With Inexact Linear System Solves For Large-Scale Optimization*. PhD thesis, Dept. of Computational and Applied Mathematics, Rice U., Houston, TX, 2006.

[22] B. Smith, P. Bjørstad, and W. Gropp. *Domain Decomposition. Parallel Multilevel Methods for Elliptic Partial Differential Equations*. Cambridge University Press, Cambridge, London, New York, 1996.

[23] A. Toselli and O. Widlund. *Domain Decomposition Methods - Algorithms and Theory*. Computational Math., Vol. 34. Springer–Verlag, Berlin, 2004.

[24] F. Tröltzsch. *Optimale Steuerung partieller Differentialgleichungen. Grundlagen, Optimalitätsbedingungen und ihre Anwendungen*. Vieweg Verlag, Braunschweig, 2005.