

# Nonlinear optimized Schwarz preconditioner for elliptic optimal control problems

Gabriele Ciaramella, Felix Kwok and Georg Müller

## 1 Introduction

Consider the nonlinear optimal control problem

$$\begin{aligned} \min_{y,u} J(y,u) &:= \frac{1}{2} \|y - y_d\|_{L^2}^2 + \frac{\nu}{2} \|u\|_{L^2}^2 + \beta \|u\|_{L^1}, \\ \text{s.t. } -\Delta y + cy + b\varphi(y) &= f + u \text{ in } \Omega, \quad y = 0 \text{ on } \partial\Omega, \\ u &\in U_{\text{ad}} := \{v \in L^2(\Omega) : |v| \leq \bar{u} \text{ in } \Omega\}, \end{aligned} \tag{1}$$

where  $\|\cdot\|_{L^r}$  denotes the usual norm for  $L^r(\Omega)$  with  $1 \leq r \leq \infty$ , the functions  $y_d, f \in L^2(\Omega)$  are given, and the scalar parameters  $b, c, \beta \geq 0$  and  $\nu, \beta \geq 0$  are known. Our model includes problems such as the simplified Ginzburg-Landau superconductivity equation as well as inverse problems where  $L^1$ -regularization is used to enhance sparsity of the control function  $u$ . For simplicity, the domain  $\Omega \subset \mathbb{R}^2$  is assumed to be a rectangle  $(0, \tilde{L}) \times (0, \hat{L})$ . The function  $\varphi : \mathbb{R} \rightarrow \mathbb{R}$  is assumed to be of class  $C^2$ , with locally bounded and locally Lipschitz second derivative and such that  $\partial_y \varphi(y) \geq 0$ . These assumptions guarantee that the Nemytskii operator  $y(\cdot) \mapsto \varphi(y(\cdot))$  is twice continuously Fréchet differentiable in  $L^\infty(\Omega)$ . In this setting, the optimal control problem (1) is well posed in the sense that there exists a minimizer  $(y, u) \in X \times L^2(\Omega)$ , with  $X := H_0^1(\Omega) \cap L^\infty(\Omega)$ , cf. [7, 1]. Our goal is to derive efficient nonlinear preconditioners for solving (1) using domain decomposition techniques.

---

G. Ciaramella  
Politecnico di Milano e-mail: gabriele.ciaramella@polimi.it

F. Kwok  
Université Laval e-mail: felix.kwok@mat.ulaval.ca

G. Müller  
Universität Konstanz e-mail: georg.mueller@uni-konstanz.de

Let  $(y, u) \in X \times L^2(\Omega)$  be a solution to (1). Then there exists an adjoint variable  $p \in X$  such that  $(y, u, p)$  satisfies the system [6, Theorem 2.3]

$$\begin{aligned} -\Delta y + cy + b\varphi(y) &= f + u && \text{in } \Omega \text{ with } y = 0 \text{ on } \partial\Omega, \\ -\Delta p + cp + b\varphi'(y)p &= y - y_d && \text{in } \Omega \text{ with } p = 0 \text{ on } \partial\Omega, \\ u &= \mu(p), \end{aligned}$$

where  $\mu : L^\infty(\Omega) \rightarrow L^2(\Omega)$  is

$$\begin{aligned} \mu(p) &= \max(0, (-\beta - p)/\nu) + \min(0, (\beta - p)/\nu) \\ &\quad - \max(0, -\bar{u} + (-p - \beta)/\nu) - \min(0, \bar{u} + (-p + \beta)/\nu). \end{aligned} \quad (2)$$

We remark that for  $\beta = 0$ , the previous formula becomes  $\mu(p) = \mathbb{P}_{U_{\text{ad}}}(-p/\nu)$ , which is the usual projection formula that leads to the optimality condition  $u = \mathbb{P}_{U_{\text{ad}}}(-p/\nu)$ ; see [7]. Moreover, if  $\beta = 0$  with  $\bar{u} = \infty$ , one obtains that  $\mu(p) = -p/\nu$ , which implies the usual optimality condition  $\nu u + p = 0$ , where  $\nu u + p$  is the gradient of the reduced cost functional  $\hat{J}(u) = J(y(u), u)$  [7].

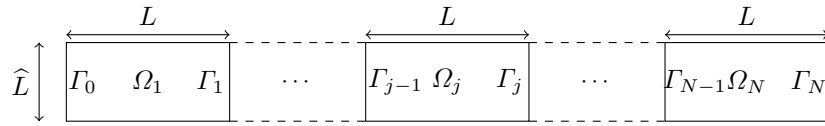
Eliminating the control using  $\mu(p)$ , the first-order optimality system becomes

$$\begin{aligned} -\Delta y + cy + b\varphi(y) &= f + \mu(p) && \text{in } \Omega \text{ with } y = 0 \text{ on } \partial\Omega, \\ -\Delta p + cp + b\varphi'(y)(p) &= y - y_d && \text{in } \Omega \text{ with } p = 0 \text{ on } \partial\Omega. \end{aligned} \quad (3)$$

This nonlinear and nonsmooth system admits a solution  $(y, p) \in X^2$  [1, 7].

## 2 Optimized Schwarz method and preconditioner

In this section, we introduce an optimized Schwarz method (OSM) for solving the optimality system (3). We consider the non-overlapping decomposition



**Fig. 1** Non-overlapping domain decomposition.

of  $\Omega$  shown in Fig. 1 and given by disjoint subdomains  $\Omega_j$ ,  $j = 1, \dots, N$  such that  $\bar{\Omega} = \cup_{j=1}^N \bar{\Omega}_j$ . The sets  $\Gamma_j := \bar{\Omega}_j \cap \bar{\Omega}_{j+1}$ ,  $j = 1, \dots, N-1$  are the interfaces. Moreover, we define  $\Gamma_j^{\text{ext}} := \partial\Omega_j \cap \partial\Omega$ ,  $j = 1, \dots, N$ , which represent the external boundaries of the subdomains. The optimality system (3) can be written as a coupled system of  $N$  subproblems defined on the subdomains  $\Omega_j$ ,  $j = 1, \dots, N$ , of the form

$$-\Delta y_j + c y_j + b \varphi(y_j) = f_j + \mu(p_j) \quad \text{in } \Omega_j, \quad (4a)$$

$$-\Delta p_j + c p_j + b \varphi'(y_j)(p_j) = y_j - y_{d,j} \quad \text{in } \Omega_j \quad (4b)$$

$$y_j = 0, p_j = 0 \quad \text{on } \Gamma_j^{\text{ext}}, \quad (4c)$$

$$q y_j + \partial_x y_j = q y_{j+1} + \partial_x y_{j+1} \quad \text{on } \Gamma_j, \quad (4d)$$

$$q p_j + \partial_x p_j = q p_{j+1} + \partial_x p_{j+1} \quad \text{on } \Gamma_j, \quad (4e)$$

$$q y_j - \partial_x y_j = q y_{j-1} - \partial_x y_{j-1} \quad \text{on } \Gamma_{j-1}, \quad (4f)$$

$$q p_j - \partial_x p_j = q p_{j-1} - \partial_x p_{j-1} \quad \text{on } \Gamma_{j-1}, \quad (4g)$$

for  $j = 1, \dots, N$ , where for  $j \in \{1, N\}$  the boundary conditions at  $\Gamma_0$  and  $\Gamma_N$ , respectively, must be replaced with homogeneous Dirichlet conditions. Here,  $q > 0$  is a parameter that can be optimized to improve the convergence of the OSM; see, e.g., [5, 2]. The system (4) leads to the OSM, which, for a given  $(y_j^0, p_j^0)_{j=1}^N$ , consists of solving the subdomain problems below for  $\mathbf{y}_j^k := (y_j^k, p_j^k)$ ,  $k = 1, 2, 3, \dots$ :

$$-\Delta y_j^k + c y_j^k + b \varphi(y_j^k) = f_j + \mu(p_j^k) \quad \text{in } \Omega_j, \quad (5a)$$

$$-\Delta p_j^k + c p_j^k + b \varphi'(y_j^k)(p_j^k) = y_j^k - y_{d,j} \quad \text{in } \Omega_j \quad (5b)$$

$$\mathbf{y}_j^k = 0, \quad \text{on } \Gamma_j^{\text{ext}}, \quad (5c)$$

$$q \mathbf{y}_j^k + \partial_x \mathbf{y}_j^k = q \mathbf{y}_{j+1}^{k-1} + \partial_x \mathbf{y}_{j+1}^{k-1} \quad \text{on } \Gamma_j, \quad (5d)$$

$$q \mathbf{y}_j^k - \partial_x \mathbf{y}_j^k = q \mathbf{y}_{j-1}^{k-1} - \partial_x \mathbf{y}_{j-1}^{k-1} \quad \text{on } \Gamma_{j-1}, \quad (5e)$$

Now, we use the OSM to introduce a nonlinear preconditioner by setting  $\mathbf{y}_j := (y_j, p_j)$ ,  $j = 1, \dots, N$ , and defining the solution maps  $S_j$  as

$$S_1(\mathbf{y}_2) = \mathbf{y}_1 \quad \text{solution to (4) with } j = 1 \text{ and } \mathbf{y}_2 \text{ given,}$$

$$S_j(\mathbf{y}_{j-1}, \mathbf{y}_{j+1}) = \mathbf{y}_j \quad \text{solution to (4) with } 2 \leq j \leq N-1 \text{ and } \mathbf{y}_{j\pm 1} \text{ given,}$$

$$S_N(\mathbf{y}_{N-1}) = \mathbf{y}_N \quad \text{solution to (4) with } j = N \text{ and } \mathbf{y}_{N-1} \text{ given.}$$

Hence, using the variable  $\mathbf{y} = (\mathbf{y}_1, \dots, \mathbf{y}_N)$ , we can rewrite (4) as

$$\mathcal{F}_P(\mathbf{y}) = 0, \quad \text{where } \mathcal{F}_P(\mathbf{y}) := \begin{bmatrix} \mathbf{y}_1 - S_1(\mathbf{y}_2) \\ \mathbf{y}_2 - S_2(\mathbf{y}_1, \mathbf{y}_3) \\ \vdots \\ \mathbf{y}_{N-1} - S_{N-1}(\mathbf{y}_{N-2}, \mathbf{y}_N) \\ \mathbf{y}_N - S_N(\mathbf{y}_{N-1}) \end{bmatrix}. \quad (6)$$

This is the nonlinearly preconditioned form of (3) induced by the OSM (4)-(5), to which we can apply a generalized Newton method. For a given initialization  $\mathbf{y}^0$ , a Newton method generates a sequence  $(\mathbf{y}^k)_{k \in \mathbb{N}}$  defined by

$$\text{solve } D\mathcal{F}_P(\mathbf{y}^k)(\mathbf{d}^k) = -\mathcal{F}_P(\mathbf{y}^k) \quad \text{and update } \mathbf{y}^{k+1} = \mathbf{y}^k + \mathbf{d}^k. \quad (7)$$

Notice that at each iteration of (7) one needs to evaluate the residual function  $\mathcal{F}_P(\mathbf{y}^k)$ , which requires the (parallel) solution of the  $N$  subproblems (4). The computational cost is therefore equivalent to one iteration of the OSM (5). As an inner solver for the subproblems, which involve the (mildly) non-differentiable function  $\mu$ , a semismooth Newton can be employed.

We now discuss the problem of solving the Jacobian linear system in (7). Let  $\mathbf{d} = (\mathbf{d}_1, \dots, \mathbf{d}_N)$ , where  $\mathbf{d}_j = (d_{y,j}, d_{p,j})$ ,  $j = 1, \dots, N$ . Then a direct calculation (omitted for brevity) shows that the action of the operator  $D\mathcal{F}_P(\mathbf{y})$  on the vector  $\mathbf{d}$  is given by  $D\mathcal{F}_P(\mathbf{y})(\mathbf{d}) = \mathbf{d} - \tilde{\mathbf{y}}(\mathbf{d})$ , where  $\tilde{\mathbf{y}} := (\tilde{\mathbf{y}}_1, \dots, \tilde{\mathbf{y}}_N)$ , and each  $\tilde{\mathbf{y}}_j = (\tilde{y}_j, \tilde{p}_j)$  satisfies the *linearized* subdomain problems

$$-\Delta \tilde{y}_j + c \tilde{y}_j + b \varphi'(y_j) \tilde{y}_j = D\mu(p_j)(\tilde{p}_j) \quad \text{in } \Omega_j, \quad (8a)$$

$$-\Delta \tilde{p}_j + c \tilde{p}_j + b \varphi''(y_j)[p_j, \tilde{y}_j] = \tilde{y}_j \quad \text{in } \Omega_j \quad (8b)$$

$$\tilde{\mathbf{y}}_j = 0, \quad \text{on } \Gamma_j^{\text{ext}}, \quad (8c)$$

$$q \tilde{\mathbf{y}}_j + \partial_x \tilde{\mathbf{y}}_j = q \mathbf{d}_{j+1} + \partial_x \mathbf{d}_{j+1} \quad \text{on } \Gamma_j, \quad (8d)$$

$$q \tilde{\mathbf{y}}_j - \partial_x \tilde{\mathbf{y}}_j = q \mathbf{d}_{j-1} - \partial_x \mathbf{d}_{j-1} \quad \text{on } \Gamma_{j-1}, \quad (8e)$$

where

$$D\mu(p)(\tilde{p}) = \frac{1}{\nu} \left[ -\mathcal{G}_{\max}(-\beta - p) - \mathcal{G}_{\min}(\beta - p) + \mathcal{G}_{\max}(-p - \beta - \nu \bar{u}) + \mathcal{G}_{\min}(-p + \beta + \nu \bar{u}) \right] \tilde{p},$$

$$\text{with } \mathcal{G}_{\max}(v)(x) = \begin{cases} 1 & \text{if } v(x) > 0, \\ 0 & \text{if } v(x) \leq 0, \end{cases} \text{ and } \mathcal{G}_{\min}(v)(x) = \begin{cases} 1 & \text{if } v(x) \leq 0, \\ 0 & \text{if } v(x) > 0, \end{cases}$$

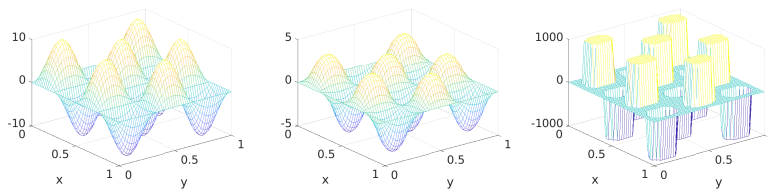
and where the boundary values for  $j \in \{1, N\}$  have to be modified as in (4). Note that this is the same linearized problem that must be solved repeatedly within the inner iterations of semismooth Newton, so its solution cost is only a fraction of the cost required to calculate  $\mathcal{F}_P(\mathbf{y})$ . Our matrix-free preconditioned semismooth Newton algorithm that corresponds to the Newton procedure (7) is summarized in Algorithm 1.

### 3 Numerical experiments

Let us begin with a two subdomain case for  $\Omega = (0, 1)^2$ ,  $y_d(x, y) = 10 \sin(4\pi x) \sin(3\pi y)$ ,  $f = 0$ ,  $c = 1$  and  $\varphi(y) = y + \exp(y)$ . The domain  $\Omega$  is discretized with a uniform mesh of 51 interior points on each edge of the unit square. The discrete optimality system is obtained by the finite difference method. Fig. 2 shows an example of the solution computed for  $b = 10$ ,  $\nu = 10^{-7}$ ,  $\bar{u} = 10^3$  and  $\beta = 10^{-2}$ . Here, we can observe how the computed

**Algorithm 1** Matrix-free preconditioned generalized Newton method

- 
- Require:** Initial guess  $\mathbf{y}^0$ , tolerance  $\epsilon$ , maximum number of iterations  $k_{\max}$ .
- 1: Compute  $S_1(\mathbf{y}_2^0)$ ,  $S_j(\mathbf{y}_{j-1}^0, \mathbf{y}_{j+1}^0)$ ,  $j = 2, \dots, N-1$ , and  $S_N(\mathbf{y}_{N-1}^0)$ .
  - 2: Set  $k = 0$  and assemble  $\mathcal{F}_P(\mathbf{y}^0)$  using (6).
  - 3: **while**  $\|\mathcal{F}_P(\mathbf{y}^k)\| \geq \epsilon$  and  $k \leq k_{\max}$  **do**
  - 4:   Compute  $\mathbf{d}^k$  by solving  $D\mathcal{F}_P(\mathbf{y}^k)(\mathbf{d}^k) = -\mathcal{F}_P(\mathbf{y}^k)$  using a matrix-free Krylov method, e.g., GMRES (together with a routine for solving (8) to compute the action of  $D\mathcal{F}_P(\mathbf{y}^k)$  on a vector  $\mathbf{d}$ ).
  - 5:   Update  $\mathbf{y}^{k+1} = \mathbf{y}^k + \mathbf{d}^k$ .
  - 6:   Set  $k = k + 1$ .
  - 7:   Compute  $S_1(\mathbf{y}_2^k)$ ,  $S_j(\mathbf{y}_{j-1}^k, \mathbf{y}_{j+1}^k)$ ,  $j = 2, \dots, N-1$ , and  $S_N(\mathbf{y}_{N-1}^k)$ .
  - 8:   Assemble  $\mathcal{F}_P(\mathbf{y}^k)$  using (6).
  - 9: **end while**
  - 10: **Output:**  $\mathbf{y}^k$ .
- 



**Fig. 2** Target  $y_d$  (left), optimal state  $y$  (middle), and optimal control  $u$  (right) computed for  $b = 10$ ,  $\nu = 10^{-7}$  and  $\beta = 10^{-2}$ .

optimal state (middle) has the same shape as the target  $y_d$  (left), but the control constraints and the  $L^1$ -penalization prevent the control function from making the state equal to the desired target.

To study the efficiency and the robustness of the proposed numerical framework, we test the nonlinearly preconditioned Newton for several values of parameters  $\nu$ ,  $\beta$ ,  $\bar{u}$ ,  $b$  and  $q$ , and compare the obtained number of iterations with the ones performed by a (damped) semismooth Newton applied directly to (3). Moreover, to improve the robustness of our preconditioned Newton method, we implemented the following continuation procedure with respect to the regularization parameter  $\nu$ : for  $k = 1$ , we set  $\nu_1 = 10^{-1}$  and solve the Jacobian system (7) *once* to obtain  $\mathbf{y}^2$ . Next, we decrease  $\nu$  by a factor of 4 ( $\nu_2 = \nu_1/4$ ), do another solve and update step (7), and so on. When we reach the true  $\nu$  prescribed by the problem, we set  $\nu_k = \nu$  and repeat (7) until convergence; see [3] for convergence results for similar continuation procedures. We apply the same continuation procedure on semismooth Newton applied directly to (3) for comparison. Note that because only one Jacobian solve is performed before  $\nu$  is updated, there are cases where semismooth Newton with continuation diverges, *even when its counterpart without continuation converges*, see Tab. 1. We initialize the four methods by randomly chosen vectors. The number of iterations performed by both methods to reach a tolerance of  $10^{-8}$  are reported in Tab. 1, where the symbol  $\times$  indicates diver-

	$q$	$b$	$\bar{u} = 10^3$			$\bar{u} = \infty$		
			$\nu = 10^{-3}$	$\nu = 10^{-5}$	$\nu = 10^{-7}$	$\nu = 10^{-3}$	$\nu = 10^{-5}$	$\nu = 10^{-7}$
$\beta = 0$	1	0	4-5-2-5	6-9-11-11	4-11-41-12	3-5-2-5	3-9-2-9	3-12-3-12
	10	0	4-5-2-5	6-9-11-11	8-11-41-12	3-5-2-5	3-8-2-9	3-11-3-12
	100	0	3-5-2-5	6-9-11-11	×-11-41-12	3-5-2-5	3-9-2-9	3-11-3-12
	1	10	6-6-4-7	×-10-12-12	×-12-38-16	6-6-4-7	×-10-22-23	×-15-×-15
	10	10	5-6-4-7	7-10-12-12	×-12-38-16	5-6-4-7	×-10-22-23	×-14-×-15
	100	10	4-6-4-7	6-10-12-12	×-13-38-16	4-6-4-7	6-10-22-23	×-13-×-15
$\beta = 10^{-2}$	1	0	5-5-3-6	6-9-8-11	×-12-43-13	4-5-3-6	5-9-6-10	×-12-8-15
	10	0	4-5-3-6	6-9-8-11	×-11-43-13	4-5-3-6	4-9-6-10	×-12-8-15
	100	0	4-5-3-6	6-10-8-11	11-12-43-13	4-5-3-6	5-9-6-10	7-12-8-15
	1	10	6-6-4-6	×-11-10-12	×-12-×-17	6-6-4-6	×-10-18-×	×-13-×-15
	10	10	5-6-4-6	×-11-10-12	×-13-×-17	5-6-4-6	×-10-18-×	×-14-×-15
	100	10	4-6-4-6	6-11-10-12	9-13-×-17	4-6-4-6	6-10-18-×	×-13-×-15

**Table 1** Two subdomains: outer iterations of preconditioned Newton (left value), preconditioned Newton with continuation (middle-left value), semismooth Newton applied to the original problem (middle-right value), and semismooth Newton with continuation applied to the original problem (right value).

gence. These results show that if the preconditioned Newton converges, then it outperforms the semismooth Newton applied directly to the full system (3). However, the preconditioned Newton does not always converge due to the lack of damping. With continuation, however, our method always converges, with an iteration count comparable (for moderate values of  $\nu$ ) or much lower (for small  $\nu$ ) than for the semismooth Newton method.

To better gauge the cost of the continuation strategy, we show the total number of inner iterations required by ‘pure’ preconditioned Newton versus the one with continuation in Tab. 2. The reported numbers are computed as  $\sum_k \max_{j=1,2} it_{j,k}$ , where  $k$  is the iteration count and  $it_{j,k}$ ,  $j = 1, 2$ , are the number of inner iterations required by the two subdomain solves performed at the  $k$ th outer iteration. (The max accounts for the fact that the two subdomain problems are supposed to be solved in parallel.) The results show that the continuation procedure actually *reduces* the total number of inner iterations for the most part, except for some very easy cases, such as  $\beta = b = 0$ ,  $\bar{u} = \infty$  (where the problem is in fact linear).

Finally, Tab. 3 shows the total number of GMRES iterations required for solving (7) (with or without continuation), together with the GMRES iteration count for semismooth Newton (with or without continuation); the latter is preconditioned by block Jacobi, using  $-\Delta + cI$  as diagonal blocks. We see that for the “easy” case of  $\nu = 10^{-3}$ , semismooth Newton requires fewer GMRES steps than preconditioned Newton, but the situation reverses for smaller  $\nu$ . In fact, for a well-chosen Robin parameter such as  $q = 10$ , the advantage of preconditioned Newton with continuation can be quite significant in these harder cases. All these numerical observations show clearly the efficiency of the proposed computational framework.

	$q$	$b$	$\bar{u} = 10^3$			$\bar{u} = \infty$		
			$\nu = 10^{-3}$	$\nu = 10^{-5}$	$\nu = 10^{-7}$	$\nu = 10^{-3}$	$\nu = 10^{-5}$	$\nu = 10^{-7}$
$\beta = 0$	1	0	6 - 5	31 - 12	$\times$ - 18	2 - 5	3 - 8	3 - 11
	10	0	5 - 5	26 - 11	96 - 19	2 - 5	3 - 8	3 - 11
	100	0	2 - 5	18 - 13	$\times$ - 19	2 - 5	2 - 8	3 - 11
	1	10	$\times$ - 17	$\times$ - 35	$\times$ - 47	27 - 17	$\times$ - 34	$\times$ - 60
	10	10	21 - 14	$\times$ - 31	103 - 43	21 - 14	$\times$ - 32	$\times$ - 53
	100	10	8 - 14	26 - 32	$\times$ - 43	8 - 14	45 - 30	$\times$ - 47
$\beta = 10^{-2}$	1	0	13 - 8	32 - 16	84 - 25	8 - 8	10 - 14	$\times$ - 25
	10	0	10 - 8	22 - 17	33 - 23	7 - 8	11 - 15	$\times$ - 24
	100	0	7 - 6	15 - 15	104 - 20	7 - 6	12 - 13	$\times$ - 22
	1	10	$\times$ - 17	$\times$ - 33	$\times$ - 45	28 - 17	$\times$ - 32	$\times$ - 47
	10	10	20 - 14	$\times$ - 33	$\times$ - 48	20 - 14	$\times$ - 30	$\times$ - 46
	100	10	10 - 14	23 - 30	125 - 44	10 - 14	40 - 26	$\times$ - 44

**Table 2** Two subdomains: total number of inner iterations of preconditioned Newton (left value) and preconditioned Newton with continuation (right value).

	$q$	$b$	$\bar{u} = 10^3$			$\bar{u} = \infty$		
			$\nu = 10^{-3}$	$\nu = 10^{-5}$	$\nu = 10^{-7}$	$\nu = 10^{-3}$	$\nu = 10^{-5}$	$\nu = 10^{-7}$
$\beta = 0$	1	0	143-128-26-17	279-163-303-123	139-175-1255-155	100-128-26-17	170-140-69-46	$\times$ -149-371-66
	10	0	90-75-26-17	179-101-303-123	254-108-1255-155	64-75-26-17	114-88-69-46	$\times$ -92-371-66
	100	0	73-90-26-17	177-114-303-123	$\times$ -125-1255-155	73-90-26-17	60-88-69-46	54-93-371-66
	1	10	266-204-49-66	$\times$ -251-397-255	$\times$ -268-1479-457	256-204-49-66	$\times$ -240-2000-1172	$\times$ -379- $\times$ -928
	10	10	124-88-49-66	226-129-397-255	$\times$ -168-1479-457	117-88-49-66	$\times$ -152-2000-1172	$\times$ -190- $\times$ -928
	100	10	122-155-49-66	139-239-397-255	$\times$ -274-1479-457	122-155-49-66	161-234-2000-1172	$\times$ -250- $\times$ -928
$\beta = 10^{-2}$	1	0	226-164-31-42	290-198-187-123	$\times$ -223-1065-168	188-164-31-42	246-168-183-109	$\times$ -218-522-380
	10	0	111-95-31-42	178-121-187-123	$\times$ -130-1065-168	115-95-31-42	143-124-183-109	$\times$ -145-522-380
	100	0	135-118-31-42	179-158-187-123	333-175-1065-168	135-118-31-42	145-147-183-109	165-173-522-380
	1	10	273-235-49-54	$\times$ -238-299-233	$\times$ -228- $\times$ -416	261-235-49-54	$\times$ -254-1362- $\times$	$\times$ -311- $\times$ -752
	10	10	139-124-49-54	$\times$ -158-299-233	$\times$ -164- $\times$ -416	138-124-49-54	$\times$ -161-1362- $\times$	$\times$ -179- $\times$ -752
	100	10	122-162-49-54	141-251-299-233	215-300- $\times$ -416	122-162-49-54	167-219-1362- $\times$	$\times$ -303- $\times$ -752

**Table 3** Two subdomains: GMRES iterations of preconditioned Newton (left value), preconditioned Newton with continuation (middle-left value), semismooth Newton applied to the original problem (middle-right value), and semismooth Newton with continuation applied to the original problem (right value).

	$N$	$b$	$\bar{u} = 10^3$			$\bar{u} = \infty$		
			$\nu = 10^{-3}$	$\nu = 10^{-5}$	$\nu = 10^{-7}$	$\nu = 10^{-3}$	$\nu = 10^{-5}$	$\nu = 10^{-7}$
$\beta = 0$	4	0	94-116-131-157	$\times$ -185-541-329	$\times$ -196-1974-436	92-112-131-157	64-115-151-246	58-128-513-528
	8	0	178-157-176-217	$\times$ -244-503-365	$\times$ -259-2261-455	121-155-176-217	83-180-140-322	59-204-438-537
	16	0	228-229-217-301	$\times$ -383-747-509	$\times$ -384-2915-554	159-238-217-301	130-281-195-444	$\times$ -314-201-569
	4	10	145-184-202-234	218-287-447-517	$\times$ -382-1207-910	143-186-202-234	171-276-633-506	$\times$ -296- $\times$ -1238
	8	10	192-247-232-314	286-381-453-557	$\times$ -498-1475-979	196-247-232-314	214-368-621-552	$\times$ -368- $\times$ -1217
	16	10	272-346-364-425	$\times$ -625-648-777	$\times$ -744-1361-1180	280-345-364-425	327-549-819-765	$\times$ -510- $\times$ -1200
$\beta = 10^{-2}$	4	0	177-139-169-204	221-220-389-301	399-255-1745-439	179-135-169-204	175-196-340-315	243-273-992-649
	8	0	231-191-266-277	$\times$ -291-448-383	$\times$ -340-1594-487	231-199-266-277	275-264-350-385	300-360-983-658
	16	0	319-241-314-372	552-395-667-528	$\times$ -452-2244-630	227-234-314-372	438-365-517-529	520-449-899-1714
	4	10	145-201-209-248	230-312-369-500	380-408-1536-851	147-201-209-248	193-312-982-492	$\times$ -392- $\times$ -1320
	8	10	196-262-267-323	356-408-402-574	$\times$ -641-1625-1020	193-261-267-323	247-406-1041-568	$\times$ -488- $\times$ -1396
	16	10	268-365-368-444	538-660-577-794	$\times$ -1017-2000-1312	260-365-368-444	432-599-1425-790	$\times$ -702- $\times$ -1686

**Table 4** Multiple subdomains: GMRES iterations of preconditioned Newton (left value), preconditioned Newton with continuation (middle-left value), semismooth Newton applied to the original problem (middle-right value), and semismooth Newton with continuation applied to the original problem (right value).

We now consider a multiple subdomain case. This time, the mesh is refined to have 101 interior points on each edge of  $\Omega$ . We then fix  $q = 100$  and

repeat the experiments above for  $N = 4, 8, 16$  subdomains. In Tab. 4, we compare the GMRES iteration counts for preconditioned Newton (with and without continuation) to those of semismooth Newton applied to (3). We see that preconditioned Newton with continuation works well in all cases, and for smaller  $\nu$  values, the iteration count is much lower than for semismooth Newton. The outer iteration counts are omitted for brevity, but we observed a behaviour similar to the two-subdomain case, and one which is robust for the mesh sizes  $h = \frac{1}{26}, \frac{1}{51}, \frac{1}{101}$ ; see [2, 4] for related scalability discussions.

## 4 Further discussion and conclusion

This short manuscript represents a proof of concept for using domain decomposition-based nonlinear preconditioning to efficiently solve nonlinear, non-smooth optimal control problems governed by elliptic equations. However, several theoretical and numerical issues must be addressed as part of a complete development of these techniques. From a theoretical point of view, to establish concrete convergence results based on classical semismooth Newton theory, it is crucial to study the (semismoothness) properties of the subdomain solution maps  $\mathcal{S}_j$ , which are implicit function of semismooth maps. Another crucial point is the proof of well-posedness of the (preconditioned) Newton linear system. From a domain decomposition perspective, more general decompositions (including cross points) must be considered. Finally, a detailed analysis of the scalability of the GMRES iterations is necessary.

## References

1. E. Casas, R. Herzog, and G. Wachsmuth. Optimality conditions and error analysis of semilinear elliptic control problems with L1 cost functional. *SIAM J. Optim.*, 22(3):795–820, 2012.
2. F. Chaouqui, G. Ciaramella, M. J. Gander, and T. Vanzan. On the scalability of classical one-level domain-decomposition methods. *Vietnam Journal of Mathematics*, 46(4):1053–1088, 2018.
3. G. Ciaramella, A. Borzi, G. Dirr, and D. Wachsmuth. Newton methods for the optimal control of closed quantum spin systems. *SIAM J. Sci. Comp.*, 37(1):A319–A346, 2015.
4. G. Ciaramella and M. J. Gander. Analysis of the parallel Schwarz method for growing chains of fixed-sized subdomains: Part I. *SIAM J. Numer. Anal.*, 55(3):1330–1356, 2017.
5. M. J. Gander. Optimized Schwarz methods. *SIAM J. Numer. Anal.*, 44(2):699–731, 2006.
6. G. Stadler. Elliptic optimal control problems with L1-control cost and applications for the placement of control devices. *Comput. Optim. Appl.*, 44(2):159–181, 2009.
7. F. Tröltzsch. *Optimal Control of Partial Differential Equations: Theory, Methods, and Applications*. American Mathematical Society, 2010.