Multilevel Active-Set Trust-Region (MASTR) Method for Bound Constrained Minimization

Alena Kopaničáková and Rolf Krause

1 Introduction

We consider a minimization problem of the following type:

$$\begin{aligned} \min_{\mathbf{x}\in\mathbb{R}^n} \quad & f(\mathbf{x}) \\ \text{subject to} \quad & \mathbf{x} \in \mathcal{F}, \end{aligned} \tag{P}$$

where $f : \mathbb{R}^n \to \mathbb{R}$ is possibly non-convex, but twice continuously differentiable objective function. The feasible set $\mathcal{F} := \{\mathbf{x} \in \mathbb{R}^n | \, \mathbf{l} \leq \mathbf{x} \leq \mathbf{u}\}$ is defined in terms of the pointwise lower bound $\mathbf{l} \in \mathbb{R}^n$ and the upper bound $\mathbf{u} \in \mathbb{R}^n$. We assume that the function f arises from the finite element (FE) discretization of a partial differential equations (PDEs). Here, $n \in \mathbb{N}$ denotes the dimension of the finite element space and it is typically very large. Problems of this type arise commonly in many scientific applications, for example in fracture mechanics [11].

Multilevel methods are known to be optimal solution strategies for systems arising from the discretization of, usually elliptic, PDEs, as their convergence rate is often independent of the problem size and the number of required arithmetic operations grows proportionally with the number of unknowns. These methods have been originally designed for unconstrained PDEs [2]. Their extension to constrained settings is not trivial as the coarse levels are often not capable of resolving the finest-level constraints sufficiently well, especially if the constraints are oscillatory [13]. The initial attempts to incorporate the constraints into the multilevel framework were associated with solving linear complementarity problems, see for instance [14, 1, 8, 5]. The devised methods employed various constraint projection rules for constructing the coarse-level variable bounds, such that coarse-level corrections are admissible by

Alena Kopaničáková
Università della Svizzera italiana, Switzerland, e-mail: alena.kopanicakova@usi.ch

Rolf Krause
Università della Svizzera italiana, Switzerland, e-mail: rolf.krause@usi.ch

the finest level. Unfortunately, these projection rules tend to be overly restrictive. As a consequence, the resulting multilevel methods converge significantly slower than standard linear multigrid. In order to enhance the convergence speed, Kornhuber proposed an active-set multigrid method [12]. The method utilizes a truncated basis approach and recovers the convergence rate of the unconstrained multigrid, once the exact active-set is detected [9, 12].

In the field of nonlinear optimization, very few existing nonlinear multilevel algorithms can be readily employed. For instance, Vallejos proposed a gradient projection based multilevel method [15]. Two multilevel line-search methods, designed for convex optimization problems, are proposed in [10]. These methods utilize constraint projection rules developed in [8] and a variant of the active-set strategy from [12]. In the context of non-convex optimization problems, Youett et al. proposed filter trust-region algorithm [16], which employs active-set multigrid method [12] for the solution of arising linearized problems. Furthermore, Gratton et al. proposed a variant of the recursive multilevel trust-region (RMTR) method [7] by utilizing the constraint projection rules from [5]. To our knowledge, this is currently the only inherently nonlinear multilevel method, which provides global convergence guarantees for non-convex bound constrained optimization problems.

In the presented work, we propose to enhance the convergence speed of the RMTR method introduced in [7]. More precisely, we present an active-set variant, called Multilevel Active-Set Trust-Region (MASTR) method. The MASTR method employs an active-set strategy, which determines, which components of the fine-level solution vector are active. These components are then held fixed and cannot be altered by the coarser levels. To this aim, we have to construct coarse-level models such that their minimization yields corrections, which fulfill this requirement. Here, we employ coarse-level models of the Galerkin type together with the truncated basis method [12]. In contrast to [10], the practical implementation of the proposed coarse-level models does not require any unconventional modifications to existing FE software packages. As it will be demonstrated by our numerical results, employing the active-set approach leads to significant speedup of the RMTR method.

2 Recursive multilevel trust-region (RMTR) method

In this work, we minimize (P) using a novel variant of the RMTR method [7]. RMTR combines the global convergence properties of the trust-region (TR) method with the efficiency of multilevel methods. By design, the RMTR method employs a hierarchy of L levels. Each level l is associated with a mesh $\mathcal{T}^l$, which encapsulates the computational domain $\Omega \in \mathbb{R}^d$, where $d \in \mathbb{N}$. The mesh $\mathcal{T}^l$ is used to construct the first-order finite-element space V^l, spanned by the basis functions $\{N_k^l\}_{k \in \mathcal{N}^l}$, where $\mathcal{N}^l$ denotes the set of nodes of the mesh $\mathcal{T}^l$. The support of a given basis function N_k^l is defined as $\omega_k^l = \overline{\{x \in \Omega \mid N_k^l(x) \neq 0\}}$.

The transfer of data between subsequent levels of the multilevel hierarchy is carried out using three transfer operators, namely prolongation $\mathbf{I}_l^{l+1} : \mathbb{R}^{n^l} \to \mathbb{R}^{n^{l+1}}$,

restriction $\mathbf{R}_{l+1}^{l} := (\mathbf{I}_{l}^{l+1})^T$ and projection $\mathbf{P}_{l+1}^{l} : \mathbb{R}^{n^{l+1}} \to \mathbb{R}^{n^l}$. In this work, all transfer operators are assembled using L^2-projection.

Algorithm

On each level l, the RMTR method approximates (P) by means of some level-dependent objective function $h^l : \mathbb{R}^{n^l} \to \mathbb{R}$ and feasible set $\mathcal{F}^l := \{\mathbf{x}^l \in \mathbb{R}^{n^l} \mid \mathbf{l}^l \leq \mathbf{x}^l \leq \mathbf{u}^l\}$. The function h^l is approximately minimized in order to obtain a coarse-level correction. This correction is then interpolated to the subsequent finer level, $l+1$, where it is used to improve the current iterate.

More precisely, the algorithm starts on the finest level, $l = L$, with an initial iterate $\mathbf{x}_0^L$ and passes through all levels until the coarsest level, $l = 1$, is reached. On each level l, the algorithm performs μ_1 pre-smoothing steps to improve the current iterate $\mathbf{x}_0^l$. The smoothing is performed using the TR method [3]. Thus, on each TR iteration i, the search direction $\mathbf{s}_i^l$ is obtained by approximately solving the following minimization problem:

$$\begin{aligned} \min_{\mathbf{s}_i^l \in \mathbb{R}^{n^l}} m_i^l(\mathbf{x}_i^l + \mathbf{s}_i^l) :=& h^l(\mathbf{x}_i^l) + \langle \nabla h^l(\mathbf{x}_i^l), \mathbf{s}_i^l \rangle + \frac{1}{2}\langle \mathbf{s}_i^l, \nabla^2 h^l(\mathbf{x}_i^l)\, \mathbf{s}_i^l \rangle, \\ & \text{such that } \ \mathbf{x}_i^l + \mathbf{s}_i^l \in \mathcal{F}^l, \\ & \|\mathbf{s}_i^l\|_\infty \leq \Delta_i^l, \end{aligned} \tag{1}$$

where m_i is the second-order Taylor approximation of h^l. The symbol $\Delta_i^l > 0$ denotes a TR radius, which controls the size of the correction $\mathbf{s}_i^l$. In contrast to line-search methods, the correction $\mathbf{s}_i^l$ is used only if $\rho_i^l > \eta_1$, where $\rho_i^l = \frac{h^l(\mathbf{x}_i^l) - h^l(\mathbf{x}_i^l + \mathbf{s}_i^l)}{m^l(\mathbf{x}_i^l) - m^l(\mathbf{x}_i^l + \mathbf{s}_i^l)}$ and $\eta_1 > 0$. Otherwise, $\mathbf{s}_i^l$ is disposed of and the size of the TR radius is reduced. The result of the pre-smoothing, the iterate $\mathbf{x}_{\mu_1}^l$, is then used to initialize the solution vector on the subsequent coarser level, i.e., $\mathbf{x}_0^{l-1} := \mathbf{P}_l^{l-1}\mathbf{x}_{\mu_1}^l$.

Once the coarsest level is reached, we apply μ^1 steps of the TR method to obtain the updated iterate $\mathbf{x}_{\mu^1}^1$. The algorithm then returns to the finest level. To this end, the correction obtained on the level l, i.e., $\mathbf{x}_{\mu^l}^l - \mathbf{x}_0^l$, is transfered to the level $l+1$, by means of the prolongation operator, thus $\mathbf{s}_{\mu_1+1}^{l+1} := \mathbf{I}_l^{l+1}(\mathbf{x}_{\mu^l}^l - \mathbf{x}_0^l)$. Here, the symbol μ^l denotes a sum of all iterations taken on a given level l. However, the quality of the prolongated coarse-level correction $\mathbf{s}_{\mu_1+1}^{l+1} := \mathbf{I}_l^{l+1}(\mathbf{x}_{\mu^l}^l - \mathbf{x}_0^l)$ has to be assessed before it is accepted on the level $l+1$. For this reason, we define a multilevel TR ratio as $\rho_{\mu_1+1}^{l+1} := \frac{h^{l+1}(\mathbf{x}_{\mu_1}^{l+1}) - h^{l+1}(\mathbf{x}_{\mu_1}^{l+1} + \mathbf{s}_{\mu_1+1}^{l+1})}{h^l(\mathbf{x}_0^l) - h^l(\mathbf{x}_{\mu^l}^l)}$. The correction $\mathbf{s}_{\mu_1+1}^{l+1}$ is accepted if $\rho_{\mu_1+1}^{l+1} > \eta_1$. If $\rho_{\mu_1+1}^{l+1} \leq \eta_1$, the correction $\mathbf{s}_{\mu_1+1}^{l+1}$ is rejected. Additionally, the TR radius has to be updated accordingly. At the end, the RMTR algorithm performs μ_2 post-smoothing steps at a given level l. This process is repeated on every level until the finest level is reached.

Construction of level-dependent objective functions and feasible sets

In this section, we discuss how to construct the objective function h^l and feasible set $\mathcal{F}^l$. For the finest level, $l = L$, we assume that $h^L := f$ and $\mathcal{F}^L := \mathcal{F}$. In coherence with the RMTR convergence theory [7], the coarse-level functions $\{h^l\}_{l=1}^{L-1}$ have to be constructed such that they are at least twice continuously differentiable, and at least first-order consistent with the function h^{l+1}. Here, we create a level-dependent objective function h^l as follows:

$$h^l(\mathbf{x}^l + \mathbf{s}^l) := \langle \mathbf{R}_{l+1}^l \nabla h_{\mu_1}^{l+1}, \mathbf{s}^l \rangle + \frac{1}{2} \langle \mathbf{s}^l, (\mathbf{R}_{l+1}^l \nabla^2 h_{\mu_1}^{l+1} \mathbf{I}_l^{l+1}) \mathbf{s}^l \rangle, \tag{2}$$

where $\mathbf{s}^l = \mathbf{x}^l - \mathbf{x}_0^l$, and $\mathbf{R}_{l+1}^l \nabla h_{\mu_1}^{l+1}$, $\mathbf{R}_{l+1}^l \nabla^2 h_{\mu_1}^{l+1} \mathbf{I}_l^{l+1}$ represent the restricted gradient and the Hessian from the level $l+1$, evaluated after μ_1 pre-smoothing steps, respectively. As we will see in Section 3, employing coarse-level models of this particular type allows for straightforward incorporation of the active set strategy within the multilevel settings.

For all levels $l < L$, the level-dependent feasible set $\mathcal{F}^l$ is created by intersecting the set $\mathcal{L}^l$ with the set $\mathcal{S}^l$, thus as $\mathcal{F}^l := \mathcal{L}^l \cap \mathcal{S}^l$. The purpose of the set $\mathcal{S}^l := \{\mathbf{x}^l \in \mathbb{R}^{n^l} \mid \mathbf{tl}^l \leq \mathbf{x}^l \leq \mathbf{tu}^l\}$ is to ensure that the size of the prolongated coarse-level correction remains bounded by the TR radius $\Delta_{\mu_1}^l$, i.e., $\|\mathbf{I}_l^{l+1} \mathbf{s}^l\|_\infty \leq \Delta_{\mu_1}^l$. To this end, we construct $\mathcal{S}^l$ by employing the projection rules especially designed for TR bounds in [7].

The function of the set $\mathcal{L}^l := \{\mathbf{x}^l \in \mathbb{R}^{n^l} \mid \mathbf{vl}^l \leq \mathbf{x}^l \leq \mathbf{vu}^l\}$ is to guarantee that the prolongated coarse-level correction produces a feasible trial point, i.e., $\mathbf{x}_{\mu_1}^{l+1} + \mathbf{I}_l^{l+1} \mathbf{s}^l \in \mathcal{F}^{l+1}$. Following [5, 7], we can construct $\mathbf{vl}^l, \mathbf{vu}^l$ in a component-wise manner as

$$\begin{aligned} (\mathbf{vl}^l)_k &:= (\mathbf{x}_0^l)_k + \max_{j \in \mathcal{N}^{l+1} \cap \dot{\omega}_k^l} [(\mathbf{vl}^{l+1} - \mathbf{x}_{\mu_1}^{l+1})_j], \\ (\mathbf{vu}^l)_k &:= (\mathbf{x}_0^l)_k + \min_{j \in \mathcal{N}^{l+1} \cap \dot{\omega}_k^l} [(\mathbf{vu}^{l+1} - \mathbf{x}_{\mu_1}^{l+1})_j], \end{aligned} \tag{3}$$

where $(\cdot)_k$ denotes the k-th component of a given vector. Note that the support ω_k^l of the basis function N_k^l (associated with k-th node of the mesh $\mathcal{T}^l$) determines which components of the variable bounds $\mathbf{vl}^{l+1}$ and $\mathbf{vu}^{l+1}$ have to be taken into account while constructing $\mathbf{vl}^l, \mathbf{vu}^l$.

3 Multilevel active-set trust-region (MASTR) method

In this section, we present how to incorporate the active-set strategy into the RMTR framework. The devised algorithm has a form of the standard V-cycle. The key idea behind the proposed MASTR method is to identify an active-set

$$\mathcal{A}^l(\mathbf{x}^l_{\mu_1}) := \{k \in \{1,\dots,n^l\} \mid (\mathbf{vl}^l)_k = (\mathbf{x}^l_{\mu_1})_k \text{ or } (\mathbf{vu}^l)_k = (\mathbf{x}^l_{\mu_1})_k\}, \tag{4}$$

before descending to the coarser level. Here, the vectors $\mathbf{vl}^l, \mathbf{vu}^l$ denote lower and upper bounds that define the set $\mathcal{L}^l$ and are obtained using formula (3). The components of the solution vector $\mathbf{x}^l_{\mu_1}$, that are active are then held fixed and cannot be altered by the coarser levels. To this end, the level-dependent objective functions $\{h^a\}_{a=1}^{l-1}$ and feasible sets $\{\mathcal{L}^a\}_{a=1}^{l-1}$ have to be constructed such that the minimization process on a given level yields coarse-level corrections, which fulfil this requirement. Following [12, 9], we construct $\{h^a\}_{a=1}^{l-1}$ and $\{\mathcal{L}^a\}_{a=1}^{l-1}$ using a truncated basis method.

Construction of truncated FE spaces

The truncated basis method [12] constructs truncated FE spaces $\{\widetilde{\mathcal{X}}^l\}_{l=1}^{L-1}$, spanned by truncated basis functions $\{\widetilde{N}^l_k\}_{k\in\mathcal{N}^l}$ by exploiting the fact that the basis functions on level l, can be written as linear combination of basis functions on level $l+1$, i.e., $N^l_k = \sum_{p=1}^{n^{l+1}} (\mathbf{I}^{l+1}_l)_{pk} N^{l+1}_p$. In contrast to the classical multilevel approaches with canonical Galerkin restriction, the actual shape and support of the truncated basis functions depend on the current fine-level iterate. In particular, the value of the truncated basis functions is set to zero at all active nodes of the finer levels, i.e., their support vanishes at the active nodes. More precisely, we can construct truncated basis functions in a recursive manner as

$$\widetilde{N}^l_k = \sum_{p=1}^{n^{l+1}} (\widetilde{\mathbf{I}}^{l+1}_l)_{pk} \widetilde{N}^{l+1}_p, \tag{5}$$

where $\widetilde{\mathbf{I}}^{l+1}_l$ is truncated prolongation operator defined by

$$(\widetilde{\mathbf{I}}^{l+1}_l)_{pk} = \begin{cases} 0, & \text{if } \ p \in \mathcal{A}^{l+1}(\mathbf{x}^l_{\mu_1}), \\ (\mathbf{I}^{l+1}_l)_{pk}, & \text{otherwise.} \end{cases} \tag{6}$$

The operator $\widetilde{\mathbf{I}}^{l+1}_l$ is obtained from the prolongation operator $\mathbf{I}^{l+1}_l$ by setting the p-th row of $\mathbf{I}^{l+1}_l$ to zero, for all $p \in \mathcal{A}^{l+1}(\mathbf{x}^l_{\mu_1})$. The application of $\widetilde{\mathbf{I}}^{l+1}_l$ in (5) removes contributions of basis functions associated with active nodes on level $l+1$, defined by the set $\mathcal{A}^{l+1}(\mathbf{x}^l_{\mu_1})$.

Remark 1 For the level $L-1$, the recursive formula (5) employs $\{N^L_k\}_{k\in\mathcal{N}^L}$, instead of $\{\tilde{N}^L_k\}_{k\in\mathcal{N}^L}$.

Construction of level-dependent objective functions and feasible sets

Using truncated FE spaces $\{\widetilde{\mathcal{X}}^l\}_{l=1}^{L-1}$, we can now construct level-dependent objective functions $\{h^l\}_{l=1}^{L-1}$ and feasible sets $\{\mathcal{F}^l\}_{l=1}^{L-1}$. In particular, for a given level $l < L$, the level-dependent objective function $h^l : \mathbb{R}^{n^l} \to \mathbb{R}$ is created as follows:

$$h^l(\mathbf{x}^l + \mathbf{s}^l) := \langle (\widetilde{\mathbf{I}}_l^{l+1})^T \nabla h_{\mu_1}^{l+1}, \mathbf{s}^l \rangle + \frac{1}{2} \langle \mathbf{s}^l, (\widetilde{\mathbf{I}}_l^{l+1})^T \nabla^2 h_{\mu_1}^{l+1} \widetilde{\mathbf{I}}_l^{l+1}) \mathbf{s}^l \rangle, \tag{7}$$

where we used truncated transfer operator $\widetilde{\mathbf{I}}_l^{l+1}$ to restrict gradient $\nabla h_{\mu_1}^{l+1}$ and Hessian $\nabla^2 h_{\mu_1}^{l+1}$ from level $l+1$ to level l. The application of $\widetilde{\mathbf{I}}_l^{l+1}$ in (7) removes the components of fine-level gradient/Hessian associated with the active-set $\mathcal{A}^{l+1}(\mathbf{x}_{\mu_1}^l)$. Please note that the formulation (7) does not require explicit representation of $\{\widetilde{\mathcal{X}}^l\}_{l=1}^{L-1}$.

The construction of each level-dependent feasible set $\mathcal{L}^l$ can be performed using projection rules defined by (3). However, formulas (3) are now determined by the support of the truncated basis functions, spanning $\widetilde{\mathcal{X}}^l$. Since the support of basis functions spanning $\widetilde{\mathcal{X}}^l$ is different from the support of the basis functions spanning $\mathcal{X}^l$, fewer components of a fine-level variable bounds are taken into account by (3). This yields less restrictive coarse-level constraints and allows for larger coarse grid corrections.

4 Numerical results

We study the performance of the proposed MASTR method using three numerical examples. Examples are defined on domain $\Omega := [0,1]^2$ with boundary $\Gamma = \partial\Omega$, decomposed into three parts: $\Gamma_l = \{0\} \times [0,1]$, $\Gamma_r = \{1\} \times [0,1]$, and $\Gamma_f = [0,1] \times \{0,1\}$. The discretization is performed using a uniform mesh and $\mathbb{Q}_1$ Lagrange finite elements.

Ex.1. MEMBRANE: Let us solve the following minimization problem [4]:

$$\begin{aligned} &\min_{u \in \mathcal{X}} \; f(u) := \frac{1}{2} \int_\Omega \|\nabla u(x)\|^2 \, dx + \int_\Omega u(x) \, dx, \\ &\text{subject to } \; \mathrm{lb}(x) \leq u, \quad \text{on } \Gamma_r, \end{aligned} \tag{8}$$

where $\mathcal{X} := \{u \in H^1(\Omega) \mid u = 0 \text{ on } \Gamma_l\}$. The lower bound lb is defined on the right part of the boundary, Γ_r, by the upper part of the circle with the radius, $r = 1$, and the center, $C = (1; -0.5; -1.3)$. Thus, the lower bound is defined as

$$\mathrm{lb}(x) = \begin{cases} (-2.6 + \sqrt{2.6^2 - 4((x_2 - 0.5)^2 - 1.0 + 1.3^2)})/2, & \text{if } x = 1, \\ -\infty, & \text{otherwise,} \end{cases}$$

where the symbols x_1, x_2 denote spatial coordinates.

Ex.2. IGNITION: Following [2, 10], we minimize following optimization problem:

$$\begin{aligned} &\min_{u \in \mathcal{X}} \; f(u) := \frac{1}{2} \int_\Omega \|\nabla u(x)\|^2 - (ue^u - e^u) \, dx - \int_\Omega f(x) u \, dx, \\ &\text{subject to } \; \mathrm{lb}(x) \leq u \leq \mathrm{ub}(x), \quad \text{a.e. in } \Omega. \end{aligned} \tag{9}$$

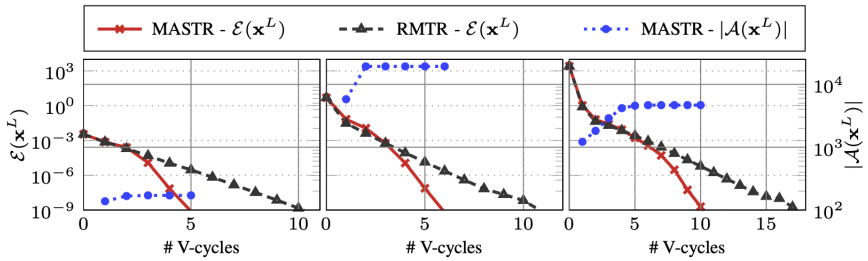

Fig. 1: The convergence of the MASTR method (red color) and RMTR without an active-set strategy (black color). The blue color illustrates the size of an active set detected by the MASTR algorithm. *Left:* MEMBRANE. *Middle:* IGNITION. *Right:* MOREBV.

The variable bounds and right-hand side are defined as

$$\begin{aligned} \mathrm{lb}(x) &= -8(x_1 - 7/16)^2 - 8(x_2 - 7/16)^2 + 0.2, \quad \mathrm{ub}(x) = 0.5, \\ f(x) &= (9\pi^2 + e^{(x_1^2 - x_1^3)\sin(3\pi x_2)}(x_1^2 - x_1^3) + 6x_1 - 2)\sin(3\pi x_1), \end{aligned}$$

where $\mathcal{X} := \{u \in H^1(\Omega) \mid u = 0 \text{ on } \Gamma\}$, and $f \in L^2(\Omega)$.

Ex.3. MOREBV: We consider the following non-convex minimization problem [6]:

$$\begin{aligned} &\min_{u \in \mathcal{X}} \; f(u) := \int_\Omega \|\Delta u(x) - 0.5(u(x) + \langle e, x\rangle + 1)^3\|_2^2 \, dx, \\ &\text{subject to } \; \mathrm{lb}(x) \le u, \quad \text{a.e. in } \Omega, \end{aligned} \tag{10}$$

where $\mathcal{X} := \{u \in H^1(\Omega) \mid u = 0 \text{ on } \Gamma\}$ and e denotes a unit vector. The lower bound is defined as $\mathrm{lb}(x) = \sin(5\pi x_1)\sin(\pi x_2)\sin(\pi(1 - x_1))\sin(\pi(1 - x_2))$, where x_1, x_2 denote spatial coordinates.

Convergence study

We compare the convergence behavior of the the proposed MASTR method with the standard RMTR method (without the active-set strategy). Both methods are implemented as part of the open-source library UTOPIA [17].

For all experiments, we consider the RMTR method configured with 6-levels. The multilevel hierarchy is obtained by uniformly refining the coarsest level mesh, consisting of 10×10 elements. This gives rise to 289×289 elements on the finest level, which corresponds to 83,521 dofs. The performed study considers an RMTR setup with one trust-region pre/post-smoothing step. On all levels $l > 1$, the trust-region subproblems (1) are solved using one iteration of successive coordinate minimization [7]. On the coarsest level, we employ the Semismooth-Newton method. The algorithms terminate, if $\mathcal{E}(\mathbf{x}^L) < 10^{-9}$ is satisfied. The criticality measure $\mathcal{E}(\mathbf{x})$ is defined as $\mathcal{E}(\mathbf{x}) := \|\mathcal{P}(\mathbf{x} - \nabla f(\mathbf{x})) - \mathbf{x}\|$, where $\mathcal{P}$ is the orthogonal projection onto the feasible set $\mathcal{F}$.

As we can see from Figure 1, using an active-set approach is beneficial, as it allows for significant speed up. We can also observe that during the active-set identification phase (first few V-cycles), both approaches are comparable. However, once the exact active-set is detected, MASTR accelerates and converges faster than standard RMTR.

Funding This work was funded by the Swiss National Science Foundation (SNF) under the project ML2 (grant no. 197041), Platform for Advanced and Scientific Computing (PASC) under the project ExaTrain and by the MATH+ (distinguished scholar R. Krause).

References

1. Brandt, A., Cryer, C.W.: Multigrid algorithms for the solution of linear complementarity problems arising from free boundary problems. SIAM journal on scientific and statistical computing **4**(4), 655–684 (1983)
2. Briggs, W.L., McCormick, S.F., et al.: A multigrid tutorial. Siam (2000)
3. Conn, A.R., Gould, N.I.M., Toint, P.L.: Trust Region Methods. MOS-SIAM Series on Optimization. SIAM (2000). DOI 10.1137/1.9780898719857
4. Domorádová, M., Dostál, Z.: Projector preconditioning for partially bound-constrained quadratic optimization. Numerical Linear Algebra with Applications **14**(10), 791–806 (2007)
5. Gelman, E., Mandel, J.: On multilevel iterative methods for optimization problems. Mathematical Programming **48**(1-3), 1–17 (1990)
6. Gratton, S., Mouffe, M., Sartenaer, A., Toint, P.L., Tomanos, D.: Numerical experience with a recursive trust-region method for multilevel nonlinear bound-constrained optimization. Optimization Methods and Software **25**(3), 359–386 (2010). DOI 10.1080/10556780903239295
7. Gratton, S., Mouffe, M., Toint, P., Weber Mendonca, M.: A recursive ℓ_∞-trust-region method for bound-constrained nonlinear optimization. IMA Journal of Numerical Analysis **28**(4), 827–861 (2008)
8. Hackbusch, W., Mittelmann, H.D.: On multi-grid methods for variational inequalities. Numerische Mathematik **42**(1), 65–76 (1983)
9. Hoppe, R.H., Kornhuber, R.: Adaptive multilevel methods for obstacle problems. SIAM journal on numerical analysis **31**(2), 301–323 (1994)
10. Kočvara, M., Mohammed, S.: A first-order multigrid method for bound-constrained convex optimization. Optimization Methods and Software **31**(3), 622–644 (2016)
11. Kopaničáková, A., Krause, R.: A recursive multilevel trust region method with application to fully monolithic phase-field models of brittle fracture. Computer Methods in Applied Mechanics and Engineering **360**, 112720 (2020)
12. Kornhuber, R.: Monotone multigrid methods for elliptic variational inequalities I. Numerische Mathematik **69**(2), 167–184 (1994)
13. Kornhuber, R., Krause, R.: Adaptive multigrid methods for Signorini's problem in linear elasticity. Computing and Visualization in Science **4**(1), 9–20 (2001)
14. Mandel, J.: A multilevel iterative method for symmetric, positive definite linear complementarity problems. applied mathematics and optimization **11**(1), 77–95 (1984)
15. Vallejos, M.: MGOPT with gradient projection method for solving bilinear elliptic optimal control problems. Computing **87**(1-2), 21–33 (2010)
16. Youett, J., Sander, O., Kornhuber, R.: A globally convergent filter-trust-region method for large deformation contact problems. SIAM Journal on Scientific Computing **41**(1), B114–B138 (2019)
17. Zulian, P., Kopaničáková, A., Nestola, M.C.G., Fink, A., Fadel, N., Rigazzi, A., Magri, V., Schneider, T., Botter, E., Mankau, J., Krause, R.: Utopia: A C++ embedded domain specific language for scientific computing. Git repository. https://bitbucket.org/zulianp/utopia (2016).