Newton-type multilevel optimization method

Ho, Chin Pang; Kocvara, Michal; Parpas, Panos

DOI:
10.1080/10556788.2019.1700256

License:
Other (please specify with Rights Statement)

Document Version
Peer reviewed version

Citation for published version (Harvard):
Ho, CP, Kocvara, M & Parpas, P 2019, 'Newton-type multilevel optimization method', *Optimization Methods and Software*. https://doi.org/10.1080/10556788.2019.1700256

ARTICLE TEMPLATE

Newton-type Multilevel Optimization Method

Chin Pang Ho[a] and Michal Kočvara[b,c] and Panos Parpas[d]

[a]School of Data Science, City University of Hong Kong, Hong Kong; [b]School of Mathematics, The University of Birmingham, United Kingdom; [c]Institute of Information Theory and Automation, Academy of Sciences of the Czech Republic, Prague, Czech Republic; [d]Department of Computing, Imperial College London, United Kingdom

ARTICLE HISTORY
Compiled November 22, 2019

ABSTRACT
Inspired by multigrid methods for linear systems of equations, multilevel optimization methods have been proposed to solve structured optimization problems. Multilevel methods make more assumptions regarding the structure of the optimization model, and as a result, they outperform single-level methods, especially for large-scale models. The impressive performance of multilevel optimization methods is an empirical observation, and no theoretical explanation has so far been proposed. In order to address this issue, we study the convergence properties of a multilevel method that is motivated by second-order methods. We take the first step toward establishing how the structure of an optimization problem is related to the convergence rate of multilevel algorithms.

KEYWORDS
Newton's method; multilevel algorithms; multigrid methods

1. Introduction

Multigrid methods are a well-known and established method for solving differential equations [3, 11, 13, 23, 24, 26]. When solving a differential equation using numerical methods, an approximation of the solution is obtained on a mesh via discretization. The computational cost of solving the discretized problem, however, varies and it depends on the choice of the mesh size used. Therefore, by considering different mesh sizes, a hierarchy of discretized models can be defined. In general, a more accurate solution can be obtained when a smaller mesh size is chosen, which results in a discretized problem in higher dimensions. We shall follow the traditional terminology in the multigrid literature and call a *fine model* to be the discretization in which its solution is sufficiently close to the solution of the original differential equation; otherwise we call it a *coarse model* [3]. The main idea of multigrid methods is to make use of the geometric similarity between different discretizations. In particular, during the iterative process of computing the solution of the fine model, one replaces part of the computations with the information from coarse models. The advantages of using multigrid methods are twofold. Firstly, coarse models are in lower dimensions compared to the fine model, and so the computational cost is reduced. Secondly and interestingly, the corrections generated by

a. Email: clint.ho@cityu.edu.hk
b. Email: m.kocvara@bham.ac.uk
d. Email: p.parpas@imperial.ac.uk

the coarse model and fine model are in fact complementary. It has been shown that using the fine model is effective in reducing the high frequency components of the residual (error) but ineffective in reducing the low frequency component of the error. Those low frequency components of the error, however, will become high frequency errors in the coarse model. Thus, they could be eliminated effectively using coarse models [3, 23].

This idea of multigrid was extended to optimization algorithms. Nash [19] proposed a multigrid framework for unconstrained infinite-dimensional convex optimization problems. Examples of such problems could be found in the area of optimal control. Following the idea of Nash, many multigrid optimization methods were further developed [10, 16–20, 25]. In particular, Wen and Goldfarb [25] provided a line search-based multigrid optimization algorithm under the framework in [19], and further extended the framework to nonconvex problems. Gratton et al. [10] provided a sophisticated trust-region version of multigrid optimization algorithms, which they called it multiscale algorithm. In this paper, we will consistently use the name *multilevel algorithms* for all these optimization algorithms, but we emphasize that the terms multilevel, multigrid, and multiscale were used interchangeably in different papers. On the other hand, we keep the name *multigrid methods* for the conventional multigrid methods that solve linear or nonlinear equations that are discretizations arising from partial differential equations (PDEs).

It is worth mentioning that different multilevel algorithms were developed beyond infinite-dimensional problems, such as Markov decision processes [14], semidefinite programming [6], artificial neural networks [5], and composite optimization for both the convex [15] and non-convex case [21]. Also, Calandra et al. [4] proposed a multilevel algorithm for adaptive cubic regularization method recently. The above algorithms all have the same aim: to speed up the computations by making use of the geometric similarity between different models in the hierarchy.

Numerical results from the papers cited above show that multilevel algorithms can take advantage of the geometric similarity between different discretizations. In particular, they outperform other state-of-the-art optimization methods, especially for large scale models. However, to the best of our knowledge, no theoretical result exists that rigorously explain these empirical observations. The contributions of this paper are:

- We provide a complete view of line search multilevel algorithm, and in particular, we connect the general framework of the multilevel algorithm with classical optimization algorithms, such as variable metric methods and block-coordinate type methods.
- We analyze the Newton-type multilevel model. The key feature of the Newton-type multilevel model is that a coarse model is created from the first and second order information of the fine model. We will call this algorithm the **Ne**wton-type **M**ultilevel **O**ptimization (**NeMO**). A global convergence analysis of NeMO is provided.
- We propose to use the composite rate for analysis of the local convergence of NeMO. As we will show later, neither linear convergence nor quadratic convergence is suitable when studying the local convergence of NeMO.
- We study the composite rate of NeMO in a case study of infinite dimensional optimization problems. We show that the linear component of the composite rate is inversely proportional to the smoothness of the residual, which agrees with the findings in conventional multigrid methods.

The rest of this paper is structured as follows: In Section 2 we provide background material for multilevel algorithms. In Section 3, we study the convergence of NeMO. We first derive the global convergence rate of NeMO, and then show that NeMO exhibits composite convergence when the current incumbent is sufficiently close to the optimum. A composite convergence rate is defined as a linear combination of linear convergence and quadratic convergence, and we denote r_1 and r_2 as the coefficient of linear rate and quadratic rate, respectively. In Section 4, we compute r_1 in problems arising from discretizations of one-dimensional PDE problems and show the relationship between r_1 and the structure of the problem. In Section 5 we illustrate the convergence of NeMO using several numerical examples.

2. Multilevel Models

In this section a broad view of the general multilevel framework will be provided. We start with a basic setting and the core idea of multilevel algorithms in [10, 17, 25]. Then we provide the formulation and details of the core topic of this paper, namely Newton-type multilevel model.

2.1. Problem Formulation

In this paper we are interested in solving,

$$\min_{\mathbf{x}_h \in \mathbb{R}^N} f_h(\mathbf{x}_h), \tag{1}$$

where $\mathbf{x}_h \in \mathbb{R}^N$, and the function $f_h : \mathbb{R}^N \to \mathbb{R}$ is continuous, differentiable, and strongly convex. We clarify the use of the subscript h. Throughout this paper, the lower case h represents an object or property that this is associated with the***fine*** model, i.e. the model we actually want to solve. To use multilevel methods, one needs to formulate a hierarchy of models with reduced dimensions called the ***coarse*** models. We only consider two models in the hierarchy: fine and coarse. In the same manner of using subscript h, we assign the upper case H to represent the association with coarse model. We assign N and n ($n \leq N$) to be the dimensions of fine model and coarse model, respectively. For instance, any vector that is within the space $\mathbb{R}^N$ is denoted with subscript h, and similarly, any vector with subscript H is within the space $\mathbb{R}^n$.

Assumption 2.1. There exists constants μ_h, L_h, and M_h such that

$$\mu_h \mathbf{I} \preccurlyeq \nabla^2 f_h(\mathbf{x}) \preccurlyeq L_h \mathbf{I}, \quad \forall \mathbf{x}_h \in \mathbb{R}^n, \tag{2}$$

and

$$\|\nabla^2 f_h(\mathbf{x}_h) - \nabla^2 f_h(\mathbf{y}_h)\| \leq M_h \|\mathbf{x}_h - \mathbf{y}_h\|, \quad \forall \mathbf{x}_h, \mathbf{y}_h \in \mathbb{R}^n. \tag{3}$$

Equation (2) implies

$$\|\nabla f_h(\mathbf{x}_h) - \nabla f_h(\mathbf{y}_h)\| \leq L_h \|\mathbf{x}_h - \mathbf{y}_h\|, \quad \forall \mathbf{x}_h, \mathbf{y}_h \in \mathbb{R}^n.$$

The above assumptions will be used throughout the paper.

Multilevel methods require mapping information across different dimensions. To this end, we define a matrix $\mathbf{P} \in \mathbb{R}^{N \times n}$ to be the prolongation operator which maps information from coarse to fine, and we define a matrix $\mathbf{R} \in \mathbb{R}^{n \times N}$ to be the restriction operator which maps information from fine to coarse. We make the following assumption on $\mathbf{P}$ and $\mathbf{R}$.

Assumption 2.2. The restriction operator $\mathbf{R}$ is the transpose of the prolongation operator $\mathbf{P}$

up to a constant c. That is,

$$\mathbf{P} = c\mathbf{R}^T, \qquad c > 0.$$

Without loss of generality, we take $c = 1$ throughout this paper to simplify the use of notation for the analysis. We also assume any useful (non-zero) information in the coarse model will not become zero after prolongation and thus make the following assumption.

Assumption 2.3. The prolongation operator $\mathbf{P}$ has full column rank, and so

$$\text{rank}(\mathbf{P}) = n.$$

Notice that Assumption 2.2 and 2.3 are standard assumptions for multilevel methods [3, 12, 25]. Since $\mathbf{P}$ has full column rank, we define the pseudoinverse and its norm

$$\mathbf{P}^+ = (\mathbf{RP})^{-1}\mathbf{R}, \quad \text{and} \quad \xi = \|\mathbf{P}^+\|. \tag{4}$$

The coarse model is constructed in the following manner. Suppose in the k^{th} iteration we have an incumbent solution $\mathbf{x}_{h,k}$ and gradient $\nabla f_{h,k} \triangleq \nabla f_h(\mathbf{x}_{h,k})$, then the corresponding coarse model is,

$$\min_{\mathbf{x}_H \in \mathbf{R}^n} \phi_H(\mathbf{x}_H) \triangleq f_H(\mathbf{x}_H) + \langle \mathbf{v}_H, \mathbf{x}_H - \mathbf{x}_{H,0} \rangle, \tag{5}$$

where,

$$\mathbf{v}_H \quad \triangleq \quad -\nabla f_{H,0} + \mathbf{R}\nabla f_{h,k},$$

$\mathbf{x}_{H,0} = \mathbf{R}\mathbf{x}_{h,k}$, and $f_H : \mathbb{R}^n \to \mathbb{R}$ is a function to be specified later. Similar to $\nabla f_{h,k}$, we denote $\nabla^2 f_{H,0} \triangleq \nabla^2 f_H(\mathbf{x}_{H,0})$ and $\nabla \phi_{H,0} \triangleq \nabla \phi_H(\mathbf{x}_{H,0})$ to simplify notation. We emphasize the construction of the coarse model (5) is well known and it is not original in this paper. See for example [10, 17, 25]. Note that when constructing the coarse model (5), one needs to add an additional linear term to $f_H(\mathbf{x}_H)$. This linear term ensures the following is satisfied,

$$\nabla \phi_{H,0} = \mathbf{R}\nabla f_{h,k}. \tag{6}$$

For infinite-dimensional optimization problems, one can define f_h and f_H using discretization with different mesh sizes. In general, f_h is a function that approximates the original problem sufficiently well, and that can be achieved using a small mesh size. Based on geometric similarity between discretizations with different meshes, $f_h \approx f_H$ even though $n \leq N$.

However, we want to emphasize $f_h \approx f_H$ is not a necessary requirement when using multilevel methods. In principle, $f_H(\mathbf{x}_H)$ can be any function. Newton-type multilevel model, as we will show later, is a quadratic model where f_H is chosen to be a quadratic approximation of f_h at some $\mathbf{x}_h$.

2.2. The General Multilevel Algorithm

The main idea of multilevel algorithms is to use the coarse model to compute search directions. When a direction from the coarse model is used we call the iteration a ***coarse correction step***.

When using coarse correction step, we compute the direction by solving the corresponding coarse model (5) and perform the update,

$$\mathbf{x}_{h,k+1} = \mathbf{x}_{h,k} + \alpha_{h,k}\hat{\mathbf{d}}_{h,k},$$

with

$$\hat{\mathbf{d}}_{h,k} \triangleq \mathbf{P}(\mathbf{x}_{H,\star} - \mathbf{x}_{H,0}), \tag{7}$$

where $\mathbf{x}_{H,\star}$ is the solution of the coarse model, and $\alpha_{h,k} \in \mathbb{R}^+$ is the stepsize. We clarify that the "hat" in $\hat{\mathbf{d}}_{h,k}$ is used to identify a coarse correction step.

We should emphasize that $\mathbf{x}_{H,\star}$ in (7) can be replaced by $\mathbf{x}_{H,r}$ for $r = 1, 2, \ldots$, i.e., the incumbent solution of the coarse mode (5) after the r^{th} iterations of some iterative method. However, for the purpose of this paper and simplicity, we ignore this case and we let (7) be the (exact) coarse correction step.

It is known that the coarse correction step $\hat{\mathbf{d}}_{h,k}$ is a descent direction for f_h if f_H is convex. The following lemma states this argument rigorously. Even though the proof is provided in [25], we provide it with our notation for the completeness of this paper.

Lemma 2.4 ([25]). *If f_H is a convex function, then the coarse correction step is a descent direction for f_h at $\mathbf{x}_{h,k}$. In particular, in the k^{th} iteration,*

$$\nabla f_{h,k}^T \hat{\mathbf{d}}_{h,k} \leq \phi_{H,\star} - \phi_{H,0} \leq 0.$$

Proof.

$$\begin{aligned}
\nabla f_{h,k}^T \hat{\mathbf{d}}_{h,k} &= \nabla f_{h,k}^T \mathbf{R}^T \left(\mathbf{x}_{H,\star} - \mathbf{x}_{H,0}\right), \\
&= \left(\mathbf{R}\nabla f_{h,k}\right)^T \left(\mathbf{x}_{H,\star} - \mathbf{x}_{H,0}\right), \\
&= \nabla \phi_{H,0}^T \left(\mathbf{x}_{H,\star} - \mathbf{x}_{H,0}\right), \\
&\leq \phi_{H,\star} - \phi_{H,0}.
\end{aligned}$$

as required, where the last inequality holds because ϕ_H is a convex function. □

Even though Lemma 2.4 states that $\hat{\mathbf{d}}_{h,k}$ is a descent direction, using coarse correction step solely is not sufficient to solve the fine model (1).

Proposition 2.5. *Assume that f_H is convex. Suppose $\nabla f_{h,k} \neq 0$ and $\nabla f_{h,k} \in$ null$(\mathbf{R})$, then the coarse correction step*

$$\hat{\mathbf{d}}_{h,k} = 0.$$

Proof. From (6), $\mathbf{x}_{H,\star} = \mathbf{x}_{H,0}$ when $\mathbf{R}\nabla f_{h,k} = 0$. Thus, $\hat{\mathbf{d}}_{h,k} = \mathbf{P}(\mathbf{x}_{H,\star} - \mathbf{x}_{H,0}) = 0$. □

Recall that $\mathbf{R} \in \mathbb{R}^{n\times N}$, and so for $n < N$, a coarse correction step could be zero and make no progress even when the first order necessary condition $\nabla f_h = 0$ has not been satisfied.

2.2.1. Fine Correction Step

Two approaches can be used when coarse correction step is not progressing nor effective. The first approach is to compute directions using standard optimization methods. We call such step the ***fine correction step***. As opposed to coarse correction step $\hat{\mathbf{d}}_{h,k}$, we abandon the use of "hat" for all fine correction steps and denote them as $\mathbf{d}_{h,k}$'s. To be precise, we can compute $\mathbf{d}_{h,k}$ using the following,

$$\begin{aligned} \mathbf{d}_{h,k} &= \arg\min_{\mathbf{d}} \frac{1}{2}\langle \mathbf{d}, \mathbf{Q}\mathbf{d}\rangle + \langle \nabla f_{h,k}, \mathbf{d}\rangle, \\ &= -\mathbf{Q}^{-1}\nabla f_{h,k}. \end{aligned} \tag{8}$$

where $\mathbf{Q} \in \mathbb{R}^{N\times N}$ is a positive definite matrix. When $\mathbf{Q} = \mathbf{I}$, $\mathbf{d}_{h,k}$ is the steepest descent search direction. When $\mathbf{Q} = \nabla^2 f_{h,k}$, $\mathbf{d}_{h,k}$ is the search direction by Newton's method. When $\mathbf{Q}$ is an approximation of the Hessian, then $\mathbf{d}_{h,k}$ is the quasi-Newton search direction.

We perform a fine correction step when a coarse correction step may not be effective. That is, when one of the following conditions holds:

$$\|\mathbf{R}\nabla f_{h,k}\| < \kappa\|\nabla f_{h,k}\| \quad \text{or} \quad \|\mathbf{R}\nabla f_{h,k}\| < \epsilon, \tag{9}$$

where $\kappa \in (0, \min(1, \|\mathbf{R}\|))$, and $\epsilon \in (0, 1)$. The above criteria prevent the use of the coarse model when $\mathbf{x}_{H,0} \approx \mathbf{x}_{H,\star}$, i.e. the coarse correction step $\hat{\mathbf{d}}_{h,k}$ is close to $\mathbf{0}$. We point out that these criteria were also proposed in [25]. We also make the following assumption on the fine correction step throughout this paper.

Assumption 2.6. There exists strictly positive constants $\nu_h, \zeta_h > 0$ such that

$$\|\mathbf{d}_{h,k}\| \leq \nu_h\|\nabla f_{h,k}\|, \quad \text{and} \quad -\nabla f_{h,k}^T\mathbf{d}_{h,k} \geq \zeta_h\|\nabla f_{h,k}\|^2,$$

where $\mathbf{d}_{h,k}$ is a fine correction step. As a consequence, there exists a constant $\Lambda_h > 0$ such that

$$f_{h,k} - f_{h,k+1} \geq \Lambda_h\|\nabla f_{h,k}\|^2,$$

where $f_{h,k+1}$ is updated using a fine correction step.

As we will show later, Assumption 2.6 is not restrictive, and Λ_h is known for well-known cases like gradient descent, Newton method, etc. Using the combination of fine and coarse correction steps is the standard approach in multilevel methods, especially for PDE-based optimization problems [10, 17, 25].

2.2.2. *Multiple* $\mathbf{P}$*'s and* $\mathbf{R}$*'s*

The second approach to overcome issue of ineffective coarse correction step is by creating multiple coarse models with different $\mathbf{P}$'s and $\mathbf{R}$'s.

Proposition 2.7. *Suppose $\mathbf{R}_1, \mathbf{R}_2, \dots, \mathbf{R}_p$ are all restriction operators that satisfy Assumption 2.2 and 2.3, where $\mathbf{R}_i \in \mathbb{R}^{n_i\times N}$ for $i = 1, 2, \dots, p$. Denote $\mathcal{S}$ to be a set that contains the rows of $\mathbf{R}_i$'s in $\mathbb{R}^N$, for $i = 1, 2, \dots, p$. If*

$$\textit{span}(\mathcal{S}) = \mathbb{R}^N,$$

then for $\nabla f_{h,k} \neq 0$ there exists at least one $\mathbf{R}_j \in \{\mathbf{R}_i\}_{i=1}^p$ such that

$$\hat{\mathbf{d}}_{h,k} \neq 0 \quad \textit{and} \quad \nabla f_{h,k}^T\hat{\mathbf{d}}_{h,k} < 0,$$

where $\hat{\mathbf{d}}_{h,k}$ *is the coarse correction step computed using* $\mathbf{R}_j$*.*

Proof. Since span$(\mathcal{S}) = \mathbb{R}^N$, then for $\nabla f_{h,k} \neq 0$, there exists one $\mathbf{R}_j$ such that $\mathbf{R}_j \nabla f_{h,k} \neq 0$. So the corresponding coarse model would have $\mathbf{x}_{H,\star} \neq \mathbf{x}_{H,0}$, and thus $\hat{\mathbf{d}}_{h,k_j} \neq 0$. □

Proposition 2.7 shows that if the rows of the restriction operators $\mathbf{R}_i$'s span $\mathbb{R}^N$, then at least one coarse correction step from these restriction operators would be nonzero and thus effective. In each iteration, one could use the similar idea as in (9) to rule out ineffective coarse models. However, this checking process could be expensive for large scale problems with large p (number of restriction operators). To omit this checking process, one could choose the following alternatives.

i. **Cyclical approach**: choose $\mathbf{R}_1, \mathbf{R}_2, \ldots, \mathbf{R}_p$ in order at each iteration, and choose $\mathbf{R}_1$ after $\mathbf{R}_p$.
ii. **Probabilistic approach**: assign a probability mass function with $\{\mathbf{R}_i\}_{i=1}^p$ as a sample space, and choose the coarse model randomly based on the mass function. The mass function has to be strictly positive for each $\mathbf{R}_i$'s.

We point out that this idea of using multiple coarse models is related to domain decomposition methods, which solve (non-)linear equations arising from PDEs. Domain decomposition methods partition the problem domain into several sub-domains, and thus decompose the original problem into several smaller problems. We refer the reader to [7] for more details about domain decomposition methods.

2.3. Connection with Variable Metric Methods

Using the above multilevel framework, in the rest of this section we will introduce different versions of multilevel algorithms: variable metric methods, block-coordinate descent, and stochastic variance reduced gradient. At the end of this section we will introduce the Newton-type multilevel model, which is an interesting case of the multilevel framework.

Recall that for variable metric methods, the direction $\mathbf{d}_{h,k}$ is computed by solving

$$\begin{aligned} \mathbf{d}_{h,k} &= \arg\min_{\mathbf{d}} \frac{1}{2}\langle \mathbf{d}, \mathbf{Q}\mathbf{d}\rangle + \langle \nabla f_{h,k}, \mathbf{d}\rangle, \\ &= -\mathbf{Q}^{-1}\nabla f_{h,k}. \end{aligned} \tag{10}$$

where $\mathbf{Q} \in \mathbb{R}^{N\times N}$ is a positive definite matrix. When $\mathbf{Q} = \mathbf{I}$, $\mathbf{d}_{h,k}$ is the steepest descent search direction. When $\mathbf{Q} = \nabla^2 f_{h,k}$, $\mathbf{d}_{h,k}$ is the search direction by Newton's method. When $\mathbf{Q}$ is an approximation of the Hessian, then $\mathbf{d}_{h,k}$ is the quasi-Newton search direction.

To show the connections between multilevel methods and variable metric methods, consider the following f_H.

$$f_H(\mathbf{x}_H) = \frac{1}{2}\langle \mathbf{x}_H - \mathbf{x}_{H,0}, \mathbf{Q}_H(\mathbf{x}_H - \mathbf{x}_{H,0})\rangle, \tag{11}$$

where $\mathbf{Q}_H \in \mathbb{R}^{n\times n}$, and $\mathbf{x}_{H,0} = \mathbf{R}\mathbf{x}_{h,k}$ as defined in (5). Applying the definition of the coarse model (5), we obtain,

$$\min_{\mathbf{x}_H \in \mathbf{R}^n} \phi_H(\mathbf{x}_H) = \frac{1}{2}\langle \mathbf{x}_H - \mathbf{x}_{H,0}, \mathbf{Q}_H(\mathbf{x}_H - \mathbf{x}_{H,0})\rangle + \langle \mathbf{R}\nabla f_{h,k}, \mathbf{x}_H - \mathbf{x}_{H,0}\rangle. \tag{12}$$

Thus from the definition in (7), the associated coarse correction step is,

$$\hat{\mathbf{d}}_{h,k} = \mathbf{P}\left(\arg\min_{\mathbf{d}_H \in \mathbf{R}^n} \underbrace{\frac{1}{2}\langle \mathbf{d}_H, \mathbf{Q}_H \mathbf{d}_H\rangle + \langle \mathbf{R}\nabla f_{h,k}, \mathbf{d}_H\rangle}_{\mathbf{d}_H = \mathbf{x}_H - \mathbf{x}_{H,0}}\right) = -\mathbf{P}\mathbf{Q}_H^{-1}\mathbf{R}\nabla f_{h,k}. \tag{13}$$

Therefore, with this specific f_H in (11), the resulting coarse model (12) is analogous to variable metric methods. In a naive case where $n = N$ and $\mathbf{P} = \mathbf{R} = \mathbf{I}$, the corresponding coarse correction step (13) would be the same as steepest descent direction, Newton direction, and quasi-Newton direction for $\mathbf{Q}_H$ that is identity matrix, Hessian, and approximation of Hessian, respectively.

2.4. Connection with Block-coordinate Descent

Interestingly, the coarse model (12) is also related to block-coordinate type methods. Suppose we have p coarse models with prolongation and restriction operators, $\{\mathbf{P}_i\}_{i=1}^p$ and $\{\mathbf{R}_i\}_{i=1}^p$, respectively. For each coarse model, we let (11) be the corresponding f_H with $\mathbf{Q}_H = \mathbf{I}$, and we further restrict our setting with the following properties.

1. $\mathbf{P}_i \in \mathbb{R}^{N \times n_i}$, $\forall i = 1, 2, \dots, p$.
2. $\mathbf{P}_i = \mathbf{R}_i^T$, $\forall i = 1, 2, \dots, p$.
3. $[\mathbf{P}_1\, \mathbf{P}_2 \dots \mathbf{P}_p] = \mathbf{I}$.

From (13), the above setting results in $\hat{\mathbf{d}}_{h,k_i} = -\mathbf{P}_i\mathbf{R}_i\nabla f_{h,k}$, where $\hat{\mathbf{d}}_{h,k_i}$ is the coarse correction step for the i^{th} model. Notice that

$$(\mathbf{P}_i\mathbf{R}_i\nabla f_{h,k})_j = \begin{cases} (\nabla f_{h,k})_j & \text{if } \sum_{q=1}^{i-1} n_q < j \leq \sum_{q=1}^{i} n_q, \\ 0 & \text{otherwise .} \end{cases}$$

Therefore, $\hat{\mathbf{d}}_{h,k_i}$ is equivalent to a block-coordinate descent update [1]. When $n_i = 1$, for $i = 1, 2, \dots, p$, it becomes a coordinate descent method. When $1 < n_i < N$, for $i = 1, 2, \dots, p$, it becomes a block-coordinate descent. When $\mathbf{P}_i$'s and $\mathbf{R}_i$'s are chosen using the cyclical approach, then it would be a cyclical (block)-coordinate descent. When $\mathbf{P}_i$'s and $\mathbf{R}_i$'s are chosen using the probabilistic approach, then it would be a randomized (block)-coordinate descent method.

2.5. The Newton-type Multilevel Model

We end this section with the core topic of this paper - the Newton-type multilevel model. The Newton-type multilevel coarse model is a special case of (12) where,

$$\mathbf{Q}_H = \nabla_H^2 f_{h,k} \triangleq \mathbf{R}\nabla^2 f_{h,k}\mathbf{P}, \tag{14}$$

and so the Newton-type multilevel (coarse) model is,

$$\min_{\mathbf{x}_H \in \mathbf{R}^n} \phi_H(\mathbf{x}_H) = \frac{1}{2}\langle \mathbf{x}_H - \mathbf{x}_{H,0}, \nabla_H^2 f_{h,k}(\mathbf{x}_H - \mathbf{x}_{H,0})\rangle + \langle \mathbf{R}\nabla f_{h,k}, \mathbf{x}_H - \mathbf{x}_{H,0}\rangle. \tag{15}$$

According to (13), the corresponding coarse correction step is

$$\hat{\mathbf{d}}_{h,k} = -\mathbf{P}[\mathbf{R}\nabla^2 f_{h,k}\mathbf{P}]^{-1}\mathbf{R}\nabla f_{h,k} = -\mathbf{P}[\nabla_H^2 f_{h,k}]^{-1}\mathbf{R}\nabla f_{h,k}. \tag{16}$$

In the context of multilevel optimization, to the best of our knowledge, this coarse model was first considered in [10]. In [10] a trust-region type multilevel method is proposed to solve PDE-based optimization problems, and the Newton-type multilevel model is described as a "radical strategy". In a later paper from Gratton et al. [9], a trust-region type multilevel method was tested numerically, and the Newton-type multilevel model showed promising numerical results.

It is worth mentioning that the above coarse correction step is equivalent to the solution of the system of linear equations,

$$\mathbf{R}\nabla^2 f_{h,k}\mathbf{P}\mathbf{d}_H = -\mathbf{R}\nabla f_{h,k}. \tag{17}$$

which is the general case of the Newton's method in which $\mathbf{P} = \mathbf{R} = \mathbf{I}$. Using Assumption 2.3, we can show that $\nabla_H^2 f_{h,k}$ is positive definite, and so equation (17) has a unique solution.

Proposition 2.8. $\mathbf{R}\nabla^2 f_h(\mathbf{x}_h)\mathbf{P}$ *is positive definite, and in particular,*

$$\mu_h \xi^{-2}\mathbf{I} \preceq \mathbf{R}\nabla^2 f_h(\mathbf{x}_h)\mathbf{P} \preceq L_h\omega^2\mathbf{I}$$

where $\omega = \max\{\|\mathbf{P}\|, \|\mathbf{R}\|\}$ *and* $\xi = \|\mathbf{P}^+\|$.

Proof.

$$\mathbf{x}^T\left(\mathbf{R}\nabla^2 f_h(\mathbf{x}_h)\mathbf{P}\right)\mathbf{x} = (\mathbf{P}\mathbf{x})^T\nabla^2 f_h(\mathbf{x}_h)(\mathbf{P}\mathbf{x}) \leq L_h\|\mathbf{P}\mathbf{x}\|^2 \leq L_h\omega^2\|\mathbf{x}\|^2.$$

Also,

$$\mathbf{x}^T\left(\mathbf{R}\nabla^2 f_h(\mathbf{x}_h)\mathbf{P}\right)\mathbf{x} = (\mathbf{P}\mathbf{x})^T\nabla^2 f_h(\mathbf{x}_h)(\mathbf{P}\mathbf{x}) \geq \mu_h\|\mathbf{P}\mathbf{x}\|^2 \geq \frac{\mu_h}{\|\mathbf{P}^+\|^2}\|\mathbf{x}\|^2 = \frac{\mu_h}{\xi^2}\|\mathbf{x}\|^2.$$

So we obtain the desired result. □

3. Convergence of NeMO

In this section we analyze NeMO (Algorithm 1). The fine correction steps in Algorithm 1 are deployed by a variable metric method, and an Armijo rule is used as stepsize strategy for both fine and coarse correction steps. We will first show that Algorithm 1 achieves a sublinear rate of convergence. We then analyze the maximum number of coarse correction steps that would be taken by Algorithm 1, and the condition that when the coarse correction steps yield quadratic reduction in the gradients in the subspace. At the end of this section, we will provide the composite convergence rate for the coarse correction steps.

To provide convergence properties when the coarse correction step is used, the following quantity will be used

$$\chi_{H,k} \triangleq [(\mathbf{R}\nabla f_{h,k})^T[\nabla_H^2 f_{h,k}]^{-1}\mathbf{R}\nabla f_{h,k}]^{1/2}.$$

Algorithm 1 NeMO

```
Input: $\mathbf{P} \in \mathbb{R}^{N\times n}$ and $\mathbf{R} \in \mathbb{R}^{N\times n}$ which satisfy Assumption 2.2 and 2.3, $\kappa \in (0, \min(1, \|\mathbf{R}\|))$, $\epsilon$, $\rho_1 \in (0, 0.5)$, $\beta_{ls} \in (0, 1)$.
Initialization: $\mathbf{x}_{h,0} \in \mathbb{R}^N$
for $k = 0, 1, 2, \dots$ do
    Compute the direction
        $\mathbf{d} = \begin{cases} \hat{\mathbf{d}}_{h,k} \text{ in (16)} & \text{if } \|\mathbf{R}\nabla f_{h,k}\| > \kappa\|\nabla f_{h,k}\| \text{ and } \|\mathbf{R}\nabla f_{h,k}\| > \epsilon, \\ \mathbf{d}_{h,k} \text{ in (10)} & \text{otherwise.} \end{cases}$
    Find the smallest $q \in \mathbb{N}$ such that for stepsize $\alpha_{h,k} = \beta_{ls}^q$,
        $f_h(\mathbf{x}_{h,k} + \alpha_{h,k}\mathbf{d}) \leq f_{h,k} + \rho_1\alpha_{h,k}\nabla^T f_{h,k}\mathbf{d}.$
    Update
        $\mathbf{x}_{h,k+1} \triangleq \mathbf{x}_{h,k} + \alpha_{h,k}\mathbf{d}.$
end for
```

Notice that $\chi_{H,k}$ is analogous to the Newton decrement, which is used to study the convergence of the Newton method [2]. In particular, $\chi_{H,k}$ has the following properties.

1. $\nabla f_{h,k}^T \hat{\mathbf{d}}_{h,k} = -\chi_{H,k}^2$.
2. $\hat{\mathbf{d}}_{h,k}^T \nabla^2 f_{h,k} \hat{\mathbf{d}}_{h,k} = \chi_{H,k}^2$.

We omit the proofs of the above properies since these can be done by using direct computation and the definition of $\chi_{H,k}$.

3.1. The Sublinear Rate

We will show that Algorithm 1 will achieve a sublinear rate of convergence. We will deploy the techniques from [1] and [2]. Starting with the following lemma, we state reduction in function value using coarse correction steps. We would like to clarify that even though NeMO is considered as a special case in [25], we take advantage of this simplification and specification to provide analysis with results that are easier to interpret. In particular, the analysis of stepsizes $\alpha_{h,k}$'s in [25] relies on the maximum number of iterations taken. This result is unfavorable and unnecessary for the setting we consider.

Lemma 3.1. *The coarse correction step $\hat{\mathbf{d}}_{h,k}$ in Algorithm 1 will lead to reduction in function value*

$$f_{h,k} - f_h(\mathbf{x}_{h,k} + \alpha_{h,k}\hat{\mathbf{d}}_{h,k}) \geq \frac{\rho_1\kappa^2\beta_{ls}\mu_h}{\omega^2 L_h^2}\|\nabla f_{h,k}\|^2,$$

where ρ_1, κ, and β_{ls} are user-defined parameters in Algorithm 1. L_h and μ_h are defined in Assumption 2.1. ω is defined in Proposition 2.8.

Proof. By convexity,

$$
\begin{aligned}
f_h(\mathbf{x}_{h,k} + \alpha\hat{\mathbf{d}}_{h,k}) &\le f_{h,k} + \alpha\langle \nabla f_{h,k}, \hat{\mathbf{d}}_{h,k}\rangle + \frac{L_h}{2}\alpha^2\|\hat{\mathbf{d}}_{h,k}\|^2, \\
&\le f_{h,k} - \alpha\chi_{H,k}^2 + \frac{L_h}{2\mu_h}\alpha^2\chi_{H,k}^2,
\end{aligned}
$$

since

$$\mu_h\|\hat{\mathbf{d}}_{h,k}\|^2 \le \hat{\mathbf{d}}_{h,k}^T\nabla^2 f_h(x_k)\hat{\mathbf{d}}_{h,k} = \chi_{H,k}^2.$$

Notice that for $\hat{\alpha} = \mu_h/L_h$, we have

$$-\hat{\alpha} + \frac{L_h}{2\mu_h}\hat{\alpha}^2 = -\hat{\alpha} + \frac{L_h}{2\mu_h}\frac{\mu_h}{L_h}\hat{\alpha} = -\frac{1}{2}\hat{\alpha},$$

and

$$
\begin{aligned}
f_h(\mathbf{x}_{h,k} + \hat{\alpha}\hat{\mathbf{d}}_{h,k}) &\le f_{h,k} - \frac{\hat{\alpha}}{2}\chi_{H,k}^2, \\
&\le f_{h,k} + \frac{\hat{\alpha}}{2}\nabla f_{h,k}^T\hat{\mathbf{d}}_{h,k}, \\
&< f_{h,k} + \rho_1\hat{\alpha}\nabla f_{h,k}^T\hat{\mathbf{d}}_{h,k},
\end{aligned}
$$

which satisfies the Armijo condition. Therefore, line search will return stepsize $\alpha_{h,k} \ge \hat{\alpha} = (\beta_{ls}\mu_h)/L_h$. Using the fact that

$$\frac{1}{\omega^2 L_h}\|\mathbf{R}\nabla f_h(x_k)\|^2 \le (\mathbf{R}\nabla f_{h,k})^T[\nabla_H^2 f_{h,k}]^{-1}\mathbf{R}\nabla f_{h,k} = \chi_{H,k}^2,$$

we obtain

$$
\begin{aligned}
f_h(\mathbf{x}_{h,k} + \alpha_{h,k}\hat{\mathbf{d}}_{h,k}) - f_{h,k} &\le \rho_1\alpha_{h,k}\nabla f_{h,k}^T\hat{\mathbf{d}}_{h,k}, \\
&\le -\rho_1\hat{\alpha}\chi_{H,k}^2, \\
&\le -\rho_1\frac{\beta_{ls}\mu_h}{\omega^2 L_h^2}\|\mathbf{R}\nabla f_{h,k}\|^2, \\
&\le -\frac{\rho_1\kappa^2\beta_{ls}\mu_h}{\omega^2 L_h^2}\|\nabla f_{h,k}\|^2,
\end{aligned}
$$

as required. □

Using the result in Lemma 3.1, we derive the guaranteed reduction in function value in the following two lemmas.

Lemma 3.2. *Let* $\Lambda \triangleq \min\left\{\Lambda_h, \frac{\rho_1\kappa^2\beta_{ls}\mu_h}{\omega^2 L_h^2}\right\}$*, then the step* $\mathbf{d}$ *in Algorithm 1 will lead to*

$$f_{h,k} - f_{h,k+1} \ge \Lambda\|\nabla f_{h,k}\|^2,$$

where ρ_1, κ, and β_{ls} are user-defined parameters in Algorithm 1. L_h and μ_h are defined in Assumption 2.1. Λ_h is defined in Assumption 2.6. ω is defined in Proposition 2.8.

Proof. This is a direct result from Lemma 3.1 and Assumption 2.6. □

Let $\mathbf{x}_{h,\star}$ denote the exact solution of (1) and let $f_{h,\star} \triangleq f(\mathbf{x}_{h,\star})$.

Lemma 3.3. *Suppose*

$$\mathcal{R}(\mathbf{x}_{h,0}) \triangleq \max_{\mathbf{x}_h \in \mathbb{R}^N} \{\|\mathbf{x}_h - \mathbf{x}_{h,\star}\| : f_h(\mathbf{x}_h) \leq f_h(\mathbf{x}_{h,0})\},$$

the step in Algorithm 1 will guarantee

$$f_{h,k} - f_{h,k+1} \geq \frac{\Lambda}{\mathcal{R}^2(\mathbf{x}_{h,0})} \left(f_{h,k} - f_{h,\star}\right)^2,$$

where Λ is defined in Lemma 3.2.

Proof. By convexity, for $k = 0, 1, 2, \ldots,$

$$\begin{aligned} f_{h,k} - f_{h,\star} &\leq \langle \nabla f_{h,k}, \mathbf{x}_{h,k} - \mathbf{x}_{h,\star} \rangle, \\ &\leq \|\nabla f_{h,k}\| \, \|\mathbf{x}_{h,k} - \mathbf{x}_{h,\star}\|, \\ &\leq \mathcal{R}(\mathbf{x}_{h,0}) \|\nabla f_{h,k}\|. \end{aligned}$$

Using Lemma 3.2, we have

$$\begin{aligned} f_{h,k} - f_{h,\star} &\leq \mathcal{R}(\mathbf{x}_{h,0}) \sqrt{\Lambda^{-1} \left(f_{h,k} - f_{h,k+1}\right)}, \\ \left(\frac{f_{h,k} - f_{h,\star}}{\mathcal{R}(\mathbf{x}_{h,0})}\right)^2 &\leq \Lambda^{-1} \left(f_{h,k} - f_{h,k+1}\right), \\ \Lambda \left(\frac{f_{h,k} - f_{h,\star}}{\mathcal{R}(\mathbf{x}_{h,0})}\right)^2 &\leq f_{h,k} - f_{h,k+1}, \end{aligned}$$

as required. □

The constant Λ in Lemma 3.3 depends on Λ_h, which is introduced in Assumption 2.6. This constant depends on both the fine correction step chosen and the user-defined parameter ρ_1 in Armijo rule. For instance,

$$\Lambda_h = \begin{cases} \dfrac{\rho_1 \mu_h}{L_h^2} & \text{if} \quad \mathbf{d}_{h,k} = -[\nabla^2 f_{h,k}]^{-1} \nabla f_{h,k}, \\ \dfrac{\rho_1}{L_h} & \text{if} \quad \mathbf{d}_{h,k} = -\nabla f_{h,k}. \end{cases}$$

The above results can be derived via direct computation on bounding the Armijo condition. In order to derive the convergence rate in this section, we use the following lemma on nonnegative scalar sequences.

Lemma 3.4. *[1] Let $\{A_k\}_{k \geq 0}$ be a nonnegative sequence of the real numbers satisfying*

$$A_k - A_{k+1} \geq \gamma A_k^2, \quad k = 0, 1, 2, \ldots,$$

and

$$A_0 \le \frac{1}{q\gamma}$$

for some positive γ and q. Then

$$A_k \le \frac{1}{\gamma(k+q)}, \quad k = 0, 1, 2, \ldots,$$

and so

$$A_k \le \frac{1}{\gamma k}, \quad k = 0, 1, 2, \ldots.$$

Proof. See Lemma 3.5 in [1]. □

Combining the above results, we obtain the rate of convergence.

Theorem 3.5. *Let $\{\mathbf{x}_k\}_{k\ge 0}$ be the sequence that is generated by Algorithm 1. Then,*

$$f_{h,k} - f_{h,\star} \le \frac{\mathcal{R}^2(\mathbf{x}_{h,0})}{\Lambda}\frac{1}{2+k},$$

where Λ and $\mathcal{R}(\cdot)$ are defined as in Lemma 3.2 and 3.3, respectively.

Proof. From Lemma 3.3,

$$f_{h,k} - f_{h,k+1} \ge \frac{\Lambda}{\mathcal{R}^2(\mathbf{x}_{h,0})}\left(f_{h,k} - f_{h,\star}\right)^2.$$

and so

$$(f_{h,k} - f_{h,\star}) - (f_{h,k+1} - f_{h,\star}) \ge \frac{\Lambda}{\mathcal{R}^2(\mathbf{x}_{h,0})}\left(f_{h,k} - f_{h,\star}\right)^2.$$

Also, we have

$$\begin{aligned} f_{h,0} - f_{h,\star} \le \frac{L_h}{2}\|\mathbf{x}_{h,0} - \mathbf{x}_{h,\star}\|^2 \le \frac{L_h}{2}\mathcal{R}^2(\mathbf{x}_{h,0}) \le \frac{L_h^2\mathcal{R}^2(\mathbf{x}_{h,0})}{2\mu_h} \quad &\le \quad \frac{L_h^2\mathcal{R}^2(\mathbf{x}_{h,0})}{2\mu_h\beta_{ls}\kappa^2\rho_1}, \\ &\le \quad \frac{\mathcal{R}^2(\mathbf{x}_{h,0})}{2\Lambda}, \end{aligned}$$

where the first inequality holds because of first order condition and the definition of L_h in Assumption 2.1. Let's $A_k \triangleq f_{h,k} - f_{h,\star}$, $\gamma \triangleq \frac{\Lambda}{R^2(\mathbf{x}_{h,0})}$, and $q \triangleq 2$. By applying Lemma 3.4, we have

$$f_{h,k} - f_{h,\star} \le \frac{\mathcal{R}^2(\mathbf{x}_{h,0})}{\Lambda}\frac{1}{2+k},$$

as required. □

Theorem 3.5 provides the sublinear convergence of Algorithm 1. We emphasize that the rate is inversely proportional to $\Lambda = \min\{\Lambda_h, \rho_1\kappa^2\mu_h/L_h^2\}$, and so small κ would result in slow convergence. Therefore, even though κ could be arbitrary small, it is not desirable in terms of worse case complexity. Note that κ is a user-defined parameter for determining whether the coarse correction step should be used. If κ is chosen to be too large, then it is less likely that the coarse correction step would be used. In the extreme case where $\kappa \geq \|\mathbf{R}\|$, the coarse correction step would not be deployed because,

$$\|\mathbf{R}\nabla f_{h,k}\| \leq \|\mathbf{R}\|\|\nabla f_{h,k}\|,$$

and so Algorithm 1 reduces to the standard variable metric method. Therefore, there is a trade-off between the worse case complexity and the likelihood that the coarse correction step is deployed.

3.2. Maximum Number of Iterations of Coarse Correction Step

We now discuss the maximum number of coarse correction steps in Algorithm 1. The following lemma will state the sufficient conditions for not taking any coarse correction step.

Lemma 3.6. *No coarse correction step in Algorithm 1 will be taken when*

$$\|\nabla f_{h,k}\| \leq \frac{\epsilon}{\omega},$$

where $\omega = \max\{\|\mathbf{P}\|, \|\mathbf{R}\|\}$*, and* ϵ *is a user-defined parameter in Algorithm 1.*

Proof. Recall that in Algorithm 1, the coarse step is only taken when $\|\mathbf{R}\nabla f_{h,k}\| > \epsilon$. We have,

$$\|\mathbf{R}\nabla f_{h,k}\| \leq \omega\|\nabla f_{h,k}\| \leq \omega\frac{\epsilon}{\omega} = \epsilon,$$

and so no coarse correction step will be taken. □

The above lemma states the condition when the coarse correction step would not be performed. We then investigate the maximum number of iterations to achieve that sufficient condition.

Lemma 3.7. *Let* $\{\mathbf{x}_k\}_{k\geq 0}$ *be a sequence generated by Algorithm 1. Then,* $\forall\bar{\epsilon}, \bar{k} > 0$ *such that,*

$$\bar{k} \geq \left(\frac{1}{\bar{\epsilon}}\right)^2 \frac{\mathcal{R}^2(\mathbf{x}_{h,0})}{\Lambda^2} - 2,$$

we obtain

$$\|\nabla f_h(\mathbf{x}_{h,\bar{k}})\| \leq \bar{\epsilon},$$

where Λ *and* $\mathcal{R}(\cdot)$ *are defined as in Lemma 3.2 and 3.3, respectively.*

Proof. From Lemma 3.2, we know that

$$\Lambda\|\nabla f_{h,k}\|^2 \quad \leq \quad f_{h,k} - f_{h,k+1}.$$

Also, from Theorem 3.5, we have,

$$f_{h,k} - f_{h,\star} \leq \frac{\mathcal{R}^2(\mathbf{x}_{h,0})}{\Lambda} \frac{1}{2+k}.$$

Therefore,

$$\begin{aligned} \|\nabla f_{h,k}\|^2 &\leq \frac{1}{\Lambda}\left(f_{h,k} - f_{h,k+1}\right), \\ &\leq \frac{1}{\Lambda}\left(f_{h,k} - f_{h,\star}\right), \\ &\leq \frac{\mathcal{R}^2(\mathbf{x}_{h,0})}{\Lambda^2} \frac{1}{2+k}. \end{aligned}$$

For

$$k = \left(\frac{1}{\bar{\epsilon}}\right)^2 \frac{\mathcal{R}^2(\mathbf{x}_{h,0})}{\Lambda^2} - 2,$$

we have

$$\|\nabla f_{h,k}\| \leq \sqrt{\frac{\mathcal{R}^2(\mathbf{x}_{h,0})}{\Lambda^2} \frac{1}{2+k}} \leq \sqrt{\frac{\mathcal{R}^2(\mathbf{x}_{h,0})}{\Lambda^2} \left(\bar{\epsilon}\right)^2 \frac{\Lambda^2}{\mathcal{R}^2(\mathbf{x}_{h,0})}} = \bar{\epsilon},$$

as required. □

By integrating the above results, we obtain the maximum number of iterations to achieve $\|\nabla f_{h,k}\| \leq \epsilon/\omega$. That is, no coarse correction step will be taken after

$$\left(\frac{\omega}{\epsilon}\right)^2 \frac{\mathcal{R}^2(\mathbf{x}_{h,0})}{\Lambda^2} - 2 \quad \text{iterations.}$$

Notice that the smaller ϵ, the more coarse correction step will be taken. Depending on the choice of $\mathbf{d}_{h,k}$, the choice of ϵ could be different. For example, if $\mathbf{d}_{h,k}$ is chosen as the Newton step where $\mathbf{d}_{h,k} = -[\nabla^2 f_{h,k}]^{-1} \nabla f_{h,k}$, one good choice of ϵ could be $3\omega(1-2\rho_1)\mu_h^2/L_h$ if μ_h and L_h are known. This is because Newton's method achieves quadratic rate of convergence when $\|\nabla f_{h,k}\| \leq 3(1-2\rho_1)\mu_h^2/L_h$ [2]. Therefore, for such ϵ, no coarse correction step would be taken when the Newton method is in its quadratically convergent phase.

3.3. Quadratic Phase in Subspace

We now state the required condition for stepsize $\alpha_{h,k} = 1$, and then we will show that when $\|\mathbf{R}\nabla f_{h,k}\|$ is sufficiently small, the coarse correction step would reduce $\|\mathbf{R}\nabla f_{h,k}\|$ quadratically. The results below are analogous to the analysis of the Newton's method in [2].

Lemma 3.8. *Suppose coarse correction step $\hat{\mathbf{d}}_{h,k}$ in Algorithm 1 is taken, then $\alpha_{h,k} = 1$ when*

$$\|\mathbf{R}\nabla f_{h,k}\| \leq \eta = \frac{3\mu_h^2}{M_h}(1-2\rho_1),$$

where ρ_1 is an user-defined parameter in Algorithm 1. M_h and μ_h are defined in Assumption 2.1.

Proof. By Lipschitz continuity (3),

$$\|\nabla^2 f_h(\mathbf{x}_{h,k} + \alpha\hat{\mathbf{d}}_{h,k}) - \nabla^2 f_{h,k}\| \leq \alpha M_h \|\hat{\mathbf{d}}_{h,k}\|,$$

which implies

$$\|\hat{\mathbf{d}}_{h,k}^T(\nabla^2 f_h(\mathbf{x}_{h,k} + \alpha\hat{\mathbf{d}}_{h,k}) - \nabla^2 f_{h,k})\hat{\mathbf{d}}_{h,k}\| \leq \alpha M_h \|\hat{\mathbf{d}}_{h,k}\|^3.$$

Let $\tilde{f}(\alpha) = f_h(\mathbf{x}_{h,k} + \alpha\hat{\mathbf{d}}_{h,k})$, then the above inequality can be rewritten as

$$|\tilde{f}''(\alpha) - \tilde{f}''(0)| \leq \alpha M_h \|\hat{\mathbf{d}}_{h,k}\|^3,$$

and so

$$\tilde{f}''(\alpha) \leq \tilde{f}''(0) + \alpha M_h \|\hat{\mathbf{d}}_{h,k}\|^3.$$

Since $\tilde{f}''(0) = \hat{\mathbf{d}}_{h,k}^T \nabla^2 f_{h,k} \hat{\mathbf{d}}_{h,k} = \chi_{H,k}^2$,

$$\tilde{f}''(\alpha) \leq \chi_{H,k}^2 + \alpha M_h \|\hat{\mathbf{d}}_{h,k}\|^3.$$

By integration,

$$\tilde{f}'(\alpha) \leq \tilde{f}'(0) + \alpha\chi_{H,k}^2 + (\alpha^2/2) M_h \|\hat{\mathbf{d}}_{h,k}\|^3.$$

Similarly, $\tilde{f}'(0) = \nabla f_{h,k}^T \hat{\mathbf{d}}_{h,k} = -\chi_{H,k}^2$, and so

$$\tilde{f}'(\alpha) \leq -\chi_{H,k}^2 + \alpha\chi_{H,k}^2 + (\alpha^2/2) M_h \|\hat{\mathbf{d}}_{h,k}\|^3.$$

Integrating the above inequality, we obtain

$$\tilde{f}(\alpha) \leq \tilde{f}(0) - \alpha\chi_{H,k}^2 + (\alpha^2/2)\chi_{H,k}^2 + (\alpha^3/6) M_h \|\hat{\mathbf{d}}_{h,k}\|^3.$$

Recall that $\mu_h \|\hat{\mathbf{d}}_{h,k}\|^2 \leq \hat{\mathbf{d}}_{h,k}^T \nabla^2 f_{h,k} \hat{\mathbf{d}}_{h,k} = \chi_{H,k}^2$; thus,

$$\tilde{f}(\alpha) \leq \tilde{f}(0) - \alpha\chi_{H,k}^2 + \frac{\alpha^2}{2}\chi_{H,k}^2 + \frac{\alpha^3 M_h}{6\mu_h^{3/2}}\chi_{H,k}^3.$$

Let $\alpha = 1$,

$$\begin{aligned} \tilde{f}(1) - \tilde{f}(0) &\leq -\chi_{H,k}^2 + \frac{1}{2}\chi_{H,k}^2 + \frac{M_h}{6\mu_h^{3/2}}\chi_{H,k}^3, \\ &\leq -\left(\frac{1}{2} - \frac{M_h}{6\mu_h^{3/2}}\chi_{H,k}\right)\chi_{H,k}^2. \end{aligned}$$

Using the fact that

$$\|\mathbf{R}\nabla f_{h,k}\| \leq \eta = \frac{3\mu_h^2}{M_h}(1-2\rho_1),$$

and

$$\chi_{H,k} = ((\mathbf{R}\nabla f_{h,k})^T[\nabla_H^2 f_{h,k}]^{-1}\mathbf{R}\nabla f_{h,k})^{1/2} \leq \frac{1}{\sqrt{\mu_h}}\|\mathbf{R}\nabla f_{h,k}\|,$$

we have

$$\chi_{H,k} \leq \frac{3\mu_h^{3/2}}{M_h}(1-2\rho_1) \quad \Longleftrightarrow \quad \rho_1 \leq \frac{1}{2} - \frac{M_h}{6\mu_h^{3/2}}\chi_{H,k}.$$

Therefore,

$$\tilde{f}(1) - \tilde{f}(0) \leq -\rho_1\chi_{H,k}^2 = \rho_1\nabla f_{h,k}^T\hat{\mathbf{d}}_{h,k},$$

and we have $\alpha_{h,k} = 1$ when $\|\mathbf{R}\nabla f_{h,k}\| \leq \eta$. □

The above lemma yields the following theorem.

Theorem 3.9. *Suppose the coarse correction step $\hat{\mathbf{d}}_{h,k}$ in Algorithm 1 is taken and $\alpha_{h,k} = 1$, then*

$$\|\mathbf{R}\nabla f_{h,k+1}\| \leq \frac{\omega^3\xi^4 M_h}{2\mu_h^2}\|\mathbf{R}\nabla f_{h,k}\|^2,$$

where M_h and μ_h are defined in Assumption 2.1, $\omega = \max\{\|\mathbf{P}\|, \|\mathbf{R}\|\}$ and $\xi = \|\mathbf{P}^+\|$.

Proof. Since $\alpha_{h,k} = 1$, we have

$$\begin{aligned}
\|\mathbf{R}\nabla f_{h,k+1}\| &= \|\mathbf{R}\nabla f_h(\mathbf{x}_{h,k} + \hat{\mathbf{d}}_{h,k}) - \mathbf{R}\nabla f_{h,k} - \mathbf{R}\nabla^2 f_{h,k}\mathbf{P}\tilde{\mathbf{d}}_{H,i^\star}\| \\
&\leq \|\mathbf{R}\|\;\|\nabla f_h(\mathbf{x}_{h,k} + \hat{\mathbf{d}}_{h,k}) - \nabla f_{h,k} - \nabla^2 f_{h,k}\hat{\mathbf{d}}_{h,k}\| \\
&\leq \omega\left\|\int_0^1 (\nabla^2 f_h(\mathbf{x}_{h,k} + t\hat{\mathbf{d}}_{h,k}) - \nabla^2 f_{h,k})\hat{\mathbf{d}}_{h,k}\,dt\right\| \\
&\leq \omega\frac{M_h}{2}\|\hat{\mathbf{d}}_{h,k}\|^2,
\end{aligned}$$

where $\tilde{\mathbf{d}}_{H,i^\star}$ is the direction $\hat{\mathbf{d}}_{h,k}$ at coarse level, i.e. $\mathbf{P}\tilde{\mathbf{d}}_{H,i^\star} = \hat{\mathbf{d}}_{h,k}$. Notice that

$$\begin{aligned}
\|\hat{\mathbf{d}}_{h,k}\| &= \|\mathbf{P}[\mathbf{R}\nabla^2 f_{h,k}\mathbf{P}]^{-1}\mathbf{R}\nabla f_{h,k}\| \\
&\leq \|\mathbf{P}\|\;\|[\mathbf{R}\nabla^2 f_{h,k}\mathbf{P}]^{-1}\|\;\|\mathbf{R}\nabla f_{h,k}\| \\
&\leq \frac{\omega\xi^2}{\mu_h}\|\mathbf{R}\nabla f_{h,k}\|.
\end{aligned}$$

Thus,

$$\|\mathbf{R}\nabla f_{h,k+1}\| \leq \frac{\omega^3\xi^4 M_h}{2\mu_h^2}\|\mathbf{R}\nabla f_{h,k}\|^2,$$

as required. □

The above theorem states the quadratic convergence of $\|\nabla f_{h,k}\|$ within the subspace range$(\mathbf{R})$. However, it does not give insight in the convergence behaviour on the full space $\mathbb{R}^N$. To address this, we study the composite rate of convergence in the next section.

3.4. Composite Convergence Rate

At the end of this section, we study the convergence properties of the coarse correction step when the incumbent is sufficiently close to the solution. In particular, we deploy the idea of composite convergence rate in [8], and consider the convergence of the coarse correction step as a combination of linear and quadratic convergence.

The reason of proving composite convergence is due to the broadness of NeMO. Suppose that $\mathbf{P} = \mathbf{R} = \mathbf{I}$, then the coarse correction step in NeMO becomes Newton's method. In such case we expect quadratic convergence when the incumbent is sufficiently close to the solution. On the other hand, suppose $\mathbf{P}$ is any column of $\mathbf{I}$ and $\mathbf{R} = \mathbf{P}^T$, then the coarse correction step is a (weighted) coordinate descent direction. One should not expect more than linear convergence in that case. Therefore, both quadratic convergence and linear convergence are not suitable for NeMO, and one needs the combination of them. In this paper, we propose to use a composite convergence, and show that it can better explain the convergence of NeMO.

We would like to emphasize the difference between our setting compared to [8]. To the best of our knowledge, composite convergence rate was used in [8] to study subsampled Newton methods for machine learning problems without dimensionality reduction. In this paper, the class of problems that we consider is not restricted to machine learning, and we focus on the Newton-type multilevel model, which is a reduced dimension model. The results presented in this section are not direct results of the approach in [8]. In particular, if the exact analysis of [8] is taken, the derived composite rate would not be useful in our setting, because the coefficient of the linear component would be greater than 1.

Theorem 3.10. *Suppose the coarse correction step $\hat{\mathbf{d}}_{h,k}$ in Algorithm 1 is taken and $\alpha_{h,k} = 1$, then*

$$\begin{aligned}\|\mathbf{x}_{h,k+1} - \mathbf{x}_{h,\star}\| \leq \|\mathbf{I} - \mathbf{P}[\nabla_H^2 f_{h,k}]^{-1}\mathbf{R}\nabla^2 f_{h,k}\|\|(\mathbf{I} - \mathbf{P}\mathbf{R})(\mathbf{x}_{h,k} - \mathbf{x}_{h,\star})\| \\ + \frac{M_h\omega^2\xi^2}{2\mu_h}\|\mathbf{x}_{h,k} - \mathbf{x}_{h,\star}\|^2, \quad (18)\end{aligned}$$

where M_h and μ_h are defined in Assumption 2.1, $\omega = \max\{\|\mathbf{P}\|, \|\mathbf{R}\|\}$ and $\xi = \|\mathbf{P}^+\|$. The operator ∇_H^2 is defined in (14).

Proof. Denote

$$\tilde{\mathbf{Q}} = \int_0^1 \nabla^2 f(\mathbf{x}_{h,\star} - t(\mathbf{x}_{h,k} - \mathbf{x}_{h,\star}))\mathrm{d}t,$$

we have

$$\begin{aligned}
\mathbf{x}_{h,k+1} - \mathbf{x}_{h,\star} &= \mathbf{x}_{h,k} - \mathbf{x}_{h,\star} - \mathbf{P}[\nabla^2_H f_{h,k}]^{-1}\mathbf{R}\nabla f_{h,k}, \\
&= \mathbf{x}_{h,k} - \mathbf{x}_{h,\star} - \mathbf{P}[\nabla^2_H f_{h,k}]^{-1}\mathbf{R}\tilde{\mathbf{Q}}(\mathbf{x}_{h,k} - \mathbf{x}_{h,\star}), \\
&= \left(\mathbf{I} - \mathbf{P}[\nabla^2_H f_{h,k}]^{-1}\mathbf{R}\tilde{\mathbf{Q}}\right)(\mathbf{x}_{h,k} - \mathbf{x}_{h,\star}), \\
&= \left(\mathbf{I} - \mathbf{P}[\nabla^2_H f_{h,k}]^{-1}\mathbf{R}\nabla^2 f_{h,k}\right)(\mathbf{x}_{h,k} - \mathbf{x}_{h,\star}) \\
&\quad + \left(\mathbf{P}[\nabla^2_H f_{h,k}]^{-1}\mathbf{R}\nabla^2 f_{h,k} - \mathbf{P}[\nabla^2_H f_{h,k}]^{-1}\mathbf{R}\tilde{\mathbf{Q}}\right)(\mathbf{x}_{h,k} - \mathbf{x}_{h,\star}), \\
&= \left(\mathbf{I} - \mathbf{P}[\nabla^2_H f_{h,k}]^{-1}\mathbf{R}\nabla^2 f_{h,k}\right)(\mathbf{I} - \mathbf{P}\mathbf{R})(\mathbf{x}_{h,k} - \mathbf{x}_{h,\star}) \\
&\quad + \mathbf{P}[\nabla^2_H f_{h,k}]^{-1}\mathbf{R}\left(\nabla^2 f_{h,k} - \tilde{\mathbf{Q}}\right)(\mathbf{x}_{h,k} - \mathbf{x}_{h,\star}).
\end{aligned}$$

Note that

$$\|\nabla^2 f_{h,k} - \tilde{\mathbf{Q}}\| = \left\|\nabla^2 f_{h,k} - \int_0^1 \nabla^2 f(\mathbf{x}_{h,\star} - t(\mathbf{x}_{h,k} - \mathbf{x}_{h,\star}))\mathrm{d}t\right\| \leq \frac{M_h}{2}\|\mathbf{x}_{h,k} - \mathbf{x}_{h,\star}\|.$$

Therefore,

$$\begin{aligned}
\|\mathbf{x}_{h,k+1} - \mathbf{x}_{h,\star}\| &\leq \|\mathbf{I} - \mathbf{P}[\nabla^2_H f_{h,k}]^{-1}\mathbf{R}\nabla^2 f_{h,k}\|\|(\mathbf{I} - \mathbf{P}\mathbf{R})(\mathbf{x}_{h,k} - \mathbf{x}_{h,\star})\| \\
&\quad + \|\mathbf{P}[\nabla^2_H f_{h,k}]^{-1}\mathbf{R}\|\frac{M_h}{2}\|\mathbf{x}_{h,k} - \mathbf{x}_{h,\star}\|^2, \\
&\leq \|\mathbf{I} - \mathbf{P}[\nabla^2_H f_{h,k}]^{-1}\mathbf{R}\nabla^2 f_{h,k}\|\|(\mathbf{I} - \mathbf{P}\mathbf{R})(\mathbf{x}_{h,k} - \mathbf{x}_{h,\star})\| \\
&\quad + \frac{M_h\omega^2\xi^2}{2\mu_h}\|\mathbf{x}_{h,k} - \mathbf{x}_{h,\star}\|^2,
\end{aligned}$$

as required. □

Theorem 3.10 provides the composite convergence rate for the coarse correction step. However, some terms remain unclear, in particular $\|\mathbf{I} - \mathbf{P}[\nabla^2_H f_{h,k}]^{-1}\mathbf{R}\nabla^2 f_{h,k}\|$. Notice that in the case when rank$(\mathbf{P}) = N$ (i.e. $\mathbf{P}$ is invertible),

$$\begin{aligned}
\|\mathbf{I} - \mathbf{P}[\nabla^2_H f_{h,k}]^{-1}\mathbf{R}\nabla^2 f_{h,k}\| &= \|\mathbf{I} - \mathbf{P}[\mathbf{R}\nabla^2 f_{h,k}\mathbf{P}]^{-1}\mathbf{R}\nabla^2 f_{h,k}\|, \\
&= \|\mathbf{I} - \mathbf{P}\mathbf{P}^{-1}[\nabla^2 f_{h,k}]^{-1}\mathbf{R}^{-1}\mathbf{R}\nabla^2 f_{h,k}\|, \\
&= 0.
\end{aligned}$$

It is intuitive to consider that $\|\mathbf{I} - \mathbf{P}[\nabla^2_H f_{h,k}]^{-1}\mathbf{R}\nabla^2 f_{h,k}\|$ should be small and less than 1 when rank$(\mathbf{P})$ is close to but not equal to N. However, the above intuition is not true, and we prove this in the following lemma.

Lemma 3.11. *Suppose rank*$(\mathbf{P}) \neq N$*, then*

$$1 \leq \|\mathbf{I} - \mathbf{P}[\nabla^2_H f_{h,k}]^{-1}\mathbf{R}\nabla^2 f_{h,k}\| \leq \sqrt{\frac{L_h}{\mu_h}},$$

where L_h and μ_h are defined in Assumption 2.1. The operator ∇^2_H is defined in (14).

Proof. Since $\nabla^2 f_{h,k}$ is a positive definite matrix, consider the eigendecomposition of $\nabla^2 f_{h,k}$,

$$\nabla^2 f_{h,k} = \mathbf{U\Sigma U}^T,$$

where $\mathbf{\Sigma}$ is a diagonal matrix containing the eigenvalues of $\nabla^2 f_{h,k}$, and $\mathbf{U}$ is a orthogonal matrix where its columns are eigenvectors of $\nabla^2 f_{h,k}$. We then have

$$\begin{aligned}
& \mathbf{I} - \mathbf{P}[\nabla^2_H f_{h,k}]^{-1}\mathbf{R}\nabla^2 f_{h,k} \\
= \;& \mathbf{I} - \mathbf{P}[\mathbf{R}\nabla^2 f_{h,k}\mathbf{P}]^{-1}\mathbf{R}\nabla^2 f_{h,k}, \\
= \;& \mathbf{U\Sigma}^{-1/2}\mathbf{\Sigma}^{1/2}\mathbf{U}^T - \mathbf{U\Sigma}^{-1/2}\mathbf{\Sigma}^{1/2}\mathbf{U}^T\mathbf{P}[\mathbf{RU\Sigma}^{1/2}\mathbf{\Sigma}^{1/2}\mathbf{U}^T\mathbf{P}]^{-1}\mathbf{RU\Sigma}^{1/2}\mathbf{\Sigma}^{1/2}\mathbf{U}^T, \\
= \;& \mathbf{U\Sigma}^{-1/2}\mathbf{\Sigma}^{1/2}\mathbf{U}^T \\
& -\mathbf{U\Sigma}^{-1/2}(\mathbf{\Sigma}^{1/2}\mathbf{U}^T\mathbf{P})[(\mathbf{\Sigma}^{1/2}\mathbf{U}^T\mathbf{P})^T(\mathbf{\Sigma}^{1/2}\mathbf{U}^T\mathbf{P})]^{-1}(\mathbf{\Sigma}^{1/2}\mathbf{U}^T\mathbf{P})^T\mathbf{\Sigma}^{1/2}\mathbf{U}^T, \\
= \;& \mathbf{U\Sigma}^{-1/2}(\mathbf{I} - \mathbf{\Gamma}_{\mathbf{\Sigma}^{1/2}\mathbf{U}^T\mathbf{P}})\mathbf{\Sigma}^{1/2}\mathbf{U}^T,
\end{aligned}$$

where $\mathbf{\Gamma}_{\mathbf{\Sigma}^{1/2}\mathbf{U}^T\mathbf{P}}$ is the orthogonal projection operator onto the range of $\mathbf{\Sigma}^{1/2}\mathbf{U}^T\mathbf{P}$, and so

$$\begin{aligned}
\|\mathbf{I} - \mathbf{P}[\nabla^2_H f_{h,k}]^{-1}\mathbf{R}\nabla^2 f_{h,k}\| &= \|\mathbf{U\Sigma}^{-1/2}(\mathbf{I} - \mathbf{\Gamma}_{\mathbf{\Sigma}^{1/2}\mathbf{U}^T\mathbf{P}})\mathbf{\Sigma}^{1/2}\mathbf{U}^T\|, \\
&= \|\mathbf{\Sigma}^{-1/2}(\mathbf{I} - \mathbf{\Gamma}_{\mathbf{\Sigma}^{1/2}\mathbf{U}^T\mathbf{P}})\mathbf{\Sigma}^{1/2}\|.
\end{aligned}$$

For the upper bound, we have

$$\|\mathbf{\Sigma}^{-1/2}(\mathbf{I} - \mathbf{\Gamma}_{\mathbf{\Sigma}^{1/2}\mathbf{U}^T\mathbf{P}})\mathbf{\Sigma}^{1/2}\| \leq \|\mathbf{\Sigma}^{-1/2}\|\|(\mathbf{I} - \mathbf{\Gamma}_{\mathbf{\Sigma}^{1/2}\mathbf{U}^T\mathbf{P}})\|\|\mathbf{\Sigma}^{1/2}\| \leq \sqrt{\frac{L_h}{\mu_h}},$$

since $\mathbf{I} - \mathbf{\Gamma}_{\mathbf{\Sigma}^{1/2}\mathbf{U}^T\mathbf{P}}$ is an orthogonal projector and $\|(\mathbf{I} - \mathbf{\Gamma}_{\mathbf{\Sigma}^{1/2}\mathbf{U}^T\mathbf{P}})\| \leq 1$. For the lower bound, we have

$$\begin{aligned}
\|\mathbf{\Sigma}^{-1/2}(\mathbf{I} - \mathbf{\Gamma}_{\mathbf{\Sigma}^{1/2}\mathbf{U}^T\mathbf{P}})\mathbf{\Sigma}^{1/2}\| &= \|\mathbf{\Sigma}^{-1/2}(\mathbf{I} - \mathbf{\Gamma}_{\mathbf{\Sigma}^{1/2}\mathbf{U}^T\mathbf{P}})(\mathbf{I} - \mathbf{\Gamma}_{\mathbf{\Sigma}^{1/2}\mathbf{U}^T\mathbf{P}})\mathbf{\Sigma}^{1/2}\|, \\
&= \|\mathbf{\Sigma}^{-1/2}(\mathbf{I} - \mathbf{\Gamma}_{\mathbf{\Sigma}^{1/2}\mathbf{U}^T\mathbf{P}})\mathbf{\Sigma}^{1/2}\mathbf{\Sigma}^{-1/2}(\mathbf{I} - \mathbf{\Gamma}_{\mathbf{\Sigma}^{1/2}\mathbf{U}^T\mathbf{P}})\mathbf{\Sigma}^{1/2}\|, \\
&\leq \|\mathbf{\Sigma}^{-1/2}(\mathbf{I} - \mathbf{\Gamma}_{\mathbf{\Sigma}^{1/2}\mathbf{U}^T\mathbf{P}})\mathbf{\Sigma}^{1/2}\|\|\mathbf{\Sigma}^{-1/2}(\mathbf{I} - \mathbf{\Gamma}_{\mathbf{\Sigma}^{1/2}\mathbf{U}^T\mathbf{P}})\mathbf{\Sigma}^{1/2}\|, \\
&= \|\mathbf{\Sigma}^{-1/2}(\mathbf{I} - \mathbf{\Gamma}_{\mathbf{\Sigma}^{1/2}\mathbf{U}^T\mathbf{P}})\mathbf{\Sigma}^{1/2}\|^2.
\end{aligned}$$

The assumption $\text{rank}(\mathbf{P}) \neq N$ implies

$$\mathbf{I} \neq \mathbf{\Gamma}_{\mathbf{\Sigma}^{1/2}\mathbf{U}^T\mathbf{P}} \quad \text{and} \quad \|\mathbf{\Sigma}^{-1/2}(\mathbf{I} - \mathbf{\Gamma}_{\mathbf{\Sigma}^{1/2}\mathbf{U}^T\mathbf{P}})\mathbf{\Sigma}^{1/2}\| \neq 0.$$

Therefore, $1 \leq \|\mathbf{\Sigma}^{-1/2}(\mathbf{I} - \mathbf{\Gamma}_{\mathbf{\Sigma}^{1/2}\mathbf{U}^T\mathbf{P}})\mathbf{\Sigma}^{1/2}\|$, as required. □

Lemma 3.11 clarifies the fact that the term $\|\mathbf{I} - \mathbf{P}[\nabla^2_H f_{h,k}]^{-1}\mathbf{R}\nabla^2 f_{h,k}\|$ is at least 1 when $n < N$. This fact reduces the usefulness of the composite convergence rate in Theorem 3.10. In Section 4, we will investigate the term $\|(\mathbf{I} - \mathbf{PR})(\mathbf{x}_{h,k} - \mathbf{x}_{h,\star})\|$ and show that it is sufficiently small in a specific case.

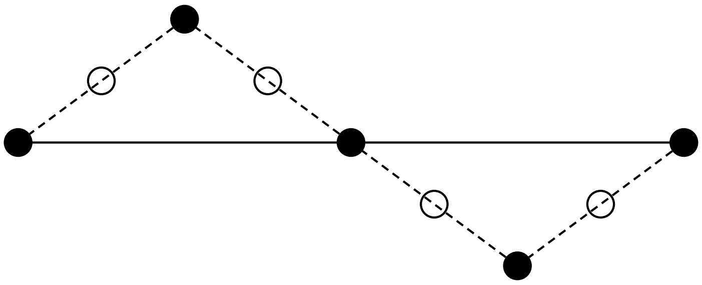

Figure 1.: $\mathbf{P}$ in (19)

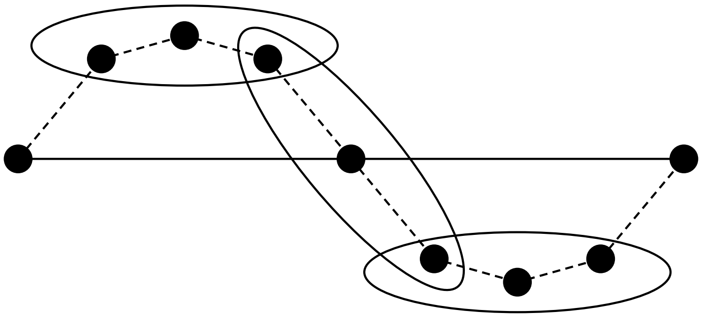

Figure 2.: $\mathbf{R}$ in (20)

4. PDE-based Problems: One-dimensional Case

In this section, we study the Newton-type multilevel model that arises from PDE-based problems. We begin with introducing the basic setting, and then we analyze the coarse correction step in this specific case. Building upon the composite rate in Section 3.4, at the end of this section we re-derive the composite rate with a more insightful bound of $\|(\mathbf{I}-\mathbf{PR})(\mathbf{x}_{h,k}-\mathbf{x}_{h,\star})\|$. As mentioned in Section 3, this quantity is critical in analyzing the performance and complexity of NeMO.

For the simplicity of the analysis, we consider specifically the one-dimensional case, i.e. the decision variable of the infinite dimensional problems is a functional in $\mathbb{R}$. We further assume that the decision variable is discretized uniformly over $[0,1]$ with value 0 on the boundary. We would like to clarify that the approach of analysis in this section could be applied to more general and high-dimensional settings.

4.1. Newton-type multilevel Model by One-dimensional Interpolations

For one dimensional problems, we consider the standard linear prolongation operator and restriction operator. Based on the traditional setting in multigrid research, we define the following Newton-type multilevel model.

- N is an even number,
- the (fine) discretized decision variable is in $\mathbf{R}^{N-1}$, and
- the coarse model is in $\mathbf{R}^{N/2-1}$.

For interpolation operator $\mathbf{P} \in \mathbb{R}^{(N-1)\times(N/2-1)}$, we consider

$$\mathbf{P} = \frac{1}{2}\begin{pmatrix} 1 & & & \\ 2 & & & \\ 1 & 1 & & \\ & 2 & & \\ & 1 & & \\ & & \ddots & 1 \\ & & & 2 \\ & & & 1 \end{pmatrix}, \tag{19}$$

and the restriction operator

$$\mathbf{R} = \frac{1}{2}\mathbf{P}^T \in \mathbb{R}^{(N/2-1)\times(N-1)}. \tag{20}$$

Notice that the $\mathbf{P}$ and $\mathbf{R}$ in (19) and (20) have geometric meanings, and they are one of the standard pairs of operators in multilevel and multigrid methods [3]. As shown in Figure 1, $\mathbf{P}$ is an interpolation operator such that one point is interpolated linearly between every two points. On the other hand, from Figure 2, $\mathbf{R}$ performs restriction by doing weighted average onto every three points. These two operators assume the boundary condition is zero for both end points. We emphasize that the approach of convergence analysis in this section is not restricted for this specific pair of $\mathbf{P}$ and $\mathbf{R}$. We believe the general approach could be applied to interpolation type operators, especially operators that are designed for PDE-based optimization problems.

4.2. Analysis

With the definitions (19) and (20), we investigate the convergence behaviour of the coarse correction step. The analytical tool we used in this section is Taylor's expansion. To deploy this technique, we consider interpolations over the elements of vectors. In particular, we consider interpolations that are twice differentiable with the following definition.

Definition 4.1. For any vector $\mathbf{r} \in \mathbb{R}^{N-1}$, we denote $\mathcal{F}_{\mathbf{r}}^{N-1}$ to be the set of twice differentiable functions such that $\forall w \in \mathcal{F}_{\mathbf{r}}^{N-1}$,

$$w(0) = w(1) = 0, \quad \textit{and} \quad w_i = w(y_i) = (\mathbf{r})_i,$$

where $y_i = i/N$ for $i = 1, 2, \dots, N-1$.

Using the definitions (19) and (20), we can estimate the "information loss" via interpolations using the following proposition.

Proposition 4.2. *Suppose* $\mathbf{P}$ *and* $\mathbf{R}$ *are defined in (19) and (20), respectively. For any vector* $\mathbf{r}_h \in \mathbb{R}^{N-1}$*, we denote* $(\mathbf{r}_h)_0 = (\mathbf{r}_h)_N = 0$ *and obtain*

$$(\mathbf{P}\mathbf{R}\mathbf{r}_h)_j = \begin{cases} \frac{1}{4}((\mathbf{r}_h)_{j-1} + 2(\mathbf{r}_h)_j + (\mathbf{r}_h)_{j+1}) & \textit{if } j \textit{ is even,} \\ \frac{1}{8}((\mathbf{r}_h)_{j-2} + 2(\mathbf{r}_h)_{j-1} + 2(\mathbf{r}_h)_j + 2(\mathbf{r}_h)_{j+1} + (\mathbf{r}_h)_{j+2}) & \textit{if } j \textit{ is odd,} \end{cases}$$

for $j = 1, 2, \dots, N-1$.

Proof. By the definition of $\mathbf{R}$ and $\mathbf{P}$, we have

$$(\mathbf{R}\mathbf{r}_h)_j = \frac{1}{4}((\mathbf{r}_h)_{2j-1} + 2(\mathbf{r}_h)_{2j} + (\mathbf{r}_h)_{2j+1}), \quad 1 \leq j \leq \frac{n}{2} - 1.$$

So

$$(\mathbf{P}\mathbf{R}\mathbf{r}_h)_j = (\mathbf{R}\mathbf{r}_h)_{j/2} = \frac{1}{4}((\mathbf{r}_h)_{j-1} + 2(\mathbf{r}_h)_j + (\mathbf{r}_h)_{j+1}) \quad \text{if } j \text{ is even,}$$

and

$$\begin{aligned}(\mathbf{P}\mathbf{R}\mathbf{r}_h)_j &= \frac{1}{2}\left((\mathbf{R}\mathbf{r}_h)_{(j-1)/2} + (\mathbf{R}\mathbf{r}_h)_{(j+1)/2}\right), \\ &= \frac{1}{8}((\mathbf{r}_h)_{j-2} + 2(\mathbf{r}_h)_{j-1} + 2(\mathbf{r}_h)_j + 2(\mathbf{r}_h)_{j+1} + (\mathbf{r}_h)_{j+2}) \quad \text{if } j \text{ is odd.}\end{aligned}$$

So we obtain the desired result. □

Using the above proposition and Taylor's expansion, we obtain the following lemma.

Lemma 4.3. *Suppose* $\mathbf{P}$ *and* $\mathbf{R}$ *are defined in (19) and (20), respectively. For any vector* $\mathbf{r}_h \in \mathbb{R}^{N-1}$,

$$\|(\mathbf{I} - \mathbf{P}\mathbf{R})\mathbf{r}_h\|_\infty \leq \min_{w \in \mathcal{F}_{\mathbf{r}_h}^{N-1}} \max_{y \in [0,1]} |w''(y)| \frac{3}{4N^2}.$$

Note that the definition of $\mathcal{F}_{\mathbf{r}_h}^{N-1}$ *follows from Definition 4.1.*

Proof. Using Proposition 4.2 and Taylor's Theorem, in the case that j is even, we obtain

$$\begin{aligned}\frac{1}{4}((\mathbf{r}_h)_{j-1} + 2(\mathbf{r}_h)_j + (\mathbf{r}_h)_{j+1}) &= \frac{1}{4}\left(w\left(y_{j-1}\right) + 2w\left(y_j\right) + w\left(y_{j+1}\right)\right), \\ &= w\left(y_j\right) + \frac{w''(y_{c1})}{8}\frac{1}{N^2} + \frac{w''(y_{c2})}{8}\frac{1}{N^2}, \\ &= (\mathbf{r}_h)_j + \frac{w''(y_{c1}) + w''(y_{c2})}{8}\frac{1}{N^2},\end{aligned}$$

where $w(\cdot) \in \mathcal{F}_{\mathbf{r}_h}^{N-1}$, $y_{j-1} \leq y_{c1} \leq y_j$, and $y_j \leq y_{c2} \leq y_{j+1}$. Similarly, in the case that j is odd, we have

$$\begin{aligned}&\frac{1}{8}((\mathbf{r}_h)_{j-2} + 2(\mathbf{r}_h)_{j-1} + 2(\mathbf{r}_h)_j + 2(\mathbf{r}_h)_{j+1} + (\mathbf{r}_h)_{j+2}) \\ &\qquad = (\mathbf{r}_h)_j + \frac{4w''(y_{c3}) + 2w''(y_{c4}) + 2w''(y_{c5}) + 4w''(y_{c6})}{16}\frac{1}{N^2}, \quad (21)\end{aligned}$$

where $y_{j-2} \leq y_{c3} \leq y_j$, $y_{j-1} \leq y_{c4} \leq y_j$, $y_j \leq y_{c5} \leq y_{j+1}$, and $y_j \leq y_{c6} \leq y_{j+2}$. Therefore,

$$\|(\mathbf{I} - \mathbf{P}\mathbf{R})\mathbf{r}_h\|_\infty \leq \max_{y \in [0,1]} |w''(y)| \frac{3}{4N^2} \quad \text{for} \quad \forall w(\cdot) \in \mathcal{F}_{\mathbf{r}_h}^{N-1}.$$

So we obtain the desired result. □

Lemma 4.3 provides upper bound of $\|(\mathbf{I}-\mathbf{PR})\mathbf{r}_h\|_\infty$, for any $\mathbf{r}_h \in \mathbf{R}^{N-1}$. This result can be used to derive the upper bound of $\|(\mathbf{I}-\mathbf{PR})(\mathbf{x}_{h,k}-\mathbf{x}_{h,\star})\|$, where $\mathbf{r}_h = \mathbf{x}_{h,k}-\mathbf{x}_{h,\star}$. As we can see, if $|w''(y)| = \mathcal{O}(1)$, where $w \in \mathcal{F}_{\mathbf{r}_h}^{N-1}$, then $\|(\mathbf{I}-\mathbf{PR})\mathbf{r}_h\|_\infty = \mathcal{O}(N^{-2})$. This can be explained by the fact that when the mesh size is fine enough (i.e. large N), linear interpolation and restriction provide very good estimations of the fine model.

In the following lemma, we provide an upper bound of $|w''|$ in terms of the original vector $\mathbf{r}_h$. The idea is to specify the interpolation method in which we construct w, and we will use cubic spline in particular. Cubic spline is one of the standard interpolation methods, and the output interpolated function w satisfies the setting in Definition 4.1 and Lemma 4.3.

Lemma 4.4. *Suppose* $\mathbf{P}$ *and* $\mathbf{R}$ *are defined in (19) and (20), respectively. For any vector* $\mathbf{r}_h \in \mathbb{R}^{N-1}$, *we obtain*

$$\|(\mathbf{I}-\mathbf{PR})\mathbf{r}_h\|_\infty \leq \frac{9}{4N^2}\|\mathbf{A}\mathbf{r}_h\|_\infty,$$

where

$$\mathbf{A} = N^2 \begin{pmatrix} 2 & -1 & & & \\ -1 & 2 & -1 & & \\ & -1 & \ddots & \ddots & \\ & & \ddots & 2 & -1 \\ & & & -1 & 2 \end{pmatrix}.$$

Proof. We follow the notation in Definition 4.1. For $w \in \mathcal{F}_{\mathbf{r}_h}^{N-1}$ that is constructed via cubic spline, in the interval (y_i, y_{i+1}), we have

$$w(y) = Aw_i + Bw_{i+1} + Cw_i'' + Dw_{i+1}'',$$

where

$$\begin{aligned} A &= \frac{y_{i+1}-y}{y_{i+1}-y_i}, \\ B &= \frac{y-y_i}{y_{i+1}-y_i}, \\ C &= \frac{1}{6}(A^3-A)(y_{i+1}-y_i)^2, \\ D &= \frac{1}{6}(B^3-B)(y_{i+1}-y_i)^2. \end{aligned}$$

It is known from [22] that

$$\frac{\mathrm{d}^2 w}{\mathrm{d}y^2} = Aw_i'' + Bw_{i+1}'', \tag{22}$$

and

$$\frac{y_i-y_{i-1}}{6}w_{i-1}'' + \frac{y_{i+1}-y_{i-1}}{3}w_i'' + \frac{y_{i+1}-y_i}{6}w_{i+1}'' = \frac{w_{i+1}-w_i}{y_{i+1}-y_i} - \frac{w_i-w_{i-1}}{y_i-y_{i-1}}, \tag{23}$$

and for $i = 1, 2, \ldots, N-1$. Using the above equation (22), at the interval (y_i, y_{i+1}), we obtain

$$\begin{aligned}
\left|\frac{\mathrm{d}^2 w}{\mathrm{d}y^2}\right| = \left|Aw''_i + Bw''_{i+1}\right| &= \left|\frac{y_{i+1}-y}{y_{i+1}-y_i}w''_i + \frac{y-y_i}{y_{i+1}-y_i}w''_{i+1}\right|, \\
&\leq \left|\frac{y_{i+1}-y}{y_{i+1}-y_i}\right||w''_i| + \left|\frac{y-y_i}{y_{i+1}-y_i}\right||w''_{i+1}|, \\
&\leq \max\{|w''_i|, |w''_{i+1}|\}.
\end{aligned}$$

Suppose $j \in \arg\max_i\{|w''_i|\}_i$, then from (23) and the fact that $y_{j+1} - y_j = 1/N$,

$$\begin{aligned}
\frac{y_{j+1}-y_{j-1}}{3}w''_j &= \frac{w_{j+1}-w_j}{y_{j+1}-y_j} - \frac{w_j - w_{j-1}}{y_j - y_{j-1}} - \frac{y_j - y_{j-1}}{6}w''_{j-1} - \frac{y_{j+1}-y_j}{6}w''_{j+1}, \\
\frac{2}{3N}w''_j &= N(w_{j+1}-w_j) - N(w_j - w_{j-1}) - \frac{1}{6N}w''_{j-1} - \frac{1}{6N}w''_{j+1}, \\
2w''_j &= 3N^2(w_{j+1} - 2w_j + w_{j-1}) - \frac{1}{2}w''_{j-1} - \frac{1}{2}w''_{j+1}.
\end{aligned}$$

Thus,

$$\begin{aligned}
|2w''_j| &\leq 3N^2|w_{j+1} - 2w_j + w_{j-1}| + \frac{1}{2}|w''_{j-1}| + \frac{1}{2}|w''_{j+1}|, \\
2|w''_j| &\leq 3N^2|w_{j+1} - 2w_j + w_{j-1}| + \frac{1}{2}|w''_j| + \frac{1}{2}|w''_j|, \\
|w''_j| &\leq 3N^2|w_{j+1} - 2w_j + w_{j-1}|.
\end{aligned}$$

Therefore,

$$|w''_i| \leq \max_i 3N^2|w_{i+1} - 2w_i + w_{i-1}|,$$

and so,

$$\|(\mathbf{I} - \mathbf{PR})\mathbf{r}_h\|_\infty \leq \max_{y\in[0,1]}|w''(y)|\frac{3}{4N^2} \leq \max_i \frac{9|w_{i+1} - 2w_i + w_{i-1}|}{4} = \frac{9}{4N^2}\|\mathbf{A}\mathbf{r}_h\|_\infty,$$

as required. □

Lemma 4.4 provides the discrete version of the result presented in Lemma 4.3. The matrix $\mathbf{A}$ is the discretized Laplacian operator, which is equivalent to twice differentiation using finite difference with a uniform mesh.

4.3. Convergence

With all the results, we revisit the composite convergence rate with the following Corollary.

Corollary 4.5. *Suppose* $\mathbf{P}$ *and* $\mathbf{R}$ *are defined in (19) and (20), respectively. If the coarse* *correction step $\hat{\mathbf{d}}_{h,k}$ in (16) is taken with $\alpha_{h,k} = 1$, then*

$$\begin{aligned}
\|\mathbf{x}_{h,k+1} - \mathbf{x}_{h,\star}\| &\leq \sqrt{\frac{L_h}{\mu_h}} \min_{w \in \mathcal{F}^{N-1}_{\mathbf{x}_{h,k} - \mathbf{x}_{h,\star}}} \max_{y \in [0,1]} |w''(y)| \frac{3}{4N^{3/2}} + \frac{M_h \omega^2 \xi^2}{2\mu_h} \|\mathbf{x}_{h,k} - \mathbf{x}_{h,\star}\|^2, \\
&\leq \frac{9}{4N^{3/2}} \sqrt{\frac{L_h}{\mu_h}} \|\mathbf{A}(\mathbf{x}_{h,k} - \mathbf{x}_{h,\star})\| + \frac{M_h \omega^2 \xi^2}{2\mu_h} \|\mathbf{x}_{h,k} - \mathbf{x}_{h,\star}\|^2,
\end{aligned}$$

where $\mathbf{A}$ is defined in Lemma 4.4. Note that M_h, L_h, and μ_h are defined in Assumption 2.1, $\omega = \max\{\|\mathbf{P}\|, \|\mathbf{R}\|\}$, and $\xi = \|\mathbf{P}^+\|$.

Proof.

$$\begin{aligned}
\|\mathbf{x}_{h,k+1} - \mathbf{x}_{h,\star}\| &\leq \|\mathbf{I} - \mathbf{P}[\nabla^2_H f_{h,k}]^{-1} \mathbf{R} \nabla^2 f_{h,k}\| \|(\mathbf{I} - \mathbf{PR})(\mathbf{x}_{h,k} - \mathbf{x}_{h,\star})\| \\
&\quad + \frac{M_h \omega^2 \xi^2}{2\mu_h} \|\mathbf{x}_{h,k} - \mathbf{x}_{h,\star}\|^2, \\
&\leq \sqrt{\frac{L_h}{\mu_h}} \min_{w \in \mathcal{F}^{N-1}_{\mathbf{x}_{h,k} - \mathbf{x}_{h,\star}}} \max_{y \in [0,1]} |w''(y)| \frac{3}{4N^{3/2}} + \frac{M_h \omega^2 \xi^2}{2\mu_h} \|\mathbf{x}_{h,k} - \mathbf{x}_{h,\star}\|^2, \\
&\leq \frac{9}{4N^{3/2}} \sqrt{\frac{L_h}{\mu_h}} \|\mathbf{A}(\mathbf{x}_{h,k} - \mathbf{x}_{h,\star})\| + \frac{M_h \omega^2 \xi^2}{2\mu_h} \|\mathbf{x}_{h,k} - \mathbf{x}_{h,\star}\|^2,
\end{aligned}$$

as required. □

Corollary 4.5 provides the convergence of using Newton-type multilevel model for PDE-based problems that we considered. This result shows the complementary of fine correction step and coarse correction step. Suppose the fine correction step can effectively reduce $\|\mathbf{A}(\mathbf{x}_{h,k} - \mathbf{x}_{h,\star})\|$, then the coarse correction step could yield major reduction based on the result shown in Corollary 4.5.

5. Numerical Experiments

In this section, we verify our convergence results with a numerical example. This example satisfies the assumptions of Section 4, and it is an one-dimensional Poisson's equation, which is a standard example in numerical analysis and multigrid algorithms. In the second part of this section, we will compare NeMO with other algorithms.

5.1. Poisson's Equation

We consider an one-dimensional Poisson's equation

$$-\frac{\mathrm{d}^2}{\mathrm{d}q^2} u = w(q) \quad \text{in } [0,1], \quad u(0) = u(1) = 0,$$

where $w(q)$ is chosen as

$$w(q) = \sin(4\pi q) + 8\sin(32\pi q) + 16\sin(64\pi q).$$

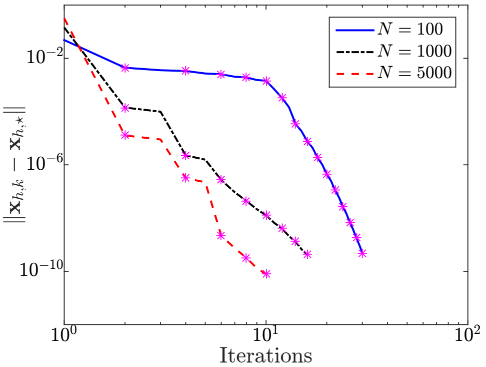

Figure 3.: Convergence of solving Poisson's equation with different N's

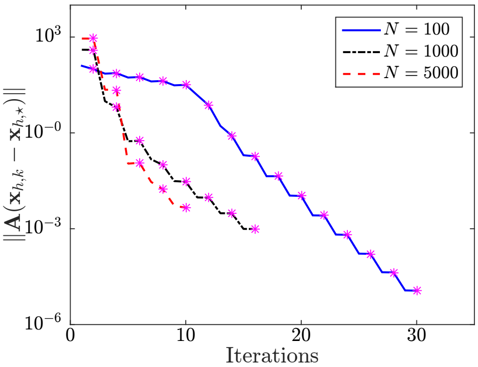

Figure 4.: The smoothing effect with different N's

We discretize the above problem and denote $\mathbf{x}, \mathbf{b} \in \mathbb{R}^{N-1}$, where $(\mathbf{x})_i = u(i/N)$ and $(\mathbf{b})_i = w(i/N)$, for $i = 1, 2, \ldots, N-1$. By using finite difference, we approximate the above equation with

$$\min_{\mathbf{x}\in\mathbb{R}^{N-1}} \frac{1}{2}\mathbf{x}^T\mathbf{A}\mathbf{x} - \mathbf{b}^T\mathbf{x}, \tag{24}$$

where $\mathbf{A}$ is defined as in Lemma 4.4, which is a discretized Laplacian operator.

Figure 3 shows the convergence results of solving (24) with different N's. In this example we use the prolongation and restriction operators that are defined in (19) and (20). Steepest descent is used to compute the fine correction step. The pink stars in Figure 3 and Figure 4 indicate where coarse correction steps were used.

As expected from Corollary 4.5, the performance of convergence is inversely proportional to the discretization level N. More interestingly, one can see the complementary of fine correction step and coarse correction step. From Figure 3, fine correction steps are often deployed after coarse correction steps. Each pair of fine and coarse correction steps provides significant improvement in convergence. Figure 4 shows the smoothing effect of the fine correction step by looking at the quantity $\|\mathbf{A}(\mathbf{x}_{h,k} - \mathbf{x}_{h,\star})\|$, where $\mathbf{A}$ is the discretized Laplacian operator, as defined in Lemma 4.4. As opposed to coarse correction steps, fine correction steps are effective in reducing $\|\mathbf{A}(\mathbf{x}_{h,k} - \mathbf{x}_{h,\star})\|$. Once the error is smoothed, coarse correction steps provide large reduction in error, as shown in Figure 3.

5.2. *Numerical Performance*

Algorithm 1 offers great flexibility with respect to the choice of its various components, such as the interpolation operator, fine-level smoother, linear solver, etc. In our numerical experiments, we have used two variants:

A1.1. The fine-level smoother is the damped Newton method with Armijo line search. Linear systems

$$\mathbf{H}_h\mathbf{d} = -\mathbf{g}_h$$

arising in the Newton method are solved by a direct solver, namely, by the Matlab's backslash operator.

A1.2. The smoother is the Newton method as in A1.1. However, assuming that we have an interpolation and a restriction operators $\mathbf{P}$ and $\mathbf{R}$ at our disposal, we can use it to solve the fine-level linear equation

$$\mathbf{H}_h \mathbf{d} = -\mathbf{g}_h$$

by a two-grid method with $\mathbf{H}_H = \mathbf{R}\mathbf{H}_h\mathbf{P}$.

We will compare the above two methods with the MG/OPT algorithm [19]

A1.3. As in A1.1 but with the coarse level matrix $\mathbf{H}_H$ being the exact Hessian of the coarse level problem.

A1.4. As in A1.2 but with the coarse level matrix $\mathbf{H}_H$ being the exact Hessian of the coarse level problem.

Further details common to all the above variants:

- Linear systems on the coarse level were solved by a direct method (Matlab backslahs operator).
- Initial point: Set as in Matlab as

  ```
  rng(’default’);
  x = 5.*randn(n,1);
  ```

  We did not use the obvious choice $\mathbf{x} = 0$, as this is, for most examples, too close to the region of quadratic convergence of the Newton method. We wanted to see the effect of NeMO when most of the iterations lie outside this region.
- Stopping tolerance: Assuming that we minimize a function f; Algorithm 1 has been stopped when $\|\nabla f(\mathbf{x})\| \leq \varepsilon_{\text{stop}}$, with $\varepsilon_{\text{stop}} = 10^{-9}$ unless specified otherwise.
- The control parameters κ and ε have been chosen as $\kappa = \dfrac{n_H}{n_h}$ and $\varepsilon = 0.1$, unless specified otherwise. (Here n_h and n_H is the number of variables on the fine and the coarse level, respectively.)
- The parameter of the standard Armijo line search is set to 0.01.
- In Algorithms A1.2 and A1.4, the fine level multigrid method was stopped as soon as the scaled residuum of the Newton equation was below 0.1.

In all examples, matrix $\mathbf{A}$ is the discretized two-dimensional Laplace operator. The discretization was performed on a square domain using the finite difference method and we considered homogeneous Dirichlet boundary conditions. When defining the levels, we started with an initial 3×3 grid as “level 1”. Each next level used regular refinement doubling the number of discretization points in each coordinate. Hence “level 2” corresponds to 5×5 and the corresponding matrix $\mathbf{A} \in \mathbb{R}^{9\times 9}$ (after elimination of the boundary points)

$$\mathbf{A} = \frac{1}{3}\begin{pmatrix} 8 & -1 & 0 & -1 & -1 & 0 & 0 & 0 & 0 \\ -1 & 8 & -1 & -1 & -1 & -1 & 0 & 0 & 0 \\ 0 & -1 & 8 & 0 & -1 & -1 & 0 & 0 & 0 \\ -1 & -1 & 0 & 8 & -1 & 0 & -1 & -1 & 0 \\ -1 & -1 & -1 & -1 & 8 & -1 & -1 & -1 & -1 \\ 0 & -1 & -1 & 0 & -1 & 8 & 0 & -1 & -1 \\ 0 & 0 & 0 & -1 & -1 & 0 & 8 & -1 & 0 \\ 0 & 0 & 0 & -1 & -1 & -1 & -1 & 8 & -1 \\ 0 & 0 & 0 & 0 & -1 & -1 & 0 & -1 & 8 \end{pmatrix}.$$

We use up to ten discterization levels with “level 10” corresponding to a problem with 1050625×1050625 matrix $\mathbf{A}$, i.e., a problem with 1050625 variables.

The interpolation operators $\mathbf{P} = P_k^{k+1}$ from level k to level $k+1$ are based on the nine-point interpolation scheme defined by the stencil $\begin{pmatrix} \frac{1}{4} & \frac{1}{2} & \frac{1}{4} \\ \frac{1}{2} & 1 & \frac{1}{2} \\ \frac{1}{4} & \frac{1}{2} & \frac{1}{4} \end{pmatrix}$. We use the full weighting restriction operators defined by $\mathbf{R} = \frac{1}{4}(P_k^{k+1})^T$; see, e.g., [11]. The interpolation operator between levels k and $k+p$ is defined by $\mathbf{P} = P_{k+p-1}^{k+p}\, P_{k+p-2}^{k+p-1} \cdots P_k^{k+1}$ and analogously for the restriction operator $\mathbf{R}$.

5.2.0.1. Example 1. Minimize the following function

$$f(x) := \frac{1}{2}\mathbf{x}^T\mathbf{A}\mathbf{x} + h\lambda\sum_{i=1}^{n}(\mathbf{x}^2 e^{\mathbf{x}} - e^{\mathbf{x}}) - \mathbf{b}^T\mathbf{x}$$

where $\lambda = 10$ and $h = 1/(n+1)$. Here $\mathbf{A}$ is a matrix resulting from discretization of the Laplacian operator on a regular finite element mesh, using bilinear quadrilateral elements and $\mathbf{b}$ is the discretization of function

$$b(x_1, x_2) = \Big(9\pi^2 + e^{(x_1^2 - x_1^3)\sin(3\pi x_2)}(x_1^2 - x_1^3) + 6x_1 - 2\Big)\sin(3\pi x_1)$$

on the same mesh.

Table 1 gives results obtained by NeMO variant A1.1 with a direct solver on all levels. In this (and the next) table the columns show the coarse level used (with 0 being the finest level); number of variables in the coarse level; total number of NeMO iterations; number of NeMO iteration on the fine level (i.e., number of times the fine level Newton equation has been solved); total CPU time on a MacBook Pro with 2.3 GHz Intel Core i5 processor running Matlab 2017b.

The first row of Table 1 shows results with coarse level zero, i.e., for the standard damped Newton metod on the fine level. Hence we compare this line with the remaining NeMO results. Indeed, once we consider coarse level 2 and more, NeMO is substantially faster than the Newton method, in terms of the CPU time. For instance, for coarse level 2, we only have to visit the fine level in 5 iteration, the “rest of the work” is performed on the coarse level. Figure 5 shows the iteration history of NeMO with coarse level 2: most of the initial iteration are performed on the coarse level, while the final iterations are done on the fine level

Table 1.: Example 1, Algorithm A1.1 with direct solver on both levels

coarse level	coarse variables	total iter	fine iter	CPU time
0	1 046 529	20	20	88.1
1	261 121	23	6	42.4
2	65 025	25	5	35.3
3	16 129	30	6	37.9
4	3 969	36	8	47.1
5	961	47	11	60.6

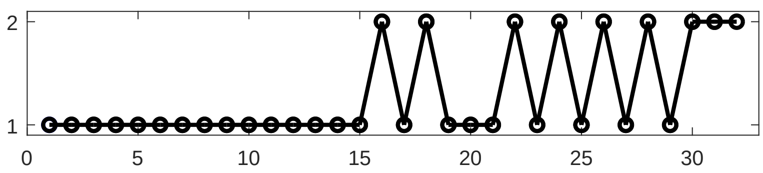

Figure 5.: Example 1, levels visited in every iteration; 1 stands for the coarse level and 2 for the fine level.

Table 1 confirms the advantage of NeMO as compared to the Newton method. However, assuming that we have an interpolation and a restriction operators $\mathbf{P}$ and $\mathbf{R}$ at our disposal, we can use it to solve the fine-level linear equation $\mathbf{H}_h\mathbf{d} = -\mathbf{g}_h$ by a two-grid method with $\mathbf{H}_H = \mathbf{R}\mathbf{H}_h\mathbf{P}$. Table 2 shows the result with this version of NeMO. In addition to the columns presented in Table 1, we also give the total number of two-grid iterations on the fine level (column "mg iter"). As before, we first solve the problem using only the fine level (coarse level 0); the method then becomes equivalent to the standard nonlinear (Newton) multigrid method. The first three rows of Table 2 show results with this method using coarse levels 1, 2 and 3 for the two-grid method. As we can see, this method is much more efficient then the Newton method with a direct solver (first row of Table 1). In the next rows of Table 2 we combine NeMO with the two-grid method for the linear equations on the fine level. As we can see, the advantage of NeMO to the nonlinear multigrid method is not so obvious in this case. NeMO with coarse level 3 is still the fastest method but only just.

Table 2.: Example 1 Algorithm A1.2 with two-grid solver on fine level

coarse level for NeMO	coarse level for mg	coarse level variables	total iter	fine iter	mg iter	CPU time
0	1	1 046 529	20	20	20	37.5
0	2	1 046 529	20	20	22	27.9
0	3	1 046 529	21	21	33	29.8
1	2	261 121	25	8	26	43.6
2	2	65 025	31	10	19	31.8
3	2	16 129	30	9	11	26.2
4	2	3 969	33	10	12	28.3
5	2	961	48	12	14	36.8

Finally, to have a complete overview, we give in Table 3 results for the MG/OPT method [19, 25] when the coarse level matrix for the linear system is computed as an exact Hessian of the objective function discretized on the coarse level. Again, the fine level linear system is either solved by a direct method (first part of Table 3) or by the two-grid method as above. One can see that using the two-grid solver would be slightly beneficial when the number of coarse level variables is small.

Table 3.: Example 1, two-level MG/OPT with direct solver on the fine level (Algorithm A1.3) and with a two-grid solver on the fine level (Algorithm A1.4)

fine level solver	coarse level for MG/OPT	coarse level variables	total iter	fine iter	mg iter	CPU time
direct	1	261 121	14	5	–	30.3
	2	65 025	21	6	–	32.7
	3	16 129	28	7	–	37.8
	4	3 969	33	9	–	47.1
	5	961	28	12	–	51.6
mg	1	261 121	19	6	23	37.1
	2	65 025	28	8	25	35.8
	3	16 129	36	10	12	29.1
	4	3 969	41	10	12	31.0
	5	961	29	12	14	24.1

6. Comments and Perspectives

In this paper we analyzed a Newton-type multilevel optimization (NeMO) algorithm. We argued that the appropriate convergence rate for this multilevel algorithm should be composite i.e. it should have both a linear and quadratic component. We then studied the linear component of the composite rate, and we showed how the hierarchical structure of the model could be used to improve it. To our knowledge, this is the first time a connection between the hierarchal structure of the model and the rate of convergence of a multilevel optimization algorithm has been made. The results presented in this paper can be generalized and refined. The local composite rate of convergence when solving PDE-based optimization can be extended to cases beyond one-dimensional problems or uniform discretization. These extensions would require more careful analysis, but the general approach presented in Section 4 can be applied.

Acknowledgements

The second author was supported by FMJH/PGMO Project No2017-0088 “Multi-level Methods in Constrained Optimization”. The third author was funded by Engineering & Physical Sciences Research Council grant number EP/M028240/1. The third author was also funded in part by JPMorgan Chase & Co. Any views or opinions expressed herein are solely those of the authors listed, and may differ from the views and opinions expressed by JPMorgan Chase & Co. or its affiliates. This material is not a product of the Research Department of J.P. Morgan Securities LLC. This material does not constitute a solicitation or offer in any jurisdiction.

References

[1] A. Beck and L. Tetruashvili. On the convergence of block coordinate descent type methods. *SIAM Journal on Optimization*, 23(4):2037–2060, 2013.

[2] S. Boyd and L. Vandenberghe. *Convex Optimization*. Cambridge University Press, New York, NY, USA, 2004.

[3] W. L. Briggs, V. E. Henson, and S. F. McCormick. *A Multigrid Tutorial*. SIAM, 2000.

[4] H. Calandra, S. Gratton, E. Riccietti, and X. Vasseur. On high-order multilevel optimization strategies, 2019.

[5] H. Calandra, S. Gratton, E. Riccietti, and X. Vasseur. On the approximation of the solution of partial differential equations by artificial neural networks trained by a multilevel levenberg-marquardt method, 2019.

[6] Juan S. Campos and Panos Parpas. A multigrid approach to SDP relaxations of sparse polynomial optimization problems. *SIAM J. Optim.*, 28(1):1–29, 2018.

[7] T. F. Chan and B. F. Smith. Domain decomposition and multigrid algorithms for elliptic problems on unstructured meshes. In *Domain decomposition methods in scientific and engineering computing (University Park, PA, 1993)*, volume 180 of *Contemp. Math.*, pages 175–189. Amer. Math. Soc., Providence, RI, 1994.

[8] M. A. Erdogdu and A. Montanari. Convergence rates of sub-sampled newton methods. In *Advances in Neural Information Processing Systems 28: Annual Conference on Neural Information Processing Systems 2015, December 7-12, 2015, Montreal, Quebec, Canada*, pages 3052–3060, 2015.

[9] S. Gratton, M. Mouffe, A. Sartenaer, P. L. Toint, and D. Tomanos. Numerical experience with a recursive trust-region method for multilevel nonlinear bound-constrained optimization. *Optimization Methods and Software*, 25(3):359–386, 2010.

[10] S. Gratton, A. Sartenaer, and P. L. Toint. Recursive trust-region methods for multiscale nonlinear optimization. *SIAM Journal on Optimization*, 19:414–444, 2008.

[11] W. Hackbusch. *Multigrid methods and applications*, volume 4 of *Springer Series in Computational Mathematics*. Springer-Verlag, Berlin, 1985.

[12] W. Hackbusch. *Multi-Grid Methods and Applications*. Springer, 2003.

[13] J. Han, Y. Yang, and H. Bi. A new multigrid finite element method for the transmission eigenvalue problems. *Applied Mathematics and Computation*, 292:96–106, 2017.

[14] C. P. Ho and P. Parpas. Singularly perturbed Markov decision processes: a multiresolution algorithm. *SIAM Journal on Control and Optimization*, 52(6):3854–3886, 2014.

[15] V. Hovhannisyan, P. Parpas, and S. Zafeiriou. MAGMA: multilevel accelerated gradient mirror descent algorithm for large-scale convex composite minimization. *SIAM J. Imaging Sci.*, 9(4):1829–1857, 2016.

[16] M. Kočvara and S. Mohammed. A first-order multigrid method for bound-constrained convex optimization. *Optimization Methods and Software*, 31(3):622–644, 2016.

[17] R. M. Lewis and S. G. Nash. Model problems for the multigrid optimization of systems governed by differential equations. *SIAM Journal on Scientific Computing*, 26(6):1811–1837 (electronic), 2005.

[18] R. M. Lewis and S. G. Nash. Using inexact gradients in a multilevel optimization algorithm. *Computational Optimization and Applications*, 56(1):39–61, 2013.

[19] S. G. Nash. A multigrid approach to discretized optimization problems. *Optimization Methods and Software*, 14(1-2):99–116, 2000. International Conference on Nonlinear Programming and Variational Inequalities (Hong Kong, 1998).

[20] S. G. Nash. Properties of a class of multilevel optimization algorithms for equality-constrained problems. *Optimization Methods and Software*, 29(1):137–159, 2014.

[21] Panos Parpas. A multilevel proximal gradient algorithm for a class of composite optimization problems. *SIAM J. Sci. Comput.*, 39(5):S681–S701, 2017.

[22] W. H. Press, S. A. Teukolsky, W. T. Vetterling, and B. P. Flannery. *Numerical recipes: the art of scientific computing. Code CD-ROM v 2.06 with Windows, DOS, or Macintosh single-screen license*. Cambridge University Press, Cambridge, 1996.

[23] G. Strang. *Computational science and engineering*. Wellesley-Cambridge Press, Wellesley, MA, 2007.

[24] U. Trottenberg, C. W. Oosterlee, and A. Schüller. *Multigrid*. Academic Press, Inc., San Diego, CA, 2001. With contributions by A. Brandt, P. Oswald and K. Stüben.

[25] Z. Wen and D. Goldfarb. A line search multigrid method for large-scale nonlinear optimization. *SIAM Journal on Optimization*, 20(3):1478–1503, 2009.

[26] P. Wesseling. *An introduction to multigrid methods*. Pure and Applied Mathematics (New York). John Wiley & Sons, Ltd., Chichester, 1992.