On the Use of Hybrid Coarse-Level Models in Multilevel Minimization Methods

Alena Kopaničáková

1 Introduction

We consider the following minimization problem:

$$\min_{\mathbf{x} \in \mathbb{R}^n} f(\mathbf{x}), \tag{1}$$

where $f : \mathbb{R}^n \to \mathbb{R}$ is a bounded, twice continuously differentiable objective function and $n \in \mathbb{N}$ is typically very large. Our goal is to minimize (1) using a nonlinear multilevel minimization (NMM) method, e.g., MG-OPT [11] or RMTR [7]. The main idea behind NMM methods is to employ a hierarchy of so-called coarse-level objective functions, denoted by $\{f^\ell\}_{\ell=1}^L$, where $L > 1$. These functions are typically obtained by exploring the structure of the underlying minimization problem, e.g., by discretizing the underlying infinite-dimensional problem with a varying discretization parameter. During the solution process, the functions $\{f^\ell\}_{\ell=1}^L$ are utilized in order to construct the search-directions for the minimization problem at hand in a computationally efficient manner.

The overall efficiency of NMM methods relies on the ability of the coarse-level objective functions $\{f^\ell\}_{\ell=1}^L$ to approximate the function f well. Indeed, the convergence theory of the majority of NMM methods requires that the local behavior of the coarse-level objective functions is at least first-order coherent with the local behavior of f. The coherence is commonly ensured by employing the so-called τ-correction [1], which corrects the coarse-level objective function f^ℓ in an additive manner. Although this approach is almost universally employed in the multilevel literature, other approaches were also considered, e.g., a second-order additive correction approach [7, 12], or Galerkin-based coarse-level models [7, 9]. In this work, we explore techniques from the surrogate-based/multi-fidelity optimization [4] in order to construct the first-order coherent coarse-level models in the context of NMM methods. In particular, we discuss how to correct functions $\{f^\ell\}_{\ell=1}^L$ using additive, multiplicative, and hybrid approaches.

Alena Kopaničáková
Brown University, USA, e-mail: alena.kopanicakova@brown.edu

2 Nonlinear multilevel minimization framework

In this work, we minimize (1) using the NMM method. To this aim, we consider a hierarchy of L levels. Each level $\ell = 1, \dots, L$ is associated with some model $h^\ell \colon \mathbb{R}^{n^\ell} \to \mathbb{R}$, where we assume that $h^{\ell-1}$ is computationally cheaper to minimize than h^ℓ and that $n^{\ell-1} < n^\ell$. As we will discuss in Section 3, the models $\{h^\ell\}_{\ell=1}^L$ are constructed during the minimization process by correcting the objective functions $\{f^\ell\}_{\ell=1}^L$ by taking into account the knowledge of the current iterate. Through this work, we assume that $h^L := f^L := f$. Transfer of the data between different levels of the multilevel hierarchy is performed using the prolongation operator $\mathbf{I}_\ell^{\ell+1} \colon \mathbb{R}^{n^\ell} \to \mathbb{R}^{n^{\ell+1}}$, and the restriction operator $\mathbf{R}_{\ell+1}^\ell \colon \mathbb{R}^{n^{\ell+1}} \to \mathbb{R}^{n^\ell}$, where $\mathbf{R}_{\ell+1}^\ell = (\mathbf{I}_\ell^{\ell+1})^T$. Moreover, we also employ the projection operator $\mathbf{P}_{\ell+1}^\ell \colon \mathbb{R}^{n^{\ell+1}} \to \mathbb{R}^{n^\ell}$ to transfer iterates from the level $\ell+1$ to ℓ. The operator $\mathbf{P}_{\ell+1}^\ell$ is constructed such that $\mathbf{x}^\ell = \mathbf{P}_{\ell+1}^\ell(\mathbf{I}_\ell^{\ell+1}\mathbf{x}^\ell)$, for any $\mathbf{x}^\ell \in \mathbb{R}^{n^\ell}$.

Using the aforementioned definitions, we now describe a generic NMM method in the form of a V-cycle, summarized in Algorithm 1. During the description, we use a superscript to denote the level and a subscript to denote the iteration index. Starting from the finest level, $\ell = L$, and initial guess $\mathbf{x}_0^\ell$, the NMM method performs μ_s nonlinear smoothing steps to approximately minimize model h^ℓ. The choice of the nonlinear smoother depends on the particular choice of the NMM method. For instance, one can employ a first-order method equipped with a line-search or trust-region globalization strategy if a variant of multilevel line-search or trust-region method is considered. The outcome of this minimization process, iterate $\mathbf{x}_{\mu_s}^\ell$, is then used to construct a coarse-level model $h^{\ell-1}$ and initial guess $\mathbf{x}_0^{\ell-1} = \mathbf{P}_\ell^{\ell-1}\mathbf{x}_{\mu_s}^\ell$. This process is repeated recursively until the coarsest level is reached.

On the coarsest level, $\ell = 1$, an NMM method approximately minimizes h^ℓ using μ_c steps of a nonlinear solution strategy, giving rise to $\mathbf{x}_*^\ell$. Afterwards, the prolongated coarse-level correction $\mathbf{s}_{\mu_s+1}^{\ell+1} := \mathbf{I}_\ell^{\ell+1}(\mathbf{x}_*^\ell - \mathbf{x}_0^\ell)$ is used to update the current iterate $\mathbf{x}_{\mu_s}^{\ell+1}$ on level $\ell+1$. However, before this update is performed, the correction $\mathbf{s}_{\mu_s+1}^\ell$ has to undergo some convergence control. The type of convergence control again depends on the particular type of the NMM method. For example, if the multilevel trust-region method is used, then $\mathbf{s}_{\mu_s+1}^{\ell+1}$ is required to provide a decrease in $h^{\ell+1}$ to be accepted by the algorithm. If a variant of a line-search method is used, then an appropriate step size has to be determined. In the end, the algorithm performs μ_s post-smoothing steps, starting from $\mathbf{x}_{\mu_s+1}^{\ell+1}$ and giving rise to $\mathbf{x}_*^{\ell+1}$. This process is again repeated on all levels until the finest level is reached.

3 Construction of coarse-level models

On each level ℓ, the NMM methods minimize the model $h^\ell \colon \mathbb{R}^{n^\ell} \to \mathbb{R}$ approximately. The result of this minimization, the iterate $\mathbf{x}_*^\ell$, is then used to construct the search

Algorithm 1 NMM($\ell,\ h^\ell,\ \mathbf{x}_0^\ell$)

```
Require: $\ell \in \mathbb{N}, h^\ell : \mathbb{R}^{n^\ell} \to \mathbb{R},\ \mathbf{x}_0^\ell \in \mathbb{R}^{n^\ell}$ and $\mu_s, \mu_c \in \mathbb{N}$
$\mathbf{x}_{\mu_s}^\ell$ = Nonlinear_smoothing($h^\ell,\ \mathbf{x}_0^\ell,\ \mu_s$)
Construct $h^{\ell-1}$ using $\mathbf{x}_{\mu_s}^\ell$, and $\nabla h^\ell(\mathbf{x}_{\mu_s}^\ell)$
if $\ell = 2$ then
    $\mathbf{x}_*^{\ell-1}$ = Nonlinear_solve($h^{\ell-1},\ \mathbf{P}_\ell^{\ell-1}\mathbf{x}_{\mu_s}^\ell,\ \mu_c$)
else
    $\mathbf{x}_*^{\ell-1}$ = NMM($\ell - 1, h^{\ell-1},\ \mathbf{P}_\ell^{\ell-1}\mathbf{x}_{\mu_s}^\ell$)
end if
$\mathbf{x}_{\mu_s+1}^\ell$ = Convergence_control($h^\ell, \mathbf{x}_{\mu_s}^\ell, \boldsymbol{I}_{\ell-1}^\ell(\mathbf{x}_*^{\ell-1} - \mathbf{P}_\ell^{\ell-1}\mathbf{x}_{\mu_s}^\ell)$)
$\mathbf{x}_*^\ell$ = Nonlinear_smoothing($h^\ell,\ \mathbf{x}_{\mu_s+1}^\ell,\ \mu_s$)
return $\mathbf{x}_*^\ell$
```

direction for the minimization on the next finer level. As a consequence, the overall efficiency of NMM methods depends on the capabilities of the models $\{h^\ell\}_{\ell=1}^L$ to approximate f as accurately as possible.

Given an initial guess $\mathbf{x}_0^\ell = \mathbf{P}_{\ell+1}^\ell \mathbf{x}_{\mu_s}^{\ell+1}$, the model h^ℓ is constructed during each V-cycle by correcting the functionf^ℓ, such that the following condition holds:

$$\nabla h^\ell(\mathbf{x}_0^\ell) = \mathbf{R}_{\ell+1}^\ell \nabla h^{\ell+1}(\mathbf{x}_{\mu_s}^{\ell+1}). \tag{2}$$

This ensures that h^ℓ and $h^{\ell+1}$ are locally first-order coherent and that the following relation holds: $\langle \nabla h^\ell(\mathbf{x}_0^\ell), \mathbf{s}^\ell \rangle = \langle \nabla h^{\ell+1}(\mathbf{x}_{\mu_s}^{\ell+1}), \mathbf{I}_\ell^{\ell+1}\mathbf{s}^\ell \rangle$. In this work, we discuss three different approaches for constructing models $\{h^\ell\}_{\ell=1}^L$, namely additive, multiplicative and hybrid. Our discussion considers only the first-order coherent models, constructed using the Taylor approximation of the associated correction function. However, models enforcing higher-order coherency as well as different approximations of the correction function could also be considered.

3.1 An additive approach

Using the additive approach, the coarse-level model $h_{\text{add}}^\ell : \mathbb{R}^{n^\ell} \to \mathbb{R}$ is obtained by correcting the low-cost function f^ℓ as follows

$$h_{\text{add}}^\ell(\mathbf{x}^\ell) = f^\ell(\mathbf{x}^\ell) + \gamma_{\text{add}}^\ell(\mathbf{x}^\ell), \tag{3}$$

where the additive correction function $\gamma_{\text{add}}^\ell : \mathbb{R}^{n^\ell} \to \mathbb{R}$ accounts for the difference between the value of f^ℓ and the fine-level model $h^{\ell+1}$, i.e.,

$$\gamma_{\text{add}}^\ell(\mathbf{x}^\ell) := h^{\ell+1}(\mathbf{I}_\ell^{\ell+1}\mathbf{x}^\ell) - f^\ell(\mathbf{x}^\ell). \tag{4}$$

Unfortunately, the evaluation of $\gamma^{\ell}_{\text{add}}$ at any given $\mathbf{x}^{\ell}$ requires an evaluation of the fine-level model $h^{\ell+1}$ at $\mathbf{I}^{\ell+1}_{\ell}\mathbf{x}^{\ell}$. As a consequence, numerical computations involving h^{ℓ}_{add} are computationally more demanding than computations performed using $h^{\ell+1}$ directly. To ease the computational burden, we evaluate $\gamma^{\ell}_{\text{add}}$ exactly only at the initial coarse-level iterate $\mathbf{x}^{\ell}_0 = \mathbf{P}^{\ell+1}_{\ell}\mathbf{x}^{\ell+1}_{\mu_s}$. Thus, we impose

$$\gamma^{\ell}_{\text{add}}(\mathbf{x}^{\ell}_0) := h^{\ell+1}(\mathbf{x}^{\ell+1}_{\mu_s}) - f^{\ell}(\mathbf{x}^{\ell}_0),$$

only at $\mathbf{x}^{\ell}_0$. For any other $\mathbf{x}^{\ell}$, we approximate the correction function $\gamma^{\ell}_{\text{add}}$ by means of the first-order Taylor approximation, defined around $\mathbf{x}^{\ell}_0$ as follows

$$\tilde{\gamma}^{\ell}_{\text{add}}(\mathbf{x}^{\ell}) = \gamma^{\ell}_{\text{add}}(\mathbf{x}^{\ell}_0) + \langle \nabla\gamma^{\ell}_{\text{add}}(\mathbf{x}^{\ell}_0),\ \mathbf{x}^{\ell} - \mathbf{x}^{\ell}_0 \rangle.$$

Replacing $\gamma^{\ell}_{\text{add}}$ with $\tilde{\gamma}^{\ell}_{\text{add}}$ in (3) gives rise to

$$h^{\ell}_{\text{add}}(\mathbf{x}^{\ell}) := f^{\ell}(\mathbf{x}^{\ell}) + h^{\ell+1}(\mathbf{x}^{\ell+1}_{\mu_s}) - f^{\ell}(\mathbf{x}^{\ell}_0) + \langle \nabla\gamma^{\ell}_{\text{add}}(\mathbf{x}^{\ell}_0), \mathbf{x}^{\ell} - \mathbf{x}^{\ell}_0 \rangle, \tag{5}$$

where

$$\nabla\gamma^{\ell}_{\text{add}}(\mathbf{x}^{\ell}_0) := \mathbf{R}^{\ell}_{\ell+1}\nabla h^{\ell+1}(\mathbf{x}^{\ell+1}_{\mu_s}) - \nabla f^{\ell}(\mathbf{x}^{\ell}_0). \tag{6}$$

Note, the quantity $h^{\ell+1}(\mathbf{x}^{\ell+1}_{\mu_s}) - f^{\ell}(\mathbf{x}^{\ell}_0)$ enforces zeroth-order coherence between $h^{\ell+1}$ and h^{ℓ}_{add} at $\mathbf{x}^{\ell+1}_{\mu_s}$ and $\mathbf{x}^{\ell}_0$, respectively, i.e., $h^{\ell}_{\text{add}}(\mathbf{x}^{\ell}_0) = h^{\ell+1}(\mathbf{x}^{\ell+1}_{\mu_s})$. However, this term does not affect the evaluation of the derivatives of h^{ℓ}_{add}, and therefore it is often neglected in practice. We also point out that the term $\nabla\gamma^{\ell}_{\text{add}}(\mathbf{x}^{\ell}_0)$, known in the multilevel literature as τ-correction, ensures that condition (2) holds.

3.2 A multiplicative approach

Optimization methods that exploit multiple fidelities often employ multiplicative correction functions [4]. In this case, the low-cost approximation f^{ℓ} associated with level ℓ is made coherent with the model $h^{\ell+1}$ as follows:

$$h^{\ell}_{\text{mult}}(\mathbf{x}^{\ell}) = \gamma^{\ell}_{\text{mult}}(\mathbf{x}^{\ell}) f^{\ell}(\mathbf{x}^{\ell}). \tag{7}$$

Here, the multiplicative correction function $\gamma^{\ell}_{\text{mult}} \colon \mathbb{R}^{n^{\ell}} \to \mathbb{R}$ is given as

$$\gamma^{\ell}_{\text{mult}}(\mathbf{x}^{\ell}) := \frac{h^{\ell+1}(\mathbf{I}^{\ell+1}_{\ell}\mathbf{x}^{\ell}) + \kappa}{f^{\ell}(\mathbf{x}^{\ell}) + \kappa}, \tag{8}$$

where $\kappa \approx \epsilon$ ensures numerical stability as the value of $f^{\ell}(\mathbf{x}^{\ell})$ approaches zero.

Similar to the additive approach, evaluating $\gamma^{\ell}_{\text{mult}}$ precisely at all coarse-level iterates is computationally expensive. Therefore, we impose (8) only at $\mathbf{x}^{\ell}_0$, i.e.,

$$\gamma^{\ell}_{\text{mult}}(\mathbf{x}^{\ell}_0) := \frac{h^{\ell+1}(\mathbf{x}^{\ell+1}_{\mu_s}) + \kappa}{f^{\ell}(\mathbf{x}^{\ell}_0) + \kappa},$$

where we explored that $\mathbf{x}^{\ell+1}_{\mu_s} = \mathbf{I}^{\ell+1}_{\ell}\mathbf{x}^{\ell}_0$. At any other iterate $\mathbf{x}^{\ell}$, we approximate $\gamma^{\ell}_{\text{mult}}$ by means of the first-order Taylor approximation, defined around $\mathbf{x}^{\ell}_0$ as

$$\tilde{\gamma}^{\ell}_{\text{mult}}(\mathbf{x}^{\ell}) = \gamma^{\ell}_{\text{mult}}(\mathbf{x}^{\ell}_0) + \langle \nabla\gamma^{\ell}_{\text{mult}}(\mathbf{x}^{\ell}_0),\ \mathbf{x}^{\ell} - \mathbf{x}^{\ell}_0 \rangle. \tag{9}$$

Replacing $\gamma^{\ell}_{\text{mult}}$ with $\tilde{\gamma}^{\ell}_{\text{mult}}$ in (7) then gives rise to the first-order coherent model

$$h^{\ell}_{\text{mult}}(\mathbf{x}^{\ell}) := \tilde{\gamma}^{\ell}_{\text{mult}}(\mathbf{x}^{\ell})\ f^{\ell}(\mathbf{x}^{\ell}). \tag{10}$$

The numerical evaluation of $\tilde{\gamma}^{\ell}_{\text{mult}}$ amounts to

$$\tilde{\gamma}^{\ell}_{\text{mult}}(\mathbf{x}^{\ell}) := \frac{h^{\ell+1}(\mathbf{x}^{\ell+1}_{\mu_s}) + \kappa}{f^{\ell}(\mathbf{x}^{\ell}_0) + \kappa} + \langle \nabla\gamma^{\ell}_{\text{mult}}(\mathbf{x}^{\ell}_0),\ \mathbf{x}^{\ell} - \mathbf{x}^{\ell}_0 \rangle,$$

where $\nabla\gamma^{\ell}_{\text{mult}}(\mathbf{x}^{\ell}_0)$ is given by

$$\nabla\gamma^{\ell}_{\text{mult}}(\mathbf{x}^{\ell}_0) := \frac{1}{f^{\ell}(\mathbf{x}^{\ell}_0) + \kappa}\left(\mathbf{R}^{\ell}_{\ell+1}\nabla h^{\ell+1}(\mathbf{x}^{\ell+1}_{\mu_s})\right) - \frac{h^{\ell+1}(\mathbf{x}^{\ell+1}_{\mu_s}) + \kappa}{(f^{\ell}(\mathbf{x}^{\ell}_0) + \kappa)^2}\nabla f^{\ell}(\mathbf{x}^{\ell}_0).$$

Straightforward calculations show that model h^{ℓ}_{mult}, defined by (10), is zeroth-order and first-order coherent with $h^{\ell+1}$ at $\mathbf{x}^{\ell}_0$ and $\mathbf{x}^{\ell+1}_{\mu_s}$, respectively.

3.3 A hybrid approach

From a computational point of view, additive and multiplicative approaches are comparable. However, their behavior is very different. The additive approach adds new terms to f^{ℓ}, which can be interpreted as uniform translation (zeroth-order), and rotation (first-order) of the function graph; see also Fig. 1. In contrast, the multiplicative approach introduces skewing, which might not be desirable if f and f^{ℓ} are in good agreement, at least locally. However, if functions f^{ℓ} and f are not in good agreement, then additional skewing can be beneficial [3], e.g., if the polynomial order of f is higher than the polynomial order of f^{ℓ}. Moreover, multiplication of f^{ℓ} with $\tilde{\gamma}^{\ell}_{\text{mult}}$ can introduce new minima on level ℓ, where $\ell < L$. For instance, let us suppose that f^{ℓ} is a second-order polynomial. Its multiplication with $\tilde{\gamma}^{\ell}_{\text{mult}}$ increases the order of the polynomial, i.e., we obtain a model h^{ℓ}_{mult} which is quartic and has, in general, more minima than quadratic function.

In general, it is not known a priori whether the additive or the multiplicative model is more suitable for a given optimization problem. To overcome this difficulty, a hybrid approach [6] can be employed. A coarse-level model h^{ℓ}_{mix} is then obtained

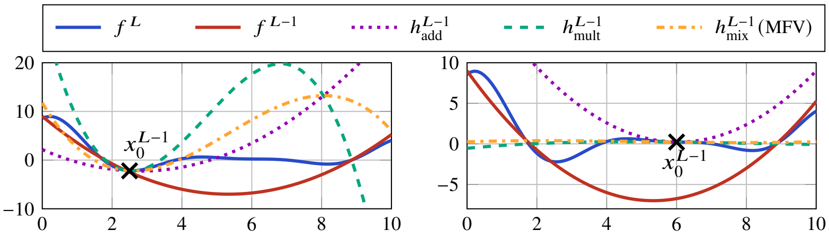


Fig. 1 Coarse-level models constructed around $x_0^{L-1} = 2.5$ and $x_0^{L-1} = 6.0$.

as a convex combination of the additive h^ℓ_{add} and the multiplicative h^ℓ_{mult} models, i.e.,

$$h^\ell_{\text{mix}}(\mathbf{x}^\ell) := w^\ell_{\text{add}}\, h^\ell_{\text{add}}(\mathbf{x}^\ell) + w^\ell_{\text{mult}}\, h^\ell_{\text{mult}}(\mathbf{x}^\ell), \tag{11}$$

where $w^\ell_{\text{add/mult}} \in \mathbb{R}$ and $w^\ell_{\text{add}} + w^\ell_{\text{mult}} = 1$. In order to maximize the approximation properties of h^ℓ_{mix}, the weights $w^\ell_{\text{add}}, w^\ell_{\text{mult}}$ have to be chosen carefully. Below, we describe two different strategies for selecting the values w^ℓ_{add} and w^ℓ_{mult}.

3.3.1 Matching function values (MFV) at the previously evaluated fine-level iterate

Following [2], the weights $w^\ell_{\text{add}}, w^\ell_{\text{mult}}$ can be selected by matching the function value at the previously evaluated fine-level iterate, denoted by $\mathbf{x}^{\ell+1}_p$, as in

$$w^\ell_{\text{add}} = \frac{h^{\ell+1}(\mathbf{x}^{\ell+1}_p) - h^\ell_{\text{mult}}(\mathbf{x}^\ell_0)}{h^\ell_{\text{add}}(\mathbf{x}^\ell_0) - h^\ell_{\text{mult}}(\mathbf{x}^\ell_0)} \qquad \text{and} \qquad w^\ell_{\text{mult}} = 1 - w^\ell_{\text{add}}. \tag{12}$$

From a computational point of view, evaluating (12) is cheap as $h^{\ell+1}(\mathbf{x}^{\ell+1}_p)$ is readily available, for instance from the $\mu_s - 1$ pre-smoothing step performed on level $\ell + 1$.

3.3.2 Bayesian updating approach

To maximize the approximation properties of h^ℓ_{mix}, it might be beneficial to take into account the history of the d^ℓ previously evaluated fine-level iterates [3]. Therefore, we consider the dataset $\mathcal{D}^\ell = \{(h^{\ell+1}(\mathbf{x}^{\ell+1}_p), h^\ell_{\text{add}}(\mathbf{P}^\ell_{\ell+1}\mathbf{x}^{\ell+1}_p), h^\ell_{\text{mult}}(\mathbf{P}^\ell_{\ell+1}\mathbf{x}^{\ell+1}_p))\}_{p=1}^{d^\ell}$, where each sample contains the function value of $h^{\ell+1}$ at $\mathbf{x}^{\ell+1}_p$, as well as the function values of the coarse-level models $h^\ell_{\text{add/mult}}$ obtained at $\mathbf{P}^\ell_{\ell+1}\mathbf{x}^{\ell+1}_p$. In this work, we construct $\mathcal{D}^\ell$ by taking into account the last d^ℓ iterates which were transferred from level $\ell + 1$ to level ℓ. For example, if $d^\ell = 3$, then $\mathcal{D}^\ell$ is constructed by taking into account the iterate $\mathbf{x}^{\ell+1}_p = \mathbf{x}^{\ell+1}_{\mu_s}$, obtained as a result of the pre-smoothing step during the previous three V-cycles. For simplicity, we use the notation $d^\ell = \infty$ to denote all previous V-cycles.

Having constructed the dataset $\mathcal{D}^\ell$, we can now employ the Bayesian posterior updating approach [3] to determine the values of $w^\ell_{\text{add/mult}}$. Starting from $w^\ell_{\text{add/mult}} = 0.5$, the weights are updated every time the model h^ℓ is constructed as follows:

$$w^\ell_{\text{add/mult}} = \frac{w^\ell_{\text{add/mult}}\psi^\ell_{\text{add/mult}}}{w^\ell_{\text{mult/add}}\psi^\ell_{\text{mult/add}} + w^\ell_{\text{add/mult}}\psi^\ell_{\text{add/mult}}}. \tag{13}$$

The model likelihoods $\psi^\ell_{\text{add/mult}}$ in (13) are evaluated as

$$\psi^\ell_{\text{add/mult}} = \left(2\pi\sigma^2_{\text{add/mult}}\right)^{-d^\ell/2} \exp(-d^\ell/2), \tag{14}$$

and the maximum likelihood estimator of the model variance is given by

$$\sigma^2_{\text{add/mult}} = \frac{1}{d^\ell}\sum_{p=1}^{d^\ell}(h^{\ell+1}(\mathbf{x}^{\ell+1}_p) - h^\ell_{\text{add/mult}}(\mathbf{P}^\ell_{\ell+1}\mathbf{x}^{\ell+1}_p)). \tag{15}$$

4 Numerical results and discussion

In this section, we investigate the influence of different coarse-level models on the performance of the NMM method using numerical examples from the field of supervised learning, namely classification using ResNets [8]. Given a dataset $\mathcal{S} = \{(\mathbf{z}_s, \mathbf{c}_s)\}_{s=1}^{n_s}$, where $\mathbf{z}_s \in \mathbb{R}^{n_{in}}$ and $\mathbf{c}_s \in \mathbb{R}^{n_{out}}$, our goal is to find parameters $\mathbf{x} \in \mathbb{R}^n$ of a ResNet, defined as RN: $\mathbb{R}^{n_{in}} \times \mathbb{R}^n \to \mathbb{R}^{n_{out}}$, by solving the following minimization problem:

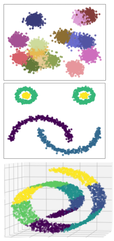
$$\min_{\mathbf{x}\in\mathbb{R}^n} f(\mathbf{x}) := \frac{1}{n_s}\sum_{s=1}^{n_s} g(\text{RN}(\mathbf{z}_s, \mathbf{x}), \mathbf{c}_s), \tag{16}$$

where g denotes the cross-entropy loss function.

Since (16) is a non-convex function, we choose the NMM method to be a variant of the RMTR method [7]. The multilevel hierarchy and transfer operators are constructed by leveraging the fact that the ResNet can be interpreted as a forward Euler discretization of an ordinary differential equation; see [10, 5] for details. Here, we construct a hierarchy of ResNets by uniformly refining a ResNet with three layers three times. Fig. 4 demonstrates the number of effective gradient evaluations[1] of the RMTR method with respect to different coarse-level models for three different datasets.

As we can observe, the choice of the coarse-level model has a significant impact on the overall efficiency of the multilevel method. For all three examples, hybrid approaches outperform purely additive and multiplicative ones. In terms of hybrid

[1] The number of effective gradient evaluations is obtained as $\sum_{\ell=1}^{L} 2^{\ell-L} W_\ell C_L$, where C_L represents a cost associated with an evaluation of the gradient on the level L, W_ℓ describes a number of gradient evaluations performed on a level ℓ, and $2^{\ell-L}$ is a coarsening factor in 1D.



Model/Example	Blobs	Smiley	Spiral
h_{add}	29 ± 5.3%	676 ± 11.2%	203 ± 12.3%
h_{mult}	32 ± 6.1%	485 ± 15.1%	153 ± 15.9%
$h_{\text{mix}}(w = 0.5)$	38 ± 4.8%	404 ± 10.3%	297 ± 11.3%
$h_{\text{mix}}(\text{MFV})$	25 ± 4.2%	352 ± 6.5%	**123 ± 7.1**%
$h_{\text{mix}}(d^{\ell} = 5)$	25 ± 3.4%	514 ± 6.3%	197 ± 6.8%
$h_{\text{mix}}(d^{\ell} = 20)$	**24 ± 2.9**%	471 ± 7.7%	156 ± 7.4%
$h_{\text{mix}}(d^{\ell} = \infty)$	25 ± 3.8%	**301 ± 6.9**%	126 ± 9.9%

Fig. 2 *Left:* Blobs, Smiley, and Spiral datasets (*Top* to *Down*). Each class is illustrated by different color. *Right:* The average number of effective gradient evaluations of the RMTR method (4 levels). Averages are obtained from 5 independent runs.

models, we observe that the Bayesian approach performs similar, or superior to MFV, especially if all prior fine-level iterates are considered ($d^{\ell} = \infty$).

Given our (limited) numerical experience, we believe that employing hybrid, and possibly other types of novel coarse-level models, provides a promising future direction for improving the efficiency and the reliability of NMM methods.

Acknowledgements This work was supported by the Swiss National Science Foundation under the project "Multilevel training of DeepONets — multiscale and multiphysics applications" (grant no. 206745).

References

1. Brandt, A. Multi-level adaptive solutions to boundary-value problems. *Mathematics of computation* **31**(138), 333–390 (1977).
2. Eldred, M. and Dunlavy, D. Formulations for surrogate-based optimization with data fit, multifidelity, and reduced-order models. In: *11th AIAA/ISSMO multidisciplinary analysis and optimization conference*, 7117 (2006).
3. Fischer, C. C., Grandhi, R. V., and Beran, P. S. Bayesian-Enhanced Low-Fidelity Correction Approach to Multifidelity Aerospace Design. *AIAA Journal* **56**(8), 3295–3306 (2018).
4. Forrester, A. I. and Keane, A. J. Recent advances in surrogate-based optimization. *Progress in aerospace sciences* **45**(1-3), 50–79 (2009).
5. Gaedke-Merzhäuser, L., Kopaničáková, A., and Krause, R. Multilevel minimization for deep residual networks. *ESAIM. Proceedings and Surveys* **71**, 131 (2021).
6. Gano, S. E., Renaud, J. E., and Sanders, B. Hybrid variable fidelity optimization by using a kriging-based scaling function. *Aiaa Journal* **43**(11), 2422–2433 (2005).
7. Gratton, S., Sartenaer, A., and Toint, P. L. Recursive Trust-Region Methods for Multiscale Nonlinear Optimization. *SIAM Journal on Optimization* **19**(1), 414–444 (2008).
8. He, K., Zhang, X., Ren, S., and Sun, J. Deep residual learning for image recognition. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*, 770–778 (2016).
9. Ho, C. P., Kočvara, M., and Parpas, P. Newton-type multilevel optimization method. *Optimization Methods and Software* 1–34 (2019).
10. Kopaničáková, A. and Krause, R. Globally Convergent Multilevel Training of Deep Residual Networks. *SIAM Journal on Scientific Computing* (0), S254–S280 (2022).
11. Nash, S. G. A multigrid approach to discretized optimization problems. *Optimization Methods and Software* **14**(1-2), 99–116 (2000).
12. Yavneh, I. and Dardyk, G. A Multilevel Nonlinear Method. *SIAM Journal on Scientific Computing* **28**(1), 24–46 (2006).