A Multilevel Proximal Algorithm for Large Scale Composite Convex Optimization

Panos Parpas · Duy V. N. Luong · Daniel Rueckert · Berc Rustem

March 23, 2014

Abstract Composite convex optimization models consist of the minimization of the sum of a smooth convex function and a non-smooth convex function. Such models arise in many applications where, in addition to the composite nature of the objective function, a hierarchy of models is readily available. It is common to take advantage of this hierarchy of models by first solving a low fidelity model and then using the solution as a starting point to a high fidelity model. We adopt an optimization point of view and show how to take advantage of the availability of a hierarchy of models in a consistent manner. We do not use the low fidelity model just for the computation of promising starting points but also for the computation of search directions. We establish the convergence and convergence rate of the proposed algorithm and compare our algorithm with two widely used algorithms for this class of models (ISTA and FISTA). Our numerical experiments on large scale image restoration problems suggest that, for certain classes of problems, the proposed algorithm is significantly faster than both ISTA and FISTA.

Keywords Composite convex optimization · Multigrid · Iterative Shrinkage Thresholding Algorithm

1 Introduction

It is often possible to exploit the structure of large scale optimization models in order to develop algorithms with lower computational complexity. A noteworthy example are composite convex optimization models that consist of the minimization of the sum of a smooth convex function and a non-smooth (but simple) convex function. For a general non–smooth convex function the subgradient algorithm converges at a rate of $O(1/\sqrt{k})$ for function values, where k is the iteration number. However, if one assumes that the non–smooth component is simple enough such

Department of Computing,
Imperial College London,
180 Queens Gate,
SW7 2AZ,
E-mail: p.parpas@imperial.ac.uk

that the proximal projection step can be performed in closed form, then the convergence rate can be improved to $O(1/k^2)$ [2,24]. Composite convex optimization models arise often and in a wide range of applications from computer science (e.g. machine learning), statistics (e.g the lasso problem), and engineering (e.g. signal processing), to name just a few.

In addition to the composition of the objective function, many of the applications described above share another common structure. The fidelity in which the optimization model captures the underlying application can often be controlled. Typical examples include the discretization of Partial Differential Equations in computer vision and optimal control [7], the number of features in machine learning applications [30], the number of states in a Markov Decision Processes [26], and so on. Indeed anytime a finite dimensional optimization models arises from an infinite dimensional model it is straightforward to define such a hierarchy of optimization models. In many areas it is common to take advantage of this structure by solving a low fidelity (coarse) model and then use the solution as a starting point in the high fidelity (fine) model (see e.g. [13,15] for examples from computer vision). In this paper we adopt an optimization point of view and show how to take advantage of the availability of a hierarchy of models in a consistent manner for composite convex optimization. We do not use the coarse model just for the computation of promising starting points but also for the computation of search directions.

The algorithm we propose is similar to the *Iterative Shrinkage Thresholding Algorithm* (ISTA) class of algorithms. There is a substantial amount of literature related to this class of algorithms and we refer the reader to [2] for a review of recent developments. The main difference between ISTA and the algorithm we propose is that we use both gradient information and a coarse model in order to compute a search direction. This modification of ISTA for the computation of the search direction is akin to multigrid algorithms developed recently by a number of authors. There exists a considerable number of papers exploring the idea of using multigrid methods in optimization [7]. However the large majority of these are concerned with solving the linear system of equations to compute a search direction using linear multigrid methods (both geometric and algebraic). A different approach, and the one we adopt in this paper is the class of multigrid algorithms proposed in [20] and further developed in [19]. The framework proposed in [20] was used for the design of a first order unconstrained line search algorithm in [31], and a trust region algorithm in [12]. The trust region framework was extended to deal with box constraints in [11]. The general equality constrained case was discussed in [21], but no convergence proof was given. Numerical experiments with multigrid are encouraging and a number of numerical studies have appeared so far, see e.g. [10,22]. The algorithm we develop combines elements from ISTA (gradient proximal steps) and the multigrid framework (coarse correction steps) developed in [20] and [31]. We call the proposed algorithm *Multilevel Iterative Shrinkage Thresholding Algorithm* (MISTA). We prefer the name multilevel to multigrid since there is no notion of grid in our algorithm.

The literature in multilevel optimization is largely concerned with models where the underlying dynamics are governed by differential equations and convergence proofs exist only for the smooth case and with simple box or equality constraints. Our main contribution is the extension of the multigrid framework for convex but possibly non-smooth problems with certain types of constraints.

Theoretically the algorithm is valid for any convex constraint but the algorithm is computationally feasible when the proximal projection step can be performed in closed form or when it has a low computational cost. Fortunately many problems in machine learning, computer vision, and statistics do satisfy our assumptions. Apart from the work in [11] that addresses box constraints, the general constrained case has not been addressed before. Existing approaches assume that the objective function is twice continuously differentiable, while the the proximal framework we develop in this paper allows for a large class of non-smooth optimization models. In addition, our convergence proof is different from the one given in [20] and [6] in that we do not assume that the algorithm used in the finest scale performs one iteration after every coarse correction step. Our proof is based on analyzing the whole sequence generated by the algorithm and does not rely on asymptotic results as in previous works [12,31]. We show that the coarse correction step satisfies the contraction property as long as the objective function is convex and the differentiable part has a Lipschitz continuous gradient. If the differentiable part is *strongly convex*, then MISTA has a Q-linear convergence rate [25]. On the other hand, if the differentiable part is only convex and has Lipschitz continuous gradients, then MISTA has an R-linear convergence rate. R-linear convergence rate is a property of ISTA, and is weaker than Q-linear convergence. A variant of ISTA is the Fast Iterative Shrinkage Thresholding Algorithm (FISTA) proposed in [2], and has a convergence rate of $O(1/k^2)$ for function values. The analysis of FISTA using the multilevel framework is technically more challenging than the simpler ISTA scheme. The acceleration of multigrid methods is an open question that is currently under investigation. Indeed many algorithmic frameworks for large scale composite convex optimization such as active set methods[18], stochastic methods [16], Newton type methods [17] as well as block coordinate descent methods [27] have recently been proposed. In principle all these algorithmic ideas could be combined with the multilevel framework developed in this paper. We chose to study ISTA because it is simpler to analyze. With the insights provided in this paper we hope to combine the multilevel framework with more advanced algorithms in the future. Despite the theoretical differences between the algorithm proposed in this paper and FISTA, our numerical experiments show that MISTA outperforms both ISTA and FISTA. In particular we found that for a difficult large scale (over 10^6 variables) image restoration problem MISTA is ten times faster than ISTA and more than three times faster than FISTA.

Outline The rest of the paper is structured as follows: in the next section we introduce our notation and assumptions. We also discuss the role of quadratic approximations in convex composite optimization models. In Section 3 we discuss the construction of different coarse models. We also describe the process of transferring information from a coarse to a fine model and vice versa. The full algorithm is given in Section 3.3 and the convergence of the algorithm is established in Section 4. We report numerical results in Section 5.

2 Composite Convex Optimization and Quadratic Approximations

The main difference between the proposed algorithm, MISTA, and existing algorithms such as ISTA and FISTA is that we do not use a quadratic approximation

for all iterations. Instead we use a coarse model approximation for some iterations. In this section we briefly describe the role of quadratic approximations in composite convex optimization, and introduce our notation.

2.1 Notation and Problem Description

We will assume that the optimization model can be formulated using only two levels of fidelity, a fine model and a coarse model. We use h and H to indicate whether a particular quantity/property is related to the fine and coarse model respectively. It is easy to generalize the algorithm to more levels but with only two levels the notation is simpler. The fine model is the convex composite optimization model,

$$\min_{x_h \in \Omega_h} \left\{ F_h(x) \triangleq f_h(x_h) + g_h(x_h) \right\}, \tag{1}$$

where $\Omega_h \subset \mathbb{R}^h$ is a closed convex set, f_h is a smooth function with a Lipschitz continuous gradient, and $g_h : \mathbb{R}^h \to \mathbb{R}$ is an extended value convex function that is possibly non-smooth. We use L_h to denote the Lipschitz constant of the gradient of f_h. When g_h is a norm then the non-smooth term in (1) is usually multiplied by a scalar $\mu \geq 0$. The parameter μ is a regularization parameter, and so the non-smooth term encourages solutions that are sparse. Sparsity is a desirable property in many applications. The algorithm we propose does not only apply when g_h is a norm. But if it is a norm, then some variants of our algorithm make use of the dual norm associated with g_h. The incumbent solution at iteration k in resolution h is denoted by $x_{h,k}$. We use $f_{h,k}$ and $\nabla f_{h,k}$ to denote $f_h(x_{h,k})$ and $\nabla f_h(x_{h,k})$ respectively. Unless otherwise specified we use $\|.\|$ to denote $\|.\|_2$.

2.2 Quadratic Approximations and ISTA

A widely used method to update $x_{h,k}$ is to perform a quadratic approximation of the smooth component of the objective function, and then solve the *proximal subproblem*,

$$x_{h,k+1} = \arg\min_{y \in \Omega_h} f_{h,k} + \langle \nabla f_{h,k}, y - x_{h,k} \rangle + \frac{L_h}{2} \|x_{h,k} - y\|^2 + g(y).$$

Note that the above can be rewritten as follows,

$$x_{h,k+1} = \arg\min_{y \in \Omega_h} \frac{L_h}{2} \left\| y - \left(x_{h,k} - \frac{1}{L_h} \nabla f_{h,k} \right) \right\|^2 + g(y).$$

When the Lipschitz constant is known, ISTA keeps updating the solution vector by solving the optimization problem above. Another example is the classical gradient projection algorithm[5] (with a fixed step-size). In this case the proximal projection step is given by,

$$\min_{y \in \mathbb{R}^h} \frac{L_h}{2} \left\| y - \left(x_{h,k} - \frac{1}{L_h} \nabla f_{h,k} \right) \right\|^2 + I_{\Omega_h}(y),$$

where I_{Ω_h} is the indicator function on Ω_h. For later use we define the generalized *proximal operator* as follows,

$$\text{prox}_h(x) = \arg\min_{y \in \Omega_h} \frac{1}{2} \|y - x\|^2 + g(y).$$

Our algorithm uses the step-size differently than ISTA/FISTA and so in proximal steps the step-size does not appear explicitly in the definition of the proximal projection problem. Our proximal update step is given by,

$$x_{h,k+1} = x_{h,k} - s_{h,k} D_{h,k} \tag{2}$$

where the *gradient mapping* $D_{h,k}$ is defined as follows,

$$D_{h,k} \triangleq \left[x_{h,k} - \text{prox}_h(x_{h,k} - \frac{1}{L_h} \nabla f_{h,k}) \right]. \tag{3}$$

Updating the incumbent solution in this manner is reminiscent of classical gradient projection algorithms [5].

In many applications g_h is a norm, and it is often necessary to refer explicitly to the regularization parameter,

$$\min_{x_h \in \Omega_h} \left\{ F_h(x) \triangleq f_h(x_h) + \mu g_h(x_h) \right\}. \tag{4}$$

For the case where the optimization model is given by (4), we will also make use of the properties of the *dual norm proximal operator* defined as follows,

$$\begin{aligned} \text{proj}_h^\star(x) = \arg\max_y \; & -\frac{1}{2} \|y - x\|_2^2 - \|x\|^2 \\ \text{s.t. } & g^\star(y) \leq \mu, \end{aligned} \tag{5}$$

where $g^\star$ is the dual norm of g. Using Fenchel duality (see Lemma 2.3 in [29]) it can be shown that,

$$\text{prox}_h(x) = x - \text{proj}_h^\star(x). \tag{6}$$

The relationship above is often used to compute the proximal projection step efficiently.

3 Multilevel Iterative Shrinkage Thresholding Algorithm

Rather than computing a search direction using a quadratic approximation, we propose to construct an approximation with favorable computational characteristics for at least some iterations. Favorable computational characteristics in the context of optimization algorithms may mean reducing the dimensions of the problem and possibly increasing the smoothness of the model. This approach facilitates the use of non-linear (but still convex) approximations around the current point. The motivation behind this class of approximations is that the global nature of the approximation would reflect global properties of the model that would yield better search directions.

There are three components to the construction of the proposed algorithm: (a) specification of the restriction/prolongation operators that transfer information between different levels; (b) construction of an appropriate hierarchy of models; (c) specification of the algorithm (smoother) to be used in the coarse model. Below we address these three components in turn.

3.1 Information transfer between levels

Multilevel algorithms require information to be transferred between levels. In the proposed algorithm we need to transfer information concerning the incumbent solution, proximal projection and gradient around the current point. At the fine level the design vector x_h is a vector in $\mathbb{R}^h$. At the coarse level the design vector is a vector in $\mathbb{R}^H$ and $H < h$. At iteration k, the proposed algorithm projects the current solution $x_{h,k}$ from the fine level to coarse level to obtain an initial point for the coarse model denoted by $x_{H,0}$. This is achieved using a suitably designed matrix (I_h^H) as follows,

$$x_{H,0} = I_h^H x_{h,k}.$$

The matrix $I_h^H \in \mathbb{R}^{H\times h}$, is called a *restriction operator* and its purpose is to transfer information from the fine to the coarse model. There are many ways to define this operator and we will discuss some possibilities for machine learning problems in Section 4. This is a standard technique in multigrid methods both for solutions of linear and nonlinear equations and for optimization algorithms [9,20]. In addition to the restriction operator we also need to transfer information from the coarse model to the fine model. This is done using the *prolongation operator* $I_H^h \in \mathbb{R}^{h\times H}$. The standard assumption in multigrid literature [9] is to assume that $I_h^H = c(I_H^h)^\top$, where c is some positive scalar. We also assume, with out loss of generality, that $c = 1$. We also make the following assumption, that is always satisfied in practice.

Assumption 1 *For a given pair of restriction/prolongation operators, there exist two constants ω_1 and ω_2, such that:*

$$\|I_h^H y_h\| \leq \omega_1 \|y_h\|$$
$$\|I_H^h y_H\| \leq \omega_2 \|y_H\|$$

for any vectors y_h in the fine level, and y_H in the coarse level.

3.2 Coarse model construction

The construction of the coarse models in multilevel algorithms is a subtle process. It is this process that sets apart rigorous multilevel algorithms with performance guarantees from other approaches (e.g. kriging methods) used in the engineering literature. A key property of the coarse model is that locally (i.e. at the initial point of the coarse model, $x_{H,0}$) the optimality conditions of the two models match. In the unconstrained case this is achieved by adding a linear term in the objective function of the coarse model [12,20,31]. In the constrained case the linear term is used to match the gradient of the Lagrangian [20]. However, the theory for the constrained case of multilevel algorithms is less developed. Here we propose an approach that contains the unconstrained approach in [20] and the box-constrained case [11] as special cases. In addition we are able to deal with the non–smooth case and through the proximal step we address the constrained case.

In the case where the optimization model is non–smooth there are many ways to construct a coarse model. We propose three ways to address the non–smooth part of the problem. All three approaches enjoy the same convergence properties,

but depending on the application some coarse models may be more appropriate since they make different assumptions regarding the non–smooth function and the prolongation/restriction operators. The three approaches are: (a) smoothing the non–smooth term, (b) a reformulation using dual norm projection, (c) non–smooth model with a projection using the indicator function. The coarse model in all three approaches has the following form,

$$F_H(x_H) \triangleq f_H(x_H) + g_H(x_H) + \langle v_H, x_H \rangle. \tag{8}$$

We assume that given the function f_h, the construction of f_H is easy (e.g. varying a descritization parameter or the resolution of an image etc.) . We also assume that f_H has a Lipschitz continuous gradient, and denote the Lipschitz constant with L_H. The second term in (8) represents information regarding the non–smooth part of the original objective function, and the third term ensures the fine and coarse model are coherent (in the sense of Lemmas 1-3). We will denote the smooth part of the objective function with,

$$\phi_H(x_H) \triangleq f_H(x_H) + \langle v_H, x_H \rangle.$$

We use L_H to denote the Lipschitz constant of the gradient of ϕ_H. Apart from f_H, the other two terms in (8) vary depending on which of the three approaches is adopted. We discuss the three options in decreasing order of generality below.

3.2.1 The smooth coarse model

The approach that requires the least assumptions is to construct a coarse model by smoothing the non–smooth part of the objective function. In other words, the second term in (8) is again a reduced order version of g_h but is also smooth. In the application we consider the non-smooth term is usually a norm or an indicator function. It is therefore easy to construct a reduced order version of g_h, and there exists many methods to smooth a non–smooth function [3]. Our theoretical results do not depend on the choice of the smoothing method. We construct the last term in (8) with,

$$v_H = L_H I_h^H D_{h,k} - (\nabla f_{H,0} + \nabla g_{H,0}). \tag{9}$$

When the coarse model is smooth, then L_H corresponds to the Lipschitz constant of (8). In addition we also assume that any constraints in the form of $x_H \in \Omega_H$ have been incorporated in g_H.

Lemma 1 *Suppose that f_H and g_H have Lipschitz continuous gradients, and that the coarse model associated with* (1) *is given by,*

$$\min_{x_H} \; f_H(x_H) + g_H(x_H) + \langle v_H, x_H \rangle, \tag{10}$$

where v_H is given by (9)*, then,*

$$D_{H,0} = I_h^H D_{h,k}. \tag{11}$$

Proof Using the definitions of the gradient mapping in (3) and the projection operator (instead of the prox operator) for the smooth objective function of the coarse level, we obtain:

$$\begin{aligned} D_{H,0} &= x_{H,0} - \text{prox}_H(x_{H,0} - \frac{1}{L_H}\nabla F_{H,0}) \\ &= x_{H,0} - \arg\min_{z\in\mathbb{R}^H} \frac{1}{2}\|z - \left(x_{H,0} - \frac{1}{L_H}\nabla F_{H,0}\right))\|^2 \\ &= \frac{1}{L_H}\nabla F_{H,0} \\ &= \frac{1}{L_H}(\nabla f_{H,0} + \nabla g_{H,0} + v_H) \\ &= I_h^H D_{h,k}, \end{aligned}$$

where in the second equality we used the fact that the objective function in (10) is smooth and so any constraints in the form of $x_H \in \Omega_H$ can be incorporated in g_H. □

The condition in (11) is referred to as the *first order coherent condition*. It ensures that at if $x_{h,k}$ is optimal in the fine level, then $x_{H,0} = I_h^H x_{h,k}$ is optimal in the coarse model. This property is crucial in establishing convergence of multilevel algorithms. The smooth case was discussed in [12,20,31], and the Lemma above extends the condition to the non-smooth case. Next we discuss a different way to construct the coarse model (and hence a different v_H term) that makes a particular assumption about the restriction and interpolation operators.

3.2.2 A non-smooth coarse model with dual norm projection

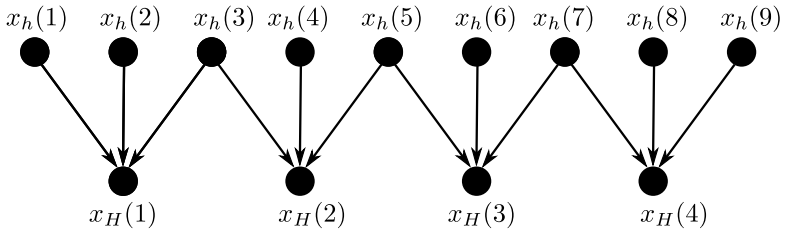
In the coarse construction method described above we imposed a restriction on the coarse model but allowed arbitrary restriction/prolongation operators. In our second method for constructing coarse models we allow for arbitrary coarse models (they can be non-smooth) but make a specific assumption regarding the information transfer operators. In particular we assume that,

$$x_H(i) = (I_h^H x_h)_i = x_h(2i), \quad i = 1, \ldots, H.$$

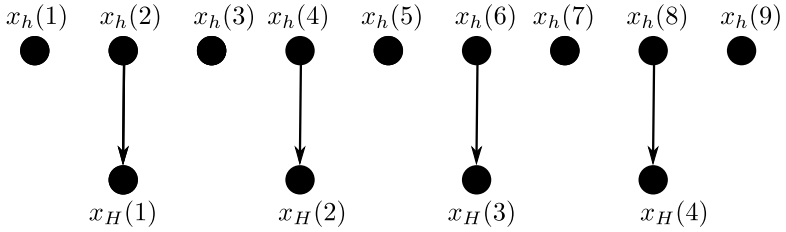
We refer to this operator as a *coordinate wise restriction operator*. The reason we discuss this class of restriction operators is that in the applications we consider the non-smooth term is usually a norm that satisfies the following,

$$\text{proj}_H^\star(I_h^H x_h) = I_h^H \text{proj}_h^\star(x_h), \tag{12}$$

where $\text{proj}_h^\star$ and $\text{proj}_H^\star$ denote projection with respect to the dual norm associated with g_h and g_H respectively (see the definition in (5)). When the restriction operator is done coordinate wise then the preceding equation is satisfied for many frequently encountered norms including the l_1, l_2 and the l_∞ norms. In the multigrid literature linear interpolation is the most frequently used restriction operator. In Figure 1 we compare the linear interpolation operator with the coordinate wise



(a) Linear interpolation restriction operator



(b) Coordinate wise restriction operator

Fig. 1 (a) The linear interpolation operator widely used in the multigrid literature. (b) The coordinate wise restriction operator is reminiscent of the techniques used in coordinate descent algorithms.

operator in terms of the information they transfer from the fine to the coarse model. In our second coarse construction method the last term in (8) is constructed with,

$$v_H = \frac{L_H}{L_h} I_h^H \nabla f_{h,k} - \nabla f_{H,0}. \tag{13}$$

Lemma 2 *Suppose that f_H has a Lipschitz continuous gradient, condition* (12) *is satified, and that both g_h and g_H are norms. For the coarse model associated with* (4) *given by,*

$$\min_{x_H} \; f_H(x_H) + \mu g_H(x_H) + \langle v_H, x_H \rangle,$$

where v_H is given by (13)*, then,*

$$D_{H,0} = I_h^H D_{h,k}.$$

Proof Since g_h is a norm, we can compute the proximal term by (6) to obtain,

$$\begin{aligned} D_{h,k} &= \left[x_{h,k} - \operatorname{prox}_h \left(x_{h,k} - \frac{1}{L_h} \nabla f_{h,k} \right) \right] \\ &= \left[x_{h,k} - \left(x_{h,k} - \frac{1}{L_h} \nabla f_{h,k} - \operatorname{proj}_h^\star \left(x_{h,k} - \frac{1}{L_h} \nabla f_{h,k} \right) \right) \right] \\ &= \frac{1}{L_h} \nabla f_{h,k} + \operatorname{proj}_h^\star \left(x_{h,k} - \frac{1}{L_h} \nabla f_{h,k} \right). \end{aligned}$$

Using the same argument for the coarse model and the definition in (13),

$$\begin{aligned} D_{H,0} &= \frac{1}{L_H}\left(\nabla f_{H,0} + v_H\right) + \text{proj}_H^\star\left(x_{H,0} - \frac{1}{L_H}\left(\nabla f_{H,0} + v_H\right)\right) \\ &= I_h^H\left(\frac{1}{L_h}\nabla f_{h,k}\right) + \text{proj}_H^\star\left(I_h^H\left(x_{h,k} - \frac{1}{L_h}\nabla f_{h,k}\right)\right) \\ &= I_h^H\left(\frac{1}{L_h}\nabla f_{h,k} + \text{proj}_h^\star(x_{h,k} - \frac{1}{L_h}\nabla f_{h,k})\right) \\ &= I_h^H D_{h,k}. \end{aligned}$$

Where in the third equality we used (12). □

Next we discuss a different way to construct the coarse model (and hence a different v_H term) that makes a particular assumption on the non-smooth component of the fine model.

3.2.3 A non-smooth coarse model with constraint projection.

When the non-smooth term is a regularization term, the proximal term is computationally tractable. In this case, the problem can equivalently be formulated using a constraint as opposed to a penalty term. In this third method for constructing coarse models we assume that the coarse non-smooth term is given by,

$$g_H(x_H) = \begin{cases} x_H \text{ if } x_H \in \Omega_H, \\ \infty \text{ otherwise.} \end{cases}$$

With this definition, the coarse model has the same form as in (8) where g_H is an indicator function on Ω_H, and the final term is constructed using the following definition for v_H,

$$v_H = L_H x_{H,0} - \left(\nabla f_{H,0} + L_H I_h^H \text{prox}_h\left(x_{h,k} - \frac{1}{L_h}\nabla f_{h,k}\right)\right). \tag{14}$$

We also make the following assumption regarding the relationship between coarse and fine feasible sets,

$$\text{proj}_H(I_h^H x_h) = I_h^H x_h, \ \forall x_h \in \Omega_h. \tag{15}$$

The condition above is satisfied for many situations of interest, for example when $\Omega_h = \mathbb{R}_+^h$ and $\Omega_H = \mathbb{R}_+^H$. It also holds for box constraints and simple linear or convex quadratic constraints. If the condition above is not possible to verify then the other two methods described in this section can still be used. Note that we only make this assumption regarding the coarse model, i.e. we do not require such a condition to hold when we prolong feasible coarse solutions to the fine model.

Lemma 3 *Suppose that that condition* (15) *is satisfied, f_H has a Lipschitz continuous gradient and that g_H is an indicator function on $\Omega_H \subset \mathbb{R}^H$. Assume that the coarse model associated with* (1) *is given by,*

$$\min_{x_H} \ f_H(x_H) + g_H(x_H) + \langle v_H, x_H \rangle,$$

where v_H is given by (14)*, then*

$$D_{H,0} = I_h^H D_{h,k}.$$

Proof Using the fact that the proximal step in the coarse model reduces to an orthogonal projection on Ω_H we obtain,

$$\begin{aligned} D_{H,0} &= x_{H,0} - \mathrm{proj}_H(x_{H,0} - \frac{1}{L_H}(\nabla f_{H,0} + v_H)) \\ &= x_{H,0} - \mathrm{proj}_H(I_h^H \mathrm{prox}_h(x_{h,k} - \frac{1}{L_h}\nabla f_{h,k})) \\ &= I_h^H \left[x_{x,k} - \mathrm{prox}_h(x_{h,k} - \frac{1}{L_h}\nabla f_{h,k}) \right] \\ &= I_h^H D_{h,k}, \end{aligned}$$

where in the third equality we used assumption (15). □

3.3 Algorithm Description

In the previous section we described ways to construct a coarse model, and specified the information transfer operators. Given these two components we are now in a position to describe the algorithm in full. It does not matter how the coarse model or the information transfer operators were constructed. The only requirement is that the first order coherence condition is satisfied. It is important to satisfy the first order coherent condition in order to establish the convergence of the algorithm. However, it does not matter how this condition is imposed in the coarse model. The prolongation/restriction operators are also satisfy assumed to $I_h^H = c(I_H^h)^\top$ for some constant $c > 0$ (with out loss of generality we assume that $c = 1$). The latter assumption is standard in the literature of multigrid methods.

Given an initial point $x_{H,0}$, the coarse model is solved in order to obtain a so called *error correction term*. The error correction term is the vector that needs to be added to the initial point of the coarse model in order to obtain an optimal solution $x_{H,\star}$ in (8),

$$e_{H,\star} = x_{H,0} - x_{H,\star}.$$

In practice the error correction term is only approximately computed, and instead of $e_{H,\star}$ we will use $e_{H,m}$, i.e. the error correction term after m iterations. After the coarse error correction term is computed, it is projected to the fine level using the prolongation operator,

$$d_{h,k} = I_H^h e_{H,m} \triangleq I_H^h \sum_{i=0}^{m-1} s_{H,i} D_{H,i}.$$

The current solution, at the fine level, is updated as follows,

$$x_{h,k+1} = x_{h,k} - s_{h,k}(x_{h,k} - x_h^+),$$

where,

$$x_h^+ = \mathrm{prox}_h(x_{h,k} - \tau d_{h,k}).$$

Algorithm 1: Multilevel Iterative Shrinkage Thresholding Algorithm

if Condition to compute search direction in the coarse model is satisfied at $x_{h,k}$ **then**

Set $x_{H,0} = I_h^H x_{h,k}$;

Compute m iterations of the coarse level

$$x_{H,m} = x_{H,0} + \sum_{i=0}^{m} s_{H,i} D_{H,i}$$

Set $d_{h,k} = I_H^h(x_{H,0} - x_{H,m})$;

Find a suitable τ and compute:

$$x^+ = \text{prox}_h(x_{h,k} - \tau d_{h,k}) \tag{16}$$

Choose a step-size $s_{h,k} \in (0, 1]$ and update:

$$x_{h,k+1} = x_{h,k} - s_{h,k}(x_{h,k} - x_h^+) \tag{17}$$

end

else

Compute gradient mapping:

$$D_{h,k} = x_{h,k} - \text{prox}_h(x_{h,k} - \frac{1}{L_h} d_{h,k})$$

Choose a step-size $s_{h,k} \in (0, 1]$ to update:

$$x_{h,k+1} = x_{h,k} - s_{h,k} D_{h,k} \tag{18}$$

end

Clearly, if $d_{h,k} = \nabla f_{h,k}$, $\tau = 1/L_H$, then the algorithm performs exactly the same step as ISTA with the proximal update step given in (2). Below we specify a conceptual version of the algorithm. When the current iterate $x_{h,k}$ is updated using the error correction term from the coarse model we call the step $k + 1$ a *coarse correction step*.

Based on our own numerical experiments and the results in [12,20,31] we perform a coarse correction iteration when the following conditions are satisfied,

$$\begin{aligned} \|I_h^H D_{h,k}\| &> \kappa \|D_{h,k}\| \\ \|x_h^k - \tilde{x}_h\| &> \eta \|\tilde{x}_h\|, \end{aligned} \tag{19}$$

where $\tilde{x}_h$ is the last point to triger a coarse correction iteration. The first condition in (19) prevents the algorithm from performing coarse iterations when the first order optimality conditions are almost satisfied. If the current fine level iterate is close to being optimal the coarse model constructs a correction term that is nearly zero. Typically, κ is the tolerance on the norm of the first-order optimality condition of (the fine) level h or alternatively $\kappa \in (0, \min(1, \min \|I_h^H\|))$. The second condition in (19) prevents a coarse correction iteration when the current point is very close to $\tilde{x}_h$. The motivation is that performing a coarse correction at a point x_h^k that satisfies both the above conditions will yield a new point close to the current x_h^k. In our implementation of MISTA we always use ISTA to perform iterations in both the coarse level and in the fine level (when a gradient mapping

is performed). It is possible to obtain better numerical performance by performing FISTA steps but since this type of steps are not covered by our theory we leave this enhancement for future work.

4 Global convergence rate analysis

In this section we establish the convergence and convergence rate of MISTA. Our main result (Theorem 3) shows that a coarse correction step is a contraction on the optimal solution. To establish our main result we need to assume that both the fine and coarse model are convex, but not necessarily strongly convex. Our other main assumption is that the differentiable part of the fine and coarse models have Lipschitz-continuous gradients. Based on existing results on ISTA it will then follow that MISTA converges with an R-linear convergence rate when $f(x)$ is convex. In addition when $f(x)$ is *strongly* convex, MISTA converges Q-linearly.

For the convergence analysis it does not matter how the coarse model is constructed. We only require the first order coherence property to hold,

$$D_{H,0} = I_h^H D_{h,k}. \tag{20}$$

Where $D_{h,k}$ is the gradient mapping defined in (3). Three examples of how this property can be satisfied are given in Lemmas 1, 2, and 3. If conditions (19) are satisfied the proposed algorithm performs a coarse correction (17). If (19) are not satisfied then MISTA performs a gradient (mapping) proximal step(18). In order to establish the convergence property of MISTA, we will show in Theorem 3 that if $x_{h,\star}$ is the optimal solution for (1), then the coarse correction step is always a contraction,

$$\|x_{h,k+1} - x_{h,\star}\|^2 \leq \sigma \|x_{h,k} - x_{h,\star}\|^2, \tag{21}$$

where $\sigma \in (0,1)$. In addition, the gradient proximal step (18) is non-expansive if $f(x)$ is convex, and is a contraction if $f(x)$ is strongly convex [28]. Clearly, the contraction property is stronger than the non-expansive property; therefore, combining this with the contraction property of the coarse correction step, MISTA converges Q-linearly if $f(x)$ is strongly convex, and R-linearly otherwise. The following theorems follow from [25,28] and establish the convergence properties of MISTA.

Theorem 1 *[28, Theorem 1] Suppose that the coarse correction step satisfies the contraction property* (21)*, and that $f(x)$ is convex and has Lipschitz-continuous gradients. Then any MISTA step is at least nonexpansive (coarse correction steps are contractions and gradient proximal steps are non-expansive),*

$$\|x_{h,k+1} - x_{h,\star}\|^2 \leq \|x_{h,k} - x_{h,\star}\|^2,$$

and the sequence $\{x_{h,k}\}$ converges R-linearly.

Theorem 2 *[28, Theorem 2] Suppose that the coarse correction step satisfies the contraction property* (21)*, and that $f(x)$ is strongly convex and has Lipschitz-continuous gradients. Then any MISTA step is always a contraction,*

$$\|x_{h,k+1} - x_{h,\star}\|^2 \leq \sigma \|x_{h,k} - x_{h,\star}\|^2,$$

where $\sigma \in (0,1)$ and the sequence $\{x_{h,k}\}$ converges Q-linearly.

If the coarse correction step is a contraction, the results above establish the linear convergence rate of MISTA. In the rest of this section we show the contraction property of the coarse correction step (17).

The first observation is that at the optimum $x_{h,\star}$ the gradient mapping, denoted by $D_{h,\star}$ is zero. This follows from the optimality conditions of proximal type algorithms and can be found in [4]. It then follows from the first order coherence property, $D_{H,\star} = I_h^H D_{h,\star}$ that the coarse correction step is zero when $D_{h,\star}$ is the gradient mapping at the optimum $x_{h,\star}$. This trivial observation is formalized in the Lemma below.

Lemma 4 *Suppose that $x_{h,\star}$ is optimal for (1). Let $D^\star_{H,i}$ denote the gradient mapping of the coarse model at iteration i when $x_{H,0} = I_h^H x_{h,\star}$. Then for all iterations i of the coarse model we must have $D^\star_{H,i} = 0$.*

Convergence proofs for first order algorithms take advantage of the following inequality,

$$\langle x_h - y_h, \nabla f(x) - \nabla f(y)\rangle \geq \frac{1}{L}\|\nabla f(x) - \nabla f(y)\|^2. \tag{22}$$

The proof of the preceding inequality can be found in [23], and it uses the facts that f is convex and has a Lipschitz continuous gradient. In our proof we will need to make use of such an inequality. However, the direction the algorithm uses is not always given by the gradient of the function. For some iterations MISTA uses a coarse correction step, and we simply cannot replace the gradients in (22) with the coarse correction term obtained from the coarse model. We are still able to establish a similar inequality in Lemma 6. In particular the main result in this section will be obtained using the following bound,

$$\langle x_{h,k} - x_{h,\star}, d_{h,k} - d_{h,\star}\rangle \geq \widehat{m}\|d_{h,k} - d_{h,\star}\|, \tag{23}$$

where $\widehat{m}$ is specified in Lemma 6. Note that if we only perform gradient mapping steps (see (18)) then $d_{h,k} = \nabla f_h(x_k)$ and $d_{h,\star} = \nabla f_h(x_{h,\star})$ and the preceding inequality simply follows from (22). However, when we perform coarse correction steps (see (17)) then $d_{h,k}$ is the sum of m applications of the proximal operator,

$$d_{h,k} = I_H^h \sum_{i=0}^{m-1} s_{H,i} D_{H,i}.$$

Obtaining the bound in (23) is not as easy as in the case where gradient steps are made. The bound in (23) makes use of the following lemma established in [1].

Lemma 5 *Suppose that the function $\phi : \Omega \to \mathbb{R}$ is convex with a L-lipschitz continuous gradient. Let D_z denote the gradient mapping defined in (3) at the point $z \in \Omega$ i.e.*

$$D_z = \left[z - prox(z - \frac{1}{L}\nabla\phi(z)) \right].$$

Then for any $x, y \in \Omega$, we must have,

$$\langle D_x - D_y, x - y\rangle \geq \frac{3}{4}\|D_x - D_y\|^2. \tag{24}$$

Proof The proof in [1] was given for a different definition of the gradient mapping but exactly the same proof can be used to establish (24). □

Next we use the Lemma above together with properties of the multilevel proximal mapping to establish the bound in (23).

Lemma 6 *Consider two coarse correction terms generated by performing m iterations at the coarse level starting from the points $x_{h,k}$ and $x_{h,\star}$:*

$$\begin{aligned} d_{h,k} &= I_H^h e_{H,m} = I_H^h \sum_{i=0}^{m-1} s_{H,i} D_{H,i} \\ d_{h,\star} &= I_H^h \sum_{i=0}^{m-1} s_{H,i} D_{H,i}^\star = 0 \end{aligned} \tag{25}$$

then the following inequality holds:

$$\langle x_{h,k} - x_{h,\star}, d_{h,k} - d_{h,\star} \rangle \geq \widehat{m} \| d_{h,k} - d_{h,\star} \|^2$$

where $\widehat{m} = (1+2m)/(4m\omega_2^2)$, and ω_2 was defined in Assumption (1).

Proof From Lemma 4, we must have that $D_{H,i}^\star = 0, \forall i$ and therefore,

$$d_{h,\star} = I_H^h \sum_{i=0}^{m-1} s_{H,i} D_{H,i}^\star = 0.$$

Using the observation above, we obtain the following equality,

$$\begin{aligned} \langle x_{h,k} - x_{h,\star}, d_{h,k} - d_{h,\star} \rangle &= \left\langle x_{h,k} - x_{h,\star}, I_H^h \sum_{i=0}^{m-1} s_{H,i} D_{H,i} - I_H^h \sum_{i=0}^{m-1} s_{H,i} D_{H,i}^\star \right\rangle \\ &= \left\langle x_{H,0} - x_{H,0}^\star, \sum_{i=0}^{m-1} s_{H,i} (D_{H,i} - D_{H,0}^\star) \right\rangle \end{aligned} \tag{26}$$

Consider the i^{th} term of the preceding equation,

$$\begin{aligned} s_{H,i} \left\langle x_{H,0} - x_{H,0}^\star, D_{H,i} - D_{H,0}^\star \right\rangle =& s_{H,i} \langle x_{H,0} - x_{H,i} + x_{H,i} - x_{H,0}^\star, D_{H,i} - D_{H,0}^\star \rangle \\ \geq& s_{H,i} \langle x_{H,0} - x_{H,i}, D_{H,i} \rangle + \frac{3}{4} s_{H,i} \| D_{H,i} - D_{H,0}^\star \|^2 \\ =& \langle x_{H,0} - x_{H,i}, x_{H,i} - x_{H,i+1} \rangle + \frac{3}{4 s_{H,i}} \| x_{H,i} - x_{H,i+1} \|^2 \\ \geq& \langle x_{H,0} - x_{H,i}, x_{H,i} - x_{H,i+1} \rangle + \frac{3}{4} \| x_{H,i} - x_{H,i+1} \|^2. \end{aligned}$$

Where the first inequality follows from Lemma 5 and the fact that $D_{H,0}^\star = 0$. To obtain the second equality we used the following,

$$D_{H,i} = \frac{x_{H,i} - x_{H,i+1}}{s_{H,i}}$$

Finally the last inequality above follows from the fact that $s_{H,i} \in (0,1]$. Substituting the bound we obtained for the i^{th}above inequality in (26) yields:

$$\begin{aligned}\langle x_{h,k} - x_{h,\star}, d_{h,k} - d_{h,\star}\rangle &\geq \sum_{i=0}^{m-1} \langle x_{H,0} - x_{H,i}, x_{H,i} - x_{H,i+1}\rangle + \frac{3}{4}\|x_{H,i} - x_{H,i+1}\|^2 \\ &= \Delta + \sum_{i=2}^{m-1} \langle x_{H,0} - x_{H,i}, x_{H,i} - x_{H,i+1}\rangle + \frac{3}{4}\|x_{H,i} - x_{H,i+1}\|^2. \end{aligned} \tag{27}$$

where,

$$\Delta = \frac{3}{4}\|x_{H,0} - x_{H,1}\|^2 + \langle x_{H,0} - x_{H,1}, x_{H,1} - x_{H,2}\rangle + \frac{3}{4}\|x_{H,1} - x_{H,2}\|^2.$$

The quantity Δ has the form:

$$\frac{3}{4}\|a\|^2 + \langle a, b\rangle + \frac{3}{4}\|b\|^2 = \frac{1}{2}\|a+b\|^2 + \frac{1}{4}\|a\|^2 + \frac{1}{4}\|b\|^2, \tag{28}$$

with $a = x_{H,0} - x_{H,1}$ and $b = x_{H,1} - x_{H,2}$. Utilizing (28) in (27) we obtain:

$$\begin{aligned}&\langle x_{h,k} - x_{h,\star}, d_{h,k} - d_{h,\star}\rangle \\ \geq\ & \frac{1}{4}\|x_{H,0} - x_{H,1}\|^2 + \frac{1}{4}\|x_{H,1} - x_{H,2}\|^2 \\ &+\frac{1}{2}\|x_{H,0} - x_{H,2}\|^2 + \langle x_{H,0} - x_{H,2}, x_{H,2} - x_{H,3}\rangle + \frac{1}{2}\|x_{H,2} - x_{H,3}\|^2 + \frac{1}{4}\|x_{H,2} - x_{H,3}\|^2 \\ &+ \sum_{i=3}^{m-1} \langle x_{H,0} - x_{H,i}, x_{H,i} - x_{H,i+1}\rangle + \frac{1}{2}\|x_{H,i} - x_{H,i+1}\|^2 + \frac{1}{4}\|x_{H,i} - x_{H,i+1}\|^2\end{aligned}$$

Note that,

$$\langle x_{H,0} - x_{H,i}, x_{H,i} - x_{H,i+1}\rangle + \frac{1}{2}\|x_{H,i} - x_{H,i+1}\|^2 = \frac{1}{2}\|x_{H,0} - x_{H,i+1}\|^2 - \frac{1}{2}\|x_{H,0} - x_{H,i}\|^2.$$

Using the preceding equality and grouping the remaining terms together we obtain,

$$\begin{aligned}\langle x_{h,k} - x_{h,\star}, d_{h,k} - d_{h,\star}\rangle &\geq \frac{1}{2}\|x_{H,0} - x_{H,m}\|^2 + \frac{1}{4}\sum_{i=0}^{m-1} \|x_{H,i} - x_{H,i+1}\|^2 \\ &\geq \frac{1}{2}\|x_{H,0} - x_{H,m}\|^2 + \frac{1}{4m}\left(\sum_{i=0}^{m-1} \|x_{H,i} - x_{H,i+1}\|\right)^2 \\ &\geq \frac{1}{2}\|x_{H,0} - x_{H,m}\|^2 + \frac{1}{4m}\|x_{H,0} - x_{H,m}\|^2 \\ &= \frac{1+2m}{4m}\|e_{H,m}\|^2\end{aligned}$$

Where to get the second inequality we used the Cauchy-Schwarz inequality, the third inequality follows from the triangle inequality. In the last equality we used

the definition of the coarse error correction term. The result now follows by using Assumption 1,

$$\frac{1+2m}{4m}\|e_{H,m}\|^2 \geq \frac{1+2m}{4m\omega_2^2}\left\|I_H^h e_{H,m}\right\|^2 = \frac{1+2m}{4m\omega_2^2}\|d_{h,k}-d_{h,\star}\|^2,$$

as required. □

Next we show that the coarse correction term satisfies a condition similar to the Lipschitz continuity of the gradient.

Lemma 7 *Suppose that a convergent algorithm with nonexpansive steps is applied at the coarse level (e.g. ISTA). Then the coarse correction term defined in* (25) *satisfies the following bound,*

$$\|d_{h,k}-d_{h,\star}\|^2 \leq \frac{16}{9}m^2\omega_1^2\omega_2^2 s_{H,0}^2\|x_{h,k}-x_{h,\star}\|^2,$$

where ω_1, ω_2 are defined in Assumption 1, m is the number of iterations in the coarse level, and $s_{H,0}$ is the step size used in the coarse algorithm.

Proof Using the definition of the coarse correction term we obtain,

$$\begin{aligned}\|d_{h,k}-d_{h,\star}\|^2 &= \left\|I_H^h\sum_{i=0}^{m-1}s_{H,i}D_{H,i}-0\right\|^2\\ &\leq \omega_2^2\left\|\sum_{i=0}^{m-1}s_{H,i}D_{H,i}\right\|^2\\ &\leq \omega_2^2\left(\sum_{i=0}^{m-1}s_{H,i}\|D_{H,i}\|\right)^2,\end{aligned} \tag{34}$$

where in the first inequality we used Assumption 1, and in the second inequality we used the triangle inequality. Since non-expansive steps are used at the coarse level we must have that,

$$\|x_{H,k+1}-x_{H,k}\| \leq \|x_{H,k}-x_{H,k-1}\|,$$

or equivalently,

$$s_{H,k}\|D_{H,k}\| \leq s_{H,k-1}\|D(x_{H,k-1})\|.$$

Using the preceding relationship we obtain,

$$\begin{aligned}\left(\sum_{i=0}^{m-1}s_{H,i}\|D_{H,i}\|\right)^2 &\leq m^2 s_{H,0}^2\|D_{H,0}\|^2\\ &= m^2 s_{H,0}^2\|D_{H,0}-D_{H,0}^\star\|^2\\ &= m^2 s_{H,0}^2\|I_h^H D_{h,k}-I_h^H D_{h,\star}\|^2\end{aligned} \tag{35}$$

where used the fact that $D^{\star}_{H,0} = 0$ in the first equality, and the first order coherence property (20) in the second equality. Using Assumption 1 and Lemma 5 we obtain,

$$\begin{aligned}\|I_h^H D_{h,k} - I_h^H D_{h,\star}\|^2 &\leq \omega_1^2 \|D_{h,k} - D_{h,\star}\|^2 \\ &\leq \frac{4}{3}\omega_1^2 \langle D_{h,k} - D_{h,\star}, x_{h,k} - x_{h,\star}\rangle \\ &\leq \frac{16}{9}\omega_1^2 \|x_{h,k} - x_{h,\star}\|^2.\end{aligned} \tag{36}$$

Using (35) and (36) in (34) we obtain the desired result. □

We are now in a position to show that the algorithm is a contraction even when coarse correction steps are used.

Theorem 3 (Contraction for coarse correction update) *Suppose that at iteration $k+1$ a coarse correction update is performed using m iterations of the coarse contraction algorithm.*

(a) Let τ denote the step size in (16) *and s denote the step size used in* (17) *then,*

$$\|x_{h,k+1} - x_{h,\star}\|^2 \leq \sigma(s,\tau)\|x_{h,k} - x_{h,\star}\|^2$$

where $\sigma(\tau, s) = 2 + \Delta(\tau)s^2$ and,

$$\Delta(\tau) = \frac{8}{9} m\omega_1^2 s_{H,0}^2 (4m\omega_2^2\tau^2 - 2\tau(1+2m)).$$

(b) Suppose that either of the following is true,
(i) $\omega \geq 1$ and $\omega_2 \leq 1$.
(ii) The number of iterations in the coarse algorithm is sufficiently large.
Then, there always exists $\tau > 0$ such that,

$$\Delta(\tau) < -1$$

and,

$$\frac{1}{\sqrt{-\Delta(\tau)}} \leq s \leq \min\left\{\frac{2}{\sqrt{-\Delta(\tau)}}, 1\right\}$$

Consequently, we have,

$$\sigma(\tau, s) < 1.$$

Proof (a) Let $r_{h,k}$ denote the difference between the optimum and iteration k i.e. $r_{h,k} = x_{h,k} - x_{h,\star}$. If a coarse correction step is performed at iteration $k+1$ we can bound the norm of $r_{h,k+1}$ as follows,

$$\begin{aligned}\|r_{h,k+1}\|^2 =& \|(1-s)r_{h,k} + s[\text{prox}_h(x_{h,k} - \tau d_{h,k}) - \text{prox}_h(x_{h,\star} - \tau d_{h,\star})]\|^2 \\ \leq& 2(1-s)^2\|r_{h,k}\|^2 + 2s^2\|\text{prox}_h(x_{h,k} - \tau d_{h,k}) - \text{prox}_h(x_{h,\star} - \tau d_{h,\star})\|^2 \\ \leq& 2(1-s)^2\|r_{h,k}\|^2 + 2s^2\|(x_{h,k} - \tau d_{h,k}) - (x_{h,\star} - \tau d_{h,\star})\|^2 \\ =& (4s^2 - 4s + 2)\|r_{h,k}\|^2 + 2s^2(\tau^2\|d_{h,k} - d_{h,\star}\|^2 - 2\tau\langle x_{h,k} - x_{h,\star}, d_{h,k} - d_{h,\star}\rangle),\end{aligned}$$

where the first inequality follows from Cauchy-Schwarz inequality and the non-expansive property of the proximal algorithm [28]. Using the fact that $s \in (0,1]$ implies that $4s^2 - 4s \leq 0$, and Lemma 6 in the bound for $r_{h,k+1}$ above, we obtain

$$\begin{aligned}\|r_{h,k+1}\|^2 \leq & 2\|x_{h,k} - x_{h,\star}\|^2 + \frac{4m\omega_2^2\tau^2 - 2\tau(1+2m)}{2m\omega_2^2} s^2 \|d_{h,k} - d_{h,\star}\|^2 \\ \leq & \left(2 + \underbrace{\frac{8}{9} m\omega_1^2 s_{H,0}^2 (4m\omega_2^2\tau^2 - 2\tau(1+2m))}_{\Delta(\tau)} .s^2 \right) \|x_{h,k} - x_{h,\star}\|^2,\end{aligned}$$

which completes the proof of part (a).
(b) In order to establish the contraction property we need to establish that there exists $s \in (0,1]$ and τ such that $0 < 2 + \Delta(\tau)s^2 < 1$, or equivalently

$$-2 < \Delta(\tau)s^2 < -1.$$

As $s \in (0,1]$, it is essential that $\Delta(\tau) < -1$. It follows from the definition of $\Delta(\tau)$ that we need to find a τ that satisfies,

$$A^2\tau^2 - B\tau + 1 < 0 \tag{37}$$

where

$$\begin{aligned} A^2 &= \frac{32}{9} m^2\omega_1^2\omega_2^2 s_{H,0}^2 \Rightarrow 2A = \frac{8\sqrt{2}}{3} m\omega_1\omega_2 s_{H,0} \\ B &= \frac{16}{9} m(1+2m)\omega_1^2 s_{H,0}^2. \end{aligned}$$

The definition of A implies that $2A = 8\sqrt{2}m\omega_1\omega_2 s_{H,0}/3$. Therefore inequality (37) can be written as,

$$(A\tau - 1)^2 - (B - 2A) < 0$$

The above inequality is always satisfied for $B > 2A$. Indeed, set $\tau = 1/A$ and use premise (b-i) in the statement of the Theorem then $2A$ is always less than B. Alternatively, if (b-i) is not true, but the number of coarse iterations m is sufficiently large, then we also have $B > 2A$. Once τ is defined such that $\Delta(\tau) < -1$, we can deduce,

$$\frac{1}{\sqrt{-\Delta(\tau)}} \leq s \leq \min\left\{ \frac{2}{\sqrt{-\Delta(\tau)}}, 1 \right\}.$$

□

The above theorem combined with the non-expansive/contraction property of gradient (mapping) proximal step and concludes the linear convergence properties of MISTA as stated in Theorem 1 and Theorem 2.

Remark. Assumption (b-i) in the statement of Theorem 3 is indeed satisfied by most common restriction/prolongation operators. For example, consider the two common restriction operators linear interpolation and coordinate wise restriction. For the linear interpolation operator I_h^H, assume that we have a fine vector $y_h \in \mathbb{R}^n$ and a coarse vector $y_H \in \mathbb{R}^{n/4}$. The operator I_h^H groups 4 fine dimension in one coarse dimension. In this case, the restriction matrix is given by,

$$I_h^H = \begin{bmatrix} 1\,1\,1\,1\,0\,\,0\,\,0\,0\ldots0\,0\,0\,0 \\ 0\,0\,0\,0\,1\,\,1\,\,1\,1\ldots0\,0\,0\,0 \\ \vdots \qquad \ddots \qquad \ddots \quad \vdots \\ 0\,0\,0\,0\,0\,\,0\,\,0\,0\ldots1\,1\,1\,1 \end{bmatrix}.$$

The prolongation operator is given by $I_H^h = \frac{1}{c}(I_h^H)^\top$, in this example $c = 4$. Clearly, we can set:

$$\omega_1 = \|I_h^H\| = \max_{y_h \neq 0} \frac{\|I_h^H y_h\|}{\|y_h\|} = \sqrt{c} \geq 1, \forall y_h \neq 0$$

as always, $c \geq 1$. On the other hand,

$$\omega_2 = \|I_H^h\| = \frac{1}{\sqrt{c}} \max_{y_H \neq 0} \frac{\|(I_h^H)^\top y_H\|}{\|y_H\|} = 1, \forall y_H \neq 0$$

For the coordinate wise operator (also known as an injection operator), assume that odd indices are omitted in the coarse vector. So, the restriction operator is defined as,

$$I_h^H = \begin{bmatrix} 0\,1\,\,0\,\,0\,\,0\,\,\ldots0 \\ 0\,0\,\,0\,\,1\,\,0\,\,\ldots0 \\ \vdots \quad \ddots \quad \ddots \quad \vdots \quad \vdots \\ 0\,0\,\,0\,\,0\,\,0\,\,\ldots1 \end{bmatrix}.$$

and the prolongation is simply given by $I_H^h = (I_h^H)^\top$. Then the upper bounds of ω_1, ω_2 are:

$$\omega_1 = \|I_h^H\| = \max_{y_h \neq 0} \frac{\|I_h^H y_h\|}{\|y_h\|} = 1 \quad \text{when } y_h(2i+1) = 0,\ i = 0, ..., h$$
$$\omega_2 = \|I_H^h\| = \max_{y_H \neq 0} \frac{\|(I_h^H)^\top y_H\|}{\|y_H\|} = 1, \forall y_H \neq 0.$$

In our numerical experiments both assumptions (b-i) and (b-ii) are always satisfied.

5 Numerical experiments

In this section we illustrate the numerical performance of the algorithm using the image restoration problem. We compare the CPU time required to achieve convergence of MISTA against ISTA and FISTA. We chose to report CPU times since the computational complexity of MISTA per iteration can be larger than ISTA or FISTA. We tested the algorithm on several images, and below we report results on a representative set of six images. All our test images have the

same size, 1024×1024. At this resolution, the optimization model at the fine scale has more than 10^6 variables (1048576, to be precise). We implemented the ISTA and FISTA algorithms with the same parameter settings as [2]. For the fine model we used the standard backtracking line strategy for ISTA as in [2]. All algorithms were implemented in MATLAB and run on a standard desktop PC. Due to space limitations, we only report detailed convergence results from the widely used cameraman image. The images we used, the source code for MISTA, and further numerical experiments can be obtained from the web-page of the first author `www.doc.ic.ac.uk/~pp500`.

5.1 Computation with the fine model

The image restoration problem consists of the following composite convex optimization model,

$$\min_{x_h \in \mathbb{R}^h} \|A_h x_h - b_h\|_2^2 + \mu_h \|W(x_h)\|_1,$$

where b_h is the vectorized version of the input image, A_h is the blurring operator based on the point spread function (PSF) and reflexive boundary conditions, and $W(x_h)$ is the wavelet transform of the image. The two dimensional version of the input image and the restored image are denoted by X_h and B_h respectively. The first term in the objective function aims to find an image that is as close to the original image as possible, and the second term enforces a relationship between the pixels and ensures that the recovered image is neither blurred nor noisy. The regularization parameter μ_h is used to balance the two objectives. In our implementation of the fine model we used $\mu_h = 10e-4$. Note that the first term is convex and differentiable, the second term is also convex but non-smooth. The blurring operator A_h, is computed by utilizing an efficient implementation provided in the HNO package [14]. In particular, we rewrite the expensive matrix computation $A_h x_h - b_h$ in the reduced form,

$$A_h^c X_h (A_h^r)^\top - B_h,$$

where A_h^c, A_h^r are the row/column blurring operators and $A_h = A_h^r \otimes A_h^c$. We illustrate the problem of image restoration using the widely used cameraman image. Figure 2(a) is the corrupted image, and the restored image is shown in Figure 2(b). The restored image was computed with MISTA. The image restoration problem fits exactly the framework of convex composite optimization. In addition it is easy to define a hierarchy of models by varying the resolution of the image. We discuss the issue of coarse model construction next.

5.2 Construction and computation with the coarse model

We described MISTA as a two level algorithm but it is easy to generalize it to many levels. In our computations we used the fine model described above and two coarse models, one with resolution 512×512 and its coarse version i.e. a model with 256×256. Each model on the hierarchy has a quarter of the variables of the model above it. We used the smoothing approach to construct the coarse models (see

(a) Corrupted image with 0.5% noise (b) Restored image

Fig. 2 (a) Corrupted cameraman image used as the input vector b, (b) Restored image.

Section 3.2.1). Following the smoothing approach we used the following objective function,

$$\min_{x_H \in \Omega_H} \|A_H x_H - b_H\|_2^2 + \langle v_H, x_H \rangle + \mu_H \sum_{i \in \Omega_H} \sqrt{W(x_H)_i^2 + \rho^2} - \rho$$

where $\rho = 0.2$ is the smoothing parameter, v_H was defined in (9), and λ_H is the regularizing parameter for the coarse model. Since the coarse model has less dimensions, the coarse problem is smoother, therefore the regularizing parameter should be reduced, we used $\mu_H = \mu_h/2$. The information transfer between levels is done via a simple linear interpolation technique to group four fine pixels into one coarse pixel. This is a standard way to construct the restriction and prolongation operators and we refer the reader to [9] for the details. The input image, and the current iterate are restricted to the coarse scale as follows,

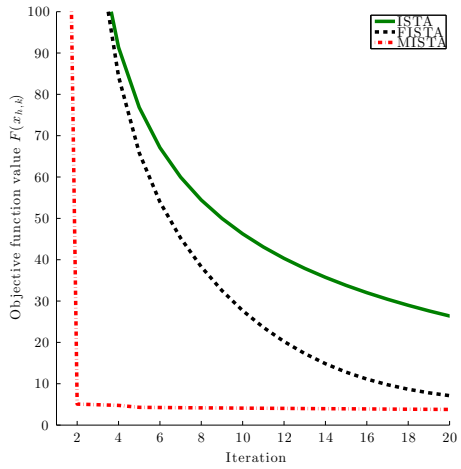
$$x_{H,0} = I_h^H x_{h,k} \quad , \quad b_H = I_h^H b_h.$$

The standard matrix restriction $A_H = I_h^H A_h (I_h^H)^\top$ is not performed explicitly as we never need to store the large matrix A_h. Instead, only column and row operators A_h^c, A_h^r are stored in memory. As a decomposition of the restriction operator is available for our problem, in particular $I_h^H = R_1 \otimes R_2$, we can obtain the coarse blurring matrix by,

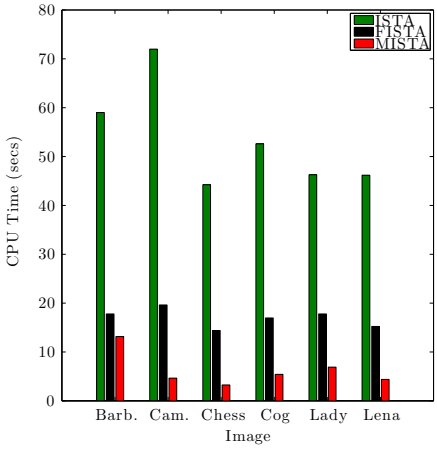
$$A_H = A_H^r \otimes A_H^c$$

where $A_H^c = R_2 A_h^c R_1^\top$ and $A_H^r = R_1 A_h^r R_2^\top$.

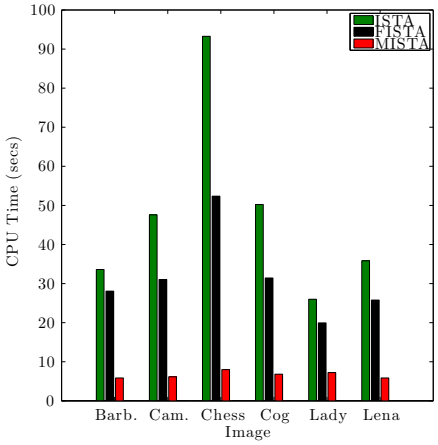
The condition to use the coarse model in MISTA is specified in (19), and we used the parameters $\kappa = 0.5$ and $\eta = 1$ in our implementation. Since at the coarse scale the problem is smooth ISTA reduces to the standard steepest descent algorithm. In our implementation we used the steepest descent algorithm with an Armijo line search.



(a) Function value comparison



(b) Images blurred with 0.5% noise



(c) Images blurred with 1% noise

Fig. 3 (a) Comparison of the three algorithms in terms of function value. MISTA clearly outperforms the other algorithms and converges in essentially 5 iterations, while others have not converged even after 100 iterations. CPU time required to find a solution within 2% of the optimum for the three algorithms. (b) Results for blurred images with 0.5% noise (c) Results for blurred images with 1% noise. Higher levels of noise lead to more ill conditioned problems. The figures in (b) and (c) compare CPU times and suggest that MISTA is on average ten times faster than ISTA and three/four times than FISTA.

5.3 Performance comparison

We compare the performance of our methods with FISTA and ISTA using a representative set of corrupted images (blurred with 0.5% additive noise). In Figure 3(a) we compare the three algorithms in terms of the progress they make in func-

tion value reduction. In this case we see that MISTA clearly outperform ISTA. This result is not surprising since MISTA is a more specialized algorithm with the same convergence properties. However, what is surprising is that MISTA still outperforms the theoretically superior FISTA Clearly, MISTA outperforms FISTA in early iterations and is comparable in latter iterations.

Figure 3 gives some idea of the performance of the algorithm but of course what matters most is the CPU time required to compute a solution. This is because an iteration of MISTA requires many iterations in a coarse model, and therefore comparing the algorithms in terms of the number of iterations is not fair. In order to level the playing field, we compare the performance of the algorithms in terms of CPU time required to find a solution that satisfies the optimality conditions within 2%. Two experiments were performed on a set of six images. The first experiment takes as input a blurred image with 0.5% additive Gaussian noise and the second experiment uses 1% additive noise. We expect the problems with the 1% additive noise to be more difficult to solve than the one with 0.5% noise. This is because the corrupted image is more ill-conditioned. Figure 3(b) shows the performance of the three algorithms on blurred images with 0.5% noise. We can see that MISTA outperforms both ISTA and FISTA by some margin. On average MISTA is four times faster than FISTA and ten times faster than ISTA. In Figure 3(c), we see an even greater improvement of MISTA over ISTA/FISTA. This is expected since the problem is more ill-conditioned (with 1% noise as opposed to 0.5% noise in Figure 3(c)), and so the fine model requires more iterations to converge. Since ISTA/FISTA perform all their computation with the ill conditioned model, CPU time increases as the amount of noise in the image increases. On the other hand, the convergence of MISTA depends less on how ill conditioned the model is since one of the effects of averaging is to decrease ill conditioning.

6 Conclusions

We developed a multilevel algorithm for composite convex optimization models (MISTA). The key idea behind MISTA is, for some iterations, to replace the quadratic approximation with a coarse approximation. The coarse model is used to compute search directions that are often superior to the search directions obtained using just gradient information. We showed how to construct coarse models in the case where the objective function is non-differentiable. We also discussed several ways to enforce the first order coherency condition for composite optimization models. We developed the multilevel algorithm based on ISTA and established its linear rate of convergence. Our initial numerical experiments show that the proposed MISTA algorithm is on average ten times faster than ISTA, and three-four times faster (on average) than the theoretically superior FISTA algorithm.

The initial numerical results are promising but still the algorithm can be improved in a number of ways. For example, we only considered the most basic prolongation and restriction operators in approximating the coarse model. The literature on the construction of these operators is quite large and there exist more advanced operators that adapt to the problem data and current solution (e.g. bootstrap AMG [8]). We expect that the numerical performance of the algorithm can be improved if these advanced techniques are used instead of the naive approach proposed here. We based our algorithm on ISTA due to its simplicity. It

is of course desirable to develop a multilevel version of FISTA, and in the process establish a better rate of convergence for MISTA. In the last few years several algorithmic frameworks for large scale composite convex optimization have been proposed. Examples include active set methods[18], stochastic methods [16], Newton type methods [17] as well as block coordinate descent methods [27]. In principle all these algorithmic ideas could be combined with the multilevel framework developed in this paper. Based on the theoretical and numerical results obtained from the multilevel version of ISTA we are hopeful that the multilevel framework can improve the numerical performance of many of the recent algorithmic developments in large scale composite convex optimization.

References

1. A. Beck and S. Sabach. A first order method for finding minimal norm-like solutions of convex optimization problems. Mathematical Programming, pages 1–22, 2013.
2. A. Beck and M. Teboulle. A fast iterative shrinkage-thresholding algorithm for linear inverse problems. SIAM Journal on Imaging Sciences, 2(1):183–202, 2009.
3. A. Beck and M. Teboulle. Smoothing and first order methods: A unified framework. SIAM Journal on Optimization, 22, 2012.
4. D.P. Bertsekas. Nonlinear Programming. Optimization and Computation Series. Athena Scientific, 1999.
5. D.P. Bertsekas. Nonlinear programming. Athena Scientific, 2004.
6. A. Borzì. On the convergence of the mg/opt method. PAMM, 5(1):735–736, 2005.
7. A. Borzì and V. Schulz. Multigrid methods for pde optimization. SIAM review, 51(2):361–395, 2009.
8. A. Brandt, J. Brannick, K. Kahl, and I. Livshits. Bootstrap amg. SIAM Journal on Scientific Computing, 33, 2011.
9. W.L. Briggs, V.E Henson, and S.F McCormick. A multigrid tutorial. Society for Industrial and Applied Mathematics (SIAM), Philadelphia, PA, second edition, 2000.
10. 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 & Software, 25(3):359–386, 2010.
11. S. Gratton, M. Mouffe, P.L Toint, and M. Weber-Mendonça. A recursive-trust-region method for bound-constrained nonlinear optimization. IMA Journal of Numerical Analysis, 28(4):827–861, 2008.
12. S. Gratton, A. Sartenae, and P.L Toint. Recursive trust-region methods for multiscale nonlinear optimization. SIAM Journal on Optimization, 19(1):414–444, 2008.
13. E. Haber and J. Modersitzki. A multilevel method for image registration. SIAM Journal on Scientific Computing, 27(5):1594–1607, 2006.
14. P.C Hansen, J.G Nagy, and D.P O'leary. Deblurring images: matrices, spectra, and filtering, volume 3. Siam, 2006.
15. N. Komodakis. Towards more efficient and effective lp-based algorithms for mrf optimization. In Computer Vision–ECCV 2010, pages 520–534. Springer, 2010.
16. G. Lan. An optimal method for stochastic composite optimization. Mathematical Programming, 133(1-2):365–397, 2012.
17. J. D Lee, Y. Sun, and M.A Saunders. Proximal newton-type methods for minimizing composite functions. arXiv preprint arXiv:1206.1623, 2014.
18. A.S Lewis and S.J Wright. A proximal method for composite minimization. arXiv preprint arXiv:0812.0423, 2008.
19. 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, 2005.
20. S. G. Nash. A multigrid approach to discretized optimization problems. Optimization Methods and Software, 14(1-2):99–116, 2000.
21. S.G Nash. Properties of a class of multilevel optimization algorithms for equality-constrained problems. Optimization Methods and Software, 29, 2014.

22. S.G. Nash and R.M Lewis. Assessing the performance of an optimization-based multilevel method. Optimization Methods and Software, 26(4-5):693–717, 2011.
23. Y. Nesterov. Introductory Lectures on Convex Optimization. Kluwer, 2004.
24. Y. Nesterov. Gradient methods for minimizing composite objective function. Mathematical Programming, 140(1):125–161, 2013.
25. J. Nocedal and S.J. Wright. Numerical Optimization. Springer Series in Operations Research. Springer-Verlag, 2006.
26. P. Parpas and M. Webster. A stochastic multiscale model for electricity generation capacity expansion. European Journal of Operational Research, 232(2):359 – 374, 2014.
27. P. Richtárik and M. Takáč. Iteration complexity of randomized block-coordinate descent methods for minimizing a composite function. Mathematical Programming, 144(1-2):1–38, 2014.
28. R.T Rockafellar. Monotone operators and the proximal point algorithm. SIAM Journal on Control and Optimization, 14(5):877–898, 1976.
29. S. Sra, S. Nowozin, and S. J. Wright. Optimization for Machine Learning. Neural Information Processing Series. The MIT Press, 2012.
30. J.J Thiagarajan, K. N. Ramamurthy, and A. Spanias. Learning stable multilevel dictionaries for sparse representation of images. IEEE Trans. on Neural Networks and Learning Systems (Under review), 2013.
31. Z. Wen and D. Goldfarb. A line search multigrid method for large-scale nonlinear optimization. SIAM Journal on Optimization, 20(3):1478–1503, 2009.