

Complexity iteration analysis for strongly convex multi-objective optimization using a Newton path-following procedure

E. Bergou * Y. Diouane[†] V. Kungurtsev [‡]

April 6, 2020

Abstract

In this note we consider the iteration complexity of solving strongly convex multi objective optimization. We discuss the precise meaning of this problem, and indicate it is loosely defined, but the most natural notion is to find a set of Pareto optimal points across a grid of scalarized problems. We derive that in most cases, performing sensitivity based path-following after obtaining one solution is the optimal strategy for this task in terms of iteration complexity.

Keywords: Multi-objective optimization; strongly convex optimization; path-following; Newton method; Complexity iteration analysis.

1 Introduction

Consider the following multi-objective optimization problem:

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

where $f : \mathbb{R}^n \rightarrow \mathbb{R}^m$ is a strongly convex and twice continuously differentiable function. Our target is to find *weak Pareto-optimality* points for the problem (1), recalling that *weak Pareto-optimality* holds at point \tilde{x} if for all $d \in \mathbb{R}^n$, there exists an $i \in \{1, \dots, m\}$ such that

$$\nabla f_i(\tilde{x})^\top d \geq 0.$$

For single objective optimization, *worst case iteration complexity* quantifies the number of iterations that could be necessary, in the worst case (i.e., for the most ill-behaved problem), for

*King Abdullah University of Science and Technology (KAUST), Thuwal, Saudi Arabia. MaIAGE, INRAE, Université Paris-Saclay, 78350 Jouy-en-Josas, France (elhoucine.bergou@inra.fr). This author received support from the AgreeSkills+ fellowship programme which has received funding from the EU's Seventh Framework Programme under grant agreement No FP7-609398 (AgreeSkills+ contract).

[†]Institut Supérieur de l'Aéronautique et de l'Espace (ISAE-SUPAERO), Université de Toulouse, 31055 Toulouse Cedex 4, France (youssef.diouane@isae.fr).

[‡]Department of Computer Science, Faculty of Electrical Engineering, Czech Technical University in Prague. (vyacheslav.kungurtsev@fel.cvut.cz). Research supported by the OP VVV project CZ.02.1.01/0.0/0.0/16_019/0000765 Research Center for Informatics.

an algorithm to achieve a certain level of satisfaction of an approximate measure of optimality, typically a small norm for the gradient [10]. Classically, the multi objective optimization community had not considered attempting to derive bounds on iteration complexity for problems in vector optimization. In [7, 4] the complexity of gradient descent for multi-objective optimization was considered. Rates were derived for obtaining some point satisfying weak Pareto-optimality. However, in deriving the complexity result, convergence of the algorithm to some Pareto optimal point is assumed. More fundamentally, though, consider the so-called scalarized problem, parametrized by $\{\lambda_i\}_{i=1,\dots,m}$

$$\min_{x \in \mathbb{R}^n} \sum_{i=1}^m \lambda_i f_i(x), \quad (2)$$

for any $\{\lambda_i\}_{i=1,\dots,m}$ satisfying $0 < \lambda_i < 1$, $\sum_i \lambda_i = 1$. A stationary point of this problem is also Pareto optimal for (1). Thus, one can find a Pareto optimal point, at least for strongly convex multi-objective problems, by simply choosing any arbitrary convex combination $\{\lambda_i\}_{i=1,\dots,m}$ and solving the resulting mono-objective problem, thus the worst case iteration complexity of finding *some* Pareto optimal point is already a known problem, it corresponds to the worst case iteration complexity of solving a single objective strongly convex optimization problem.

In the multi-objective optimization literature, e.g., [9], scalarization is typically, at most, a step in the process of finding the solution of a multi-objective problem, where the definition of solution depends on the context. In particular, it can be that the goal of the optimization is 1) tracing the Pareto front itself, so in some sense finding all, or some adequate approximation to all, stationary points, or 2) find an appropriately *best* point of the Pareto front through some secondary metrics, or using an interactive environment with a human participant who grades potential solutions.

In this note, we shall concern ourselves with the first task: establish complexity bounds for some appropriate notion of finding the entire Pareto front. To this end we define the problem as, for all $\lambda \in \Lambda \subset \mathbb{R}^m$, find

$$\min_{x \in \mathbb{R}^n} \sum_{i=1}^m \lambda_i f_i(x), \quad (3)$$

where Λ is some finite grid of elements λ satisfying $0 < \lambda_i < 1$, $\sum_i \lambda_i = 1$. Given the constraint on the sum, we can consider $m - 1$ dimensions as free which in turn entirely determine the remaining λ_i . We thus divide each side of the hypercube $[0, 1]$ by some desired width of the grid d , and thus there are $\lfloor \frac{1}{d} \rfloor^{m-1}$ total possible grid points, where $\lfloor a \rfloor$ denotes the greatest integer less than or equal to a . Conversely, since they form a rectangular lattice, we can define the quantity d denoting the maximum width of any two neighbors on the grid, that is

$$d = \min_{(\lambda, \lambda') \in \Lambda, \lambda \neq \lambda'} \left(\max_{1 \leq i \leq m} |\lambda_i - \lambda'_i| \right).$$

We organize this paper as follows. In section 2, we describe our algorithm to solve the multi objective strongly convex problem. We explain how to use a Newton path following procedure to find the entire Pareto front. Section 3 addresses the convergence of our algorithm by characterizing its Complexity. A numerical illustration about the efficiency of our approach is given in section 4. Conclusions are given in section 5.

2 Pathfollowing for finding the entire Pareto front.

Recall that for a fixed $\lambda^0 \in \Lambda$, using only first order information one can solve a strongly convex optimization problem of the type (3) at best linearly (see for instance [3, Theorem 3.18], using a gradient descent based method). Namely, in order to obtain ϵ distance to the solution of (3) for a fixed $\lambda^0 \in \Lambda$, $\mathcal{O}(\log(1/\epsilon))$ iterations must be taken, with each iteration involving the computation of one gradient. As a result, naively, one can obtain the entire Pareto front by solving each of the $\lfloor \frac{1}{d} \rfloor^{m-1}$ scalarized problems defined across the grid points independently with a gradient descent, to obtain an overall complexity of $\mathcal{O}\left(\log(1/\epsilon) \lfloor \frac{1}{d} \rfloor^{m-1}\right)$.

In this note, we will propose finding the entire Pareto front by performing path-following using the implicit function theorem. Later we will show that the proposed strategy will reduce the overall iteration complexity drastically relative to naively solving every scalarized problem separately. To start with, for $\lambda^{(0)} \in \Lambda$ we obtain the solution $x^{(0)} \in \mathbb{R}^n$ of the problem (3), (for instance, by using a gradient descent method using the following stopping criterion $\left\| \sum_{j=1}^m \lambda_j^{(0)} \nabla f_i(x) \right\| \leq \epsilon$). Note that such point $x^{(0)}$ gives the Pareto optimal point of the problem (1) associated to $\lambda^{(0)}$. Now, let $\lambda^{(1)}$ be one of the closest neighbors of $\lambda^{(0)}$ in the finite grid Λ , our goal is to apply a predictor-corrector scheme to deduce a new $x^{(1)}$ corresponding to the Pareto optimal solution of the problem (1) associated with $\lambda^{(1)}$.

Pathfollowing, or tracing a set of solutions for a parametrized nonlinear system of equations across a range of parameters, is an important algorithmic tool, for which an introduction can be found in [1]. Closest to our work, a predictor-corrector pathfollowing procedure for strongly convex optimization problems (interpreted as strongly regular variational inequalities) is given in [6]. In this work it is shown that for this parametric problem a property of uniform strong regularity holds and a procedure involving one tangential predictor (Euler) and one corrector (Newton) step result in a series of iterates whose distance to a set of solutions to the parametric variational inequality is of the order of d^4 , where d is, in this context, the grid spacing. Thus there exists C such that if $d \leq C(\epsilon)^{1/4}$, a set of solutions with approximate optimality ϵ across a set of parameters can be found. If applied to the multiobjective Pareto front context, the number of Euler-Newton continuation steps is the number of grid points, which corresponds to $d^{-1} \geq C^{-1}\epsilon^{-1/4}$. If the desired grid is already small enough, then it is clear that this pathfollowing procedure outperforms the naive method of solving the standalone problem at every grid point. Otherwise, it depends on the magnitude of the desired number of additional grid points required to perform pathfollowing.

We consider an alternative predictor-corrector scheme that is more aggressive in its use of potentially longer tangential steps and multiple Newton iterations. In particular, this is more suitable for obtaining the set of solutions across the Pareto front with the tightest iteration complexity bound. This predictor-corrector procedure will be repeated until we handle all the elements from Λ . A formal description of the algorithm is given as Algorithm 1.

The predictor step (Step 1 of Algorithm 1) is formulated using the implicit function theorem. Therefore, first, let's recall the implicit function theorem adapted to our context.

Theorem 2.1 *Let $g : \mathbb{R}^{n+m} \rightarrow \mathbb{R}^n$ be a continuously differentiable function for a parametrized system of equations,*

$$g(x, \lambda) = 0, \quad \text{where } x \in \mathbb{R}^n \text{ and } \lambda \in \mathbb{R}^m.$$

Consider that there exists a solution satisfying $g(x_0, \lambda_0) = 0$. If the Jacobian matrix $J_{g,x}(x_0, \lambda_0)$

Algorithm 1: A Newton path-following procedure for the entire Pareto front.

Input: Let $\Lambda = \{\lambda^{(0)}, \dots, \lambda^{(p-1)}\} \subset \mathbb{R}_+^m$ be some finite grid of p elements satisfying, for all $j = 0, \dots, p-1$, $\sum_i^m \lambda_i^{(j)} = 1$, $\lambda^{(j+1)}$ is one of the closest neighbors not yet visited of $\lambda^{(j)}$.

Output: The entire Pareto front by performing path-following associated with Λ : $x^{(0)}, x^{(1)}, \dots, x^{(p-1)}$.

Compute an initial Pareto optimal point $x^{(0)}$, i.e.,

$$x^{(0)} = \arg \min_x f^{(0)}(x), \text{ where } f^{(0)}(x) = \sum_{i=1}^m \lambda_i^{(0)} f_i(x). \quad (4)$$

Set $k = 0$.

Step 1: Compute a predictor $\bar{x}^{(k+1)}$, i.e.,

$$\bar{x}^{(k+1)} = x^{(k)} + \left[\sum_{j=1}^m \lambda_j^{(k)} \nabla^2 f_j \left(x^{(k)} \right) \right]^{-1} \left(\sum_{i=1}^m \left(\lambda_i^{(k+1)} - \lambda_i^{(k)} \right) \nabla f_i \left(x^{(k)} \right) \right) \quad (5)$$

Step 2: Apply a Newton correction to compute $x^{(k+1)}$, i.e., starting from $\bar{x}^{(k+1)}$ run the Newton method to solve

$$x^{(k+1)} = \arg \min_x f^{(k)}(x), \text{ where } f^{(k)}(x) = \sum_{i=1}^m \lambda_i^{(k+1)} f_i(x). \quad (6)$$

If $k = p - 1$ **then Stop, otherwise** increment k by 1 and go to **Step 1**.

of g with respect to x is invertible, then there exists an open neighborhood $\mathcal{B} \subset \mathbb{R}^m$ such that there exists a unique continuously differentiable path $\tilde{x}(\lambda)$ defined on $\lambda \in \mathcal{B}$ with $\tilde{x}(\lambda_0) = x_0$ and $g(\tilde{x}(\lambda), \lambda) = 0$ for all $\lambda \in \mathcal{B}$. Furthermore, it holds that the derivative of $\tilde{x}(\lambda)$ over \mathcal{B} is given by

$$\frac{\partial \tilde{x}}{\partial \lambda}(\lambda) = -[J_{g,x}(\tilde{x}(\lambda), \lambda)]^{-1} \frac{\partial g}{\partial \lambda}(\tilde{x}(\lambda), \lambda). \quad (7)$$

We consider applying Theorem 2.1 to the optimality conditions of (2) given by the following parametrized system of equations

$$g(x, \lambda) = \sum_{i=1}^m \lambda_i \nabla f_i(x) = 0.$$

Precisely, for a given iteration index k , consider that we have a solution $x^{(k)}$ at some $\lambda^{(k)} \in \Lambda$, i.e.,

$$\sum_{j=1}^m \lambda_j^{(k)} \nabla f_j(x^{(k)}) = 0. \quad (8)$$

Since we consider strongly convex objectives, it holds that the matrix

$$\sum_{i=1}^m \lambda_i^{(k)} \nabla^2 f_i(x)$$

is invertible for all $x \in \mathbb{R}^n$, in particular the inverse norm is bounded by the inverse of the weighted sum of the strong convexity constants of the problem. Thus by Theorem 2.1 we have that there exists a unique path $\tilde{x}^{(k)}(\lambda)$ for any choice of $\lambda \in \mathcal{B}_k$ such that $\tilde{x}^{(k)}(\lambda^{(k)}) = x^{(k)}$ and $\sum_{i=1}^m \lambda_i \nabla f_i(\tilde{x}^{(k)}(\lambda)) = 0$, for some ball $\mathcal{B}^{(k)}$ around $\lambda^{(k)}$. Furthermore, the derivative of the path given by (7) is defined for all $\lambda \in \mathcal{B}^{(k)}$ to satisfy,

$$\frac{\partial \tilde{x}^{(k)}}{\partial \lambda}(\lambda) = - \left[\sum_{j=1}^m \lambda_j \nabla^2 f_j \left(\tilde{x}^{(k)}(\lambda) \right) \right]^{-1} \left[\nabla f_1 \left(\tilde{x}^{(k)}(\lambda) \right), \dots, \nabla f_m \left(\tilde{x}^{(k)}(\lambda) \right) \right].$$

Consider now a Taylor expansion of $\tilde{x}^{(k)}(\lambda)$ for all $\lambda \in \mathcal{B}^{(k)}$, this is given by

$$\tilde{x}^{(k)}(\lambda) = x^{(k)} - \left[\sum_{j=1}^m \lambda_j^{(k)} \nabla^2 f_j \left(x^{(k)} \right) \right]^{-1} \sum_{i=1}^m \left(\lambda_i - \lambda_i^{(k)} \right) \nabla f_i \left(x^{(k)} \right) + \mathcal{O} \left(\left\| \lambda - \lambda^{(k)} \right\|^2 \right) \quad (9)$$

Motivated by the discussion on Newton's method applied for path-following in [5, Chapter 5], for all $\lambda \in \mathcal{B}^{(k)}$, we define a predictor $\bar{x}^{(k)}(\lambda)$ by computing

$$\bar{x}^{(k)}(\lambda) = x^{(k)} - \left[\sum_{j=1}^m \lambda_j^{(k)} \nabla^2 f_j \left(x^{(k)} \right) \right]^{-1} \sum_{i=1}^m \left(\lambda_i - \lambda_i^{(k)} \right) \nabla f_i \left(x^{(k)} \right) \quad (10)$$

which is precisely the ‘‘tangent continuation method’’ with the order $p = 2$ as given in [5, Page 239]. We define the remainder term as $\eta^{(k)}$.

Assuming that $\lambda^{(k+1)}$ is close enough to $\lambda^{(k)}$ (i.e., $\lambda^{(k+1)} \in \mathcal{B}^{(k)}$), the predictor step $\bar{x}^{(k+1)}$ given by (5) in Algorithm 1 is defined as follows

$$\bar{x}^{(k+1)} = \bar{x}^{(k)} \left(\lambda^{k+1} \right).$$

There is a remaining algorithmic necessity before this becomes practical as the predictor $\bar{x}^{(k+1)}$ does not necessarily satisfy the desired level of stationarity. To achieve a point closer to the actual solution, we consider a ‘‘corrector’’ step $x^{(k+1)}$ by using a form of the Newton step. For a particular level of error, the ordinary Newton method is quadratically convergent towards the solution starting from the predicted point if the original point is sufficiently close. Thus we require that the predictor $\bar{x}^{(k+1)}$ is sufficiently accurate and determine the size of the step $\lambda^{(k+1)} - \lambda^{(k)}$ appropriately.

3 Characterizing the Complexity of Algorithm 1

Based on the ideas above, we can consider iteration complexity in a new sense. For a given iteration k , consider having a point solution for a particular $\lambda^{(k)}$, up to an optimality tolerance with a desired ϵ . Then consider path-following from $\lambda^{(k)}$ to some $\lambda^{(k+1)}$ where $\lambda^{(k+1)} - \lambda^{(k)}$ is small enough (in terms of desired grid-spacing d) to be able to determine the associated solution on the Pareto front. The same procedure is repeated across all the grid Λ until all solutions of the Pareto front have been found.

Before developing our complexity analysis, we state formally our working assumptions on the objective function f .

Assumption 3.1 *The objective function $f : \mathbb{R}^n \rightarrow \mathbb{R}^m$ is twice continuously differentiable and strongly convex. In particular, there exists two positive constant $c > 0$ and $L > 0$, for all $i \in \{1, \dots, m\}$ and $y \in \mathbb{R}^n$, it holds that,*

$$c\|y\|^2 \leq y^\top \nabla^2 f_i(x)y \leq L\|y\|^2 \quad (11)$$

for all $x \in \mathbb{R}^n$. In other words, for all $i \in \{1, \dots, m\}$, the eigenvalues of the Hessian of f_i are uniformly bounded from below by c , and above by L , everywhere.

This implies the following condition regarding scaling invariance properties appropriate for Newton methods [5].

Lemma 3.1 *For all $\lambda \in \Lambda$, the mapping $x \rightarrow \sum_{i=1}^p \lambda_i \nabla^2 f_i(x)$ is affine covariant Lipschitz, i.e., meaning that there exists $\omega > 0$ such that for all $x, y \in \mathbb{R}^n$, one has*

$$\left\| \left(\sum_{j=1}^m \lambda_j \nabla^2 f_j(x) \right)^{-1} \left(\sum_{j=1}^m \lambda_j \nabla f_j(y) - \sum_{j=1}^m \lambda_j \nabla f_j(x) \right) \right\| \leq \omega \|x - y\|. \quad (12)$$

Proof. Follows from Assumption 3.1 with $\omega = \frac{L}{c}$. ■

Lemma 3.2 *Consider Assumption 3.1. There exists some η depending only on properties of $\{f_k(\cdot)\}$ such that $\eta^{(k)} \leq \eta \|\lambda^{(k+1)} - \lambda^{(k)}\|$ for all k .*

Proof. Assumption 3.1 which is equivalent to strong regularity [11] implies Lipschitz continuity of the mapping $\tilde{x}(\lambda)$ introduced in Theorem 2.1 for all $\lambda^{(k)}$ by [6, Theorem 3.2].

Since strong regularity holds, we can apply [2, Theorem 5.60], noting that the problem is unconstrained and in that notation $G(x) \equiv 0$ and we let the perturbation $u(t) = \tilde{\lambda}^{(k)}(t)$ be $\tilde{\lambda}^{(k)}(t) = t\lambda^{(k+1)} + (1-t)\lambda^{(k)}$. The application of the Theorem implies that for $\|\lambda^{(k+1)} - \lambda^{(k)}\|$ sufficiently small it holds that there is a unique $\tilde{x}(\tilde{\lambda}^{(k)}(t))$ that is continuously differentiable. Since uniform regularity implies solution uniqueness, we can extend $\tilde{x}(\lambda)$ as needed across all $t\lambda^{(k+1)} + (1-t)\lambda^{(k)}$ for $t \in [0, 1]$. And since the grid is compact and again by uniform regularity, and there is a unique solution for all $\lambda^{(k)}$, it holds that this applies for all solution paths across the grid and $\tilde{x}(\lambda)$ is globally Lipschitz continuous.

Now we have shown there exists some C_L ,

$$\|\tilde{x}(\lambda^{(k+1)}) - \tilde{x}(\lambda^{(k)})\| \leq C_L \|\lambda^{(k+1)} - \lambda^{(k)}\|$$

and since,

$$\begin{aligned} \tilde{x}(\lambda^{(k+1)}) &= \tilde{x}(\lambda^{(k)}) + \frac{\partial \tilde{x}^{(k)}}{\partial \lambda}(\lambda^{(k)})^T (\lambda^{(k+1)} - \lambda^{(k)}) \\ &\quad + \frac{1}{2} (\lambda^{(k+1)} - \lambda^{(k)})^T \frac{\partial^2 \tilde{x}^{(k)}}{\partial \lambda^2}(\lambda^{(k)} + t\xi^{(k)}(\lambda^{(k+1)} - \lambda^{(k)})) (\lambda^{(k+1)} - \lambda^{(k)}) \end{aligned}$$

We have that,

$$\begin{aligned} \eta^{(k)} &= \left\| \frac{1}{2} (\lambda^{(k+1)} - \lambda^{(k)})^T \frac{\partial^2 \tilde{x}^{(k)}}{\partial \lambda^2}(\lambda^{(k)} + t\xi^{(k)}(\lambda^{(k+1)} - \lambda^{(k)})) (\lambda^{(k+1)} - \lambda^{(k)}) \right\| \\ &\leq \left\| \tilde{x}(\lambda^{(k+1)}) - \tilde{x}(\lambda^{(k)}) \right\| + \left\| \frac{\partial \tilde{x}^{(k)}}{\partial \lambda}(\lambda^{(k)})^T (\lambda^{(k+1)} - \lambda^{(k)}) \right\| \\ &\leq C_L \|\lambda^{(k+1)} - \lambda^{(k)}\| + \left\| \sup_{\lambda} \frac{\partial \tilde{x}^{(k)}}{\partial \lambda}(\lambda) \right\| \|\lambda^{(k+1)} - \lambda^{(k)}\| := \eta \|\lambda^{(k+1)} - \lambda^{(k)}\| \end{aligned}$$

■

The next result shows that applying the correction step will require a number of iterations of the ordinary Newton method of order $\log \log \left(\frac{1}{\epsilon}\right)$ to get an ϵ -Pareto optimal solution.

Lemma 3.3 *Let Assumption 3.1 hold. For a given iteration index k , consider $\lambda^{(k)} \in \Lambda$ and $x^{(k)}$ such that*

$$\left\| \sum_{j=1}^m \lambda_j^{(k)} \nabla f_i \left(x^{(k)} \right) \right\| \leq \epsilon.$$

Let $\lambda^{(k+1)} \in \Lambda$ such that

$$\left\| \lambda^{(k+1)} - \lambda^{(k)} \right\|_{\infty} \leq \frac{2}{\omega \eta}, \quad (13)$$

where η is as in Lemma 3.2 then, the ordinary Newton method with the starting point $\bar{x}^{(k+1)}$ (as given by (5)) converges and the computational cost of achieving a solution point $x^{(k+1)}$ such that

$$\left\| \sum_{j=1}^m \lambda_j^{(k+1)} \nabla f_i \left(x^{(k+1)} \right) \right\| \leq \epsilon.$$

is of order $\log \log \left(\frac{1}{\epsilon}\right)$.

Proof. From (13) and [5, Theorem 5.2], one conclude the ordinary Newton method with the starting point $\bar{x}^{(k+1)}$ converges towards a solution point $[\bar{x}^{(k+1)}]_{\infty}$ such that

$$\sum_{j=1}^m \lambda_j^{(k+1)} \nabla f_i \left([\bar{x}^{(k+1)}]_{\infty} \right) = 0.$$

Let $[\bar{x}^{(k+1)}]_j$ be the j^{th} iterate produced by an ordinary Newton method with the starting point $\bar{x}^{(k+1)}$. We note that by the definition of $\eta^{(k)}$ and using (13), one has

$$\left\| [\bar{x}^{(k+1)}]_0 - [\bar{x}^{(k+1)}]_{\infty} \right\| \leq \eta^{(k)} \leq \eta \left\| \lambda^{(k+1)} - \lambda^{(k)} \right\| \leq \frac{2}{\omega}. \quad (14)$$

In this case, using [5, Theorem 2.3], ordinary Newton method with the starting point $\bar{x}^{(k+1)}$ converges quadratically, i.e., for each iteration j of the ordinary Newton method, on has

$$\left\| [\bar{x}^{(k+1)}]_j - [\bar{x}^{(k+1)}]_{\infty} \right\| \leq \omega^{2^j} \left\| [\bar{x}^{(k+1)}]_0 - [\bar{x}^{(k+1)}]_{\infty} \right\|^{2^j}.$$

Hence, using (14), one deduces that

$$\left\| [\bar{x}^{(k+1)}]_j - [\bar{x}^{(k+1)}]_{\infty} \right\| \leq 2^{2^j}.$$

Thus,

$$\begin{aligned} \left\| \sum_{j=1}^m \lambda_j^{(k)} \nabla f_i \left([\bar{x}^{(k+1)}]_j \right) \right\| &= \left\| \sum_{j=1}^m \lambda_j^{(k)} \nabla f_i \left([\bar{x}^{(k+1)}]_j \right) - \sum_{j=1}^m \lambda_j^{(k)} \nabla f_i \left([\bar{x}^{(k+1)}]_{\infty} \right) \right\| \\ &\leq L \left\| [\bar{x}^{(k+1)}]_j - [\bar{x}^{(k+1)}]_{\infty} \right\| \leq L 2^{2^j}. \end{aligned}$$

This implies that the computational cost of achieving the desired level of stationarity is $O(\log \log(\frac{1}{\epsilon}))$. ■

Thus the complexity of Algorithm 1 is just of the order of complexity for solving a standalone strongly convex problem (i.e., computing $x^{(0)}$) added with $\frac{1}{d^{m-1}}$ multiplied by the cost of a predictor and a Newton step. We formalize this with the following theorem,

Theorem 3.1 *Let Assumption 3.1 hold. Define N_ϵ to be the number of iterations required to obtain $x^{(0)}$, a point that has distance at most ϵ from the optimal point corresponding to (2) at $\lambda^{(0)}$. Assume that the maximum width between any two neighbors on the grid Λ is,*

$$d \leq \min\left(\frac{2}{\omega\eta}, \bar{d}\right), \quad (15)$$

with \bar{d} the minimal desired distance between lattice points.

Then, the overall iteration complexity of Algorithm 1 is

$$N_\epsilon + \mathcal{O}\left(\left\lceil \frac{1}{\bar{d}} \right\rceil^{m-1} \log \log\left(\frac{1}{\epsilon}\right)\right).$$

Proof. First, note that, for each iteration k of Algorithm 1, the complexity of the predictor step is constant as its computational cost does not depend on ϵ . For the corrector Newton step, since one has

$$\|\lambda^{(k+1)} - \lambda^{(k)}\|_\infty \leq d \leq \frac{2}{\omega\eta},$$

Lemma 3.3 implies that complexity of such step is $O(\log \log(\frac{1}{\epsilon}))$. The proof is thus completed since the total number of lattice points in the grid Λ is at most $\lceil \frac{1}{\bar{d}} \rceil^{m-1}$. ■

Note that by using a gradient solver, the first term N_ϵ is of order $\log(1/\epsilon)$. Hence, one can see that the complexity is generally favorable compared to the naive method of solving the strongly convex problem at every grid point separately, as $\log \log(\frac{1}{\epsilon}) \ll \log(1/\epsilon)$ for small ϵ .

Remark 3.1 *Note that both the naive method of solving every problem across the grid points and path-following are both about equally parallelizable with perfect speedup as long as the number of grid points is larger than the number of processors. We can split the grid into disjoint components, and each processor finds one point in its part of the convex hull of allowable $\{\lambda_i\}$ and proceeds to pathfollow across the grid component assigned to it.*

4 Numerical Illustration

To show the numerical efficiency of our approach compared to the naive method (which corresponds to the Gradient Descent method applied sequentially to the set of problems (2) by varying λ), we will show the potential of the proposed approach on a very simple problem [8], defined by

$$f(x) = [(x_1 - 1)^2 + (x_1 - x_2)^2, (x_2 - 3)^2 + (x_1 - x_2)^2]^\top$$

Since we have two objective functions, the vector λ has two components λ_1 and λ_2 where $\lambda_1 + \lambda_2 = 1$. In our experiment, we discretize λ in a uniform grid with a grid stepsize d (the desired distance between the lattice points).

In our Matlab illustration, we will call **Multi-GD** the naive method and **GD+Newton Pathfollowing** the implementation of our Algorithm 1 (where we used Gradient Descent method to find the first Pareto optimal point and then apply the Newton path-following procedure). For the Gradient Descent method, we used a random initial point x_0 and a stepsize equal to $1/\lambda_{\max}$ where λ_{\max} is the maximum eigenvalue for the Hessians of f_1 and f_2 . We stopped the methods when the norm of the gradient is less than 10^{-7} .

The obtained results are depicted in Figure 1, one can see that both methods are able to find a similar Pareto Front (independently of the value of the grid stepsize d). In term of the elapsed CPU time to find the front, our proposed algorithm is shown to be faster than the naive method. In particular, one can see that, for different values of d , the method **GD+Newton Pathfollowing** runs 10 times faster than the **Multi-GD** method. We conducted other experiments (not reported here) on many toy problems, and in all the experiments, in term of running time our method was outperforming the naive method, while finding essentially the same front.

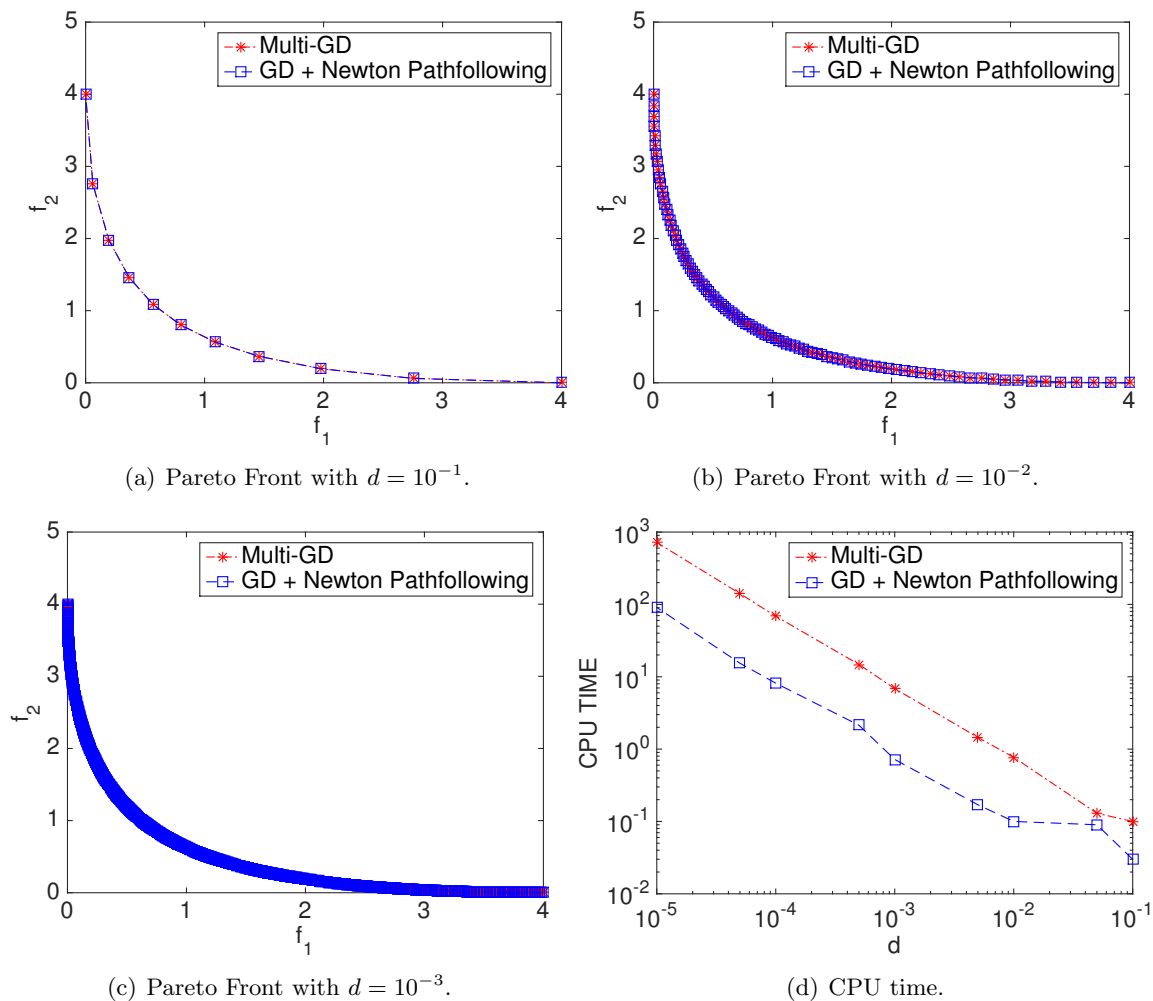


Figure 1: Pareto Front and CPU time comparison, using **Multi-GD** and **GD+ Newton Path-following**, for different values of d .

5 Conclusion

In this note we studied the complexity of a class of strongly convex multi objective optimization problems. We observed that such a problem is not uniquely defined, given different criteria of what it means to solve multiobjective optimization. Picking the most context-independent criterion – finding the set of all Pareto optimal points on a front, we demonstrated that in most cases, finding the solution of one scalarized problem and path-following across the grid to obtain the others is superior to finding the solution of every problem independently.

References

- [1] E. L. Allgower and K. Georg. *Introduction to numerical continuation methods*, volume 45. SIAM, 2003.
- [2] J. F. Bonnans and A. Shapiro. *Perturbation analysis of optimization problems*. Springer Science & Business Media, 2013.
- [3] S. Bubeck. Convex optimization: Algorithms and complexity. *Foundations and Trends® in Machine Learning*, 8:231–357, 2015.
- [4] L. Calderòn, M. A. Diniz-Ehrhardt, and J. M. Martinez. On high-order model regularization for multiobjective optimization. Technical report, 2018. In Optimization Online.
- [5] P. Deuffhard. *Newton methods for nonlinear problems: affine invariance and adaptive algorithms*, volume 35. Springer Science & Business Media, 2011.
- [6] A. L. Dontchev, M. I. Krastanov, R. T. Rockafellar, and V. M. Veliov. An euler–newton continuation method for tracking solution trajectories of parametric variational inequalities. *SIAM Journal on Control and Optimization*, 51(3):1823–1840, 2013.
- [7] J. Fliege, A. I. F. Vaz, and L. N. Vicente. Complexity of gradient descent for multiobjective optimization. *Optim. Methods Softw.*, 0:1–12, 2018.
- [8] S. Huband, P. Hingston, L. Barone, and L. While. A review of multiobjective test problems and a scalable test problem toolkit. *IEEE Transactions on Evolutionary Computation*, 10:477–506, 2006.
- [9] R. T. Marler and J. S. Arora. Survey of multi-objective optimization methods for engineering. *Structural and multidisciplinary optimization*, 26:369–395, 2004.
- [10] Y. Nesterov. *Introductory lectures on convex optimization: A basic course*, volume 87. Springer Science & Business Media, 2013.
- [11] S. M. Robinson. Strongly regular generalized equations. *Mathematics of Operations Research*, 5(1):43–62, 1980.