THE HONG KONG POLYTECHNIC UNIVERSITY 香港理工大學

A Proximal Gradient Method for Composite Multi-Objective Optimization

DEPARTMENT OF APPLIED MATHEMATICS 應 用 數 學 系

Yiyang Li (Supervisor: Prof. Xiaojun Chen)

Introduction

Modern machine learning models increasingly require balancing **conflicting objectives** such as accuracy, sparsity, and robustness. Formally, this challenge can be framed as a **composite multi-objective optimization problem**:

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

where $F : \mathbb{R}^n \to \mathbb{R}^m$ is defined as

$$F(x) := \begin{bmatrix} f_1(x) \\ f_2(x) \\ \vdots \\ f_m(x) \end{bmatrix} = \begin{bmatrix} h_1(x) \\ h_2(x) \\ \vdots \\ h_m(x) \end{bmatrix} + \begin{bmatrix} g_1(x) \\ g_2(x) \\ \vdots \\ g_m(x) \end{bmatrix} =: H(x) + G(x).$$

Each component $f_i = h_i + g_i$, where h_i is smooth but nonconvex and g_i is proper, lower semi-continuous, and convex but nonsmooth. Under a machine learning setting,

$$h_i(x) := \frac{1}{N}\sum_{j=1}^{N} h_{i,j}(x) \tag{2}$$

with each $h_{i,j}$ corresponds to one loss objective for one sample. The g_i's are the regularizers, for example ℓ_1-norm penalty for sparsity.

Contribution We propose a novel Conflict-Aware, Curvature-Informed Proximal Gradient (CACI-PG) method for composite multi-objective optimization. Our algorithm strategically integrates:

1. a conflict-aware gradient aggregation scheme,
2. an adaptive curvature-aware scaling to navigate complex landscapes,
3. and a proximal regularization step to effectively manage non-smooth convex regularizers.

Experimental Results

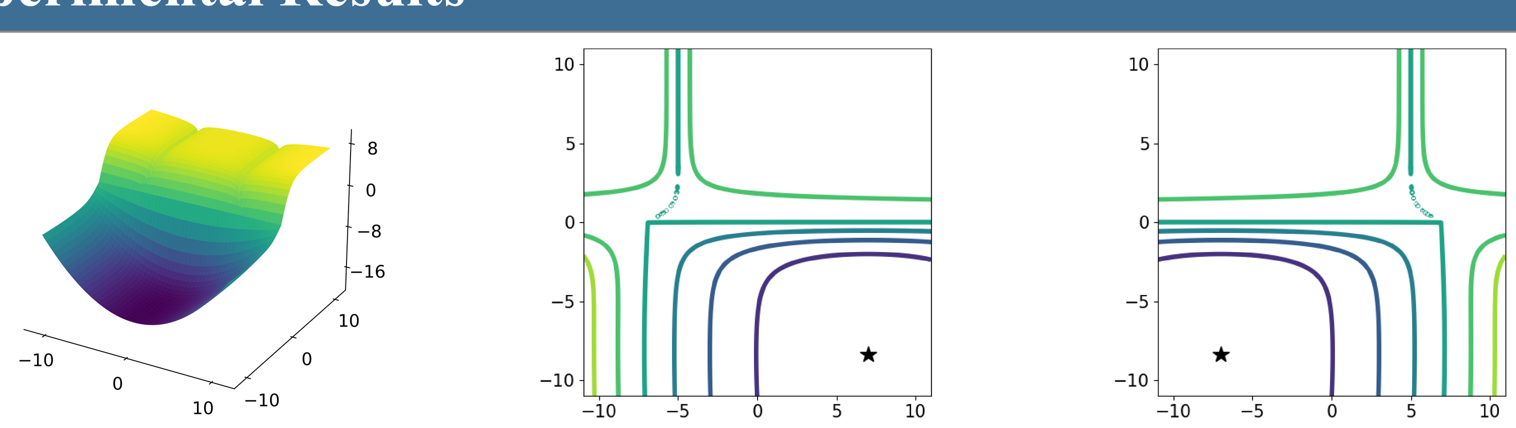

Figure: Overview of the Multi-Objective Model and each objective.

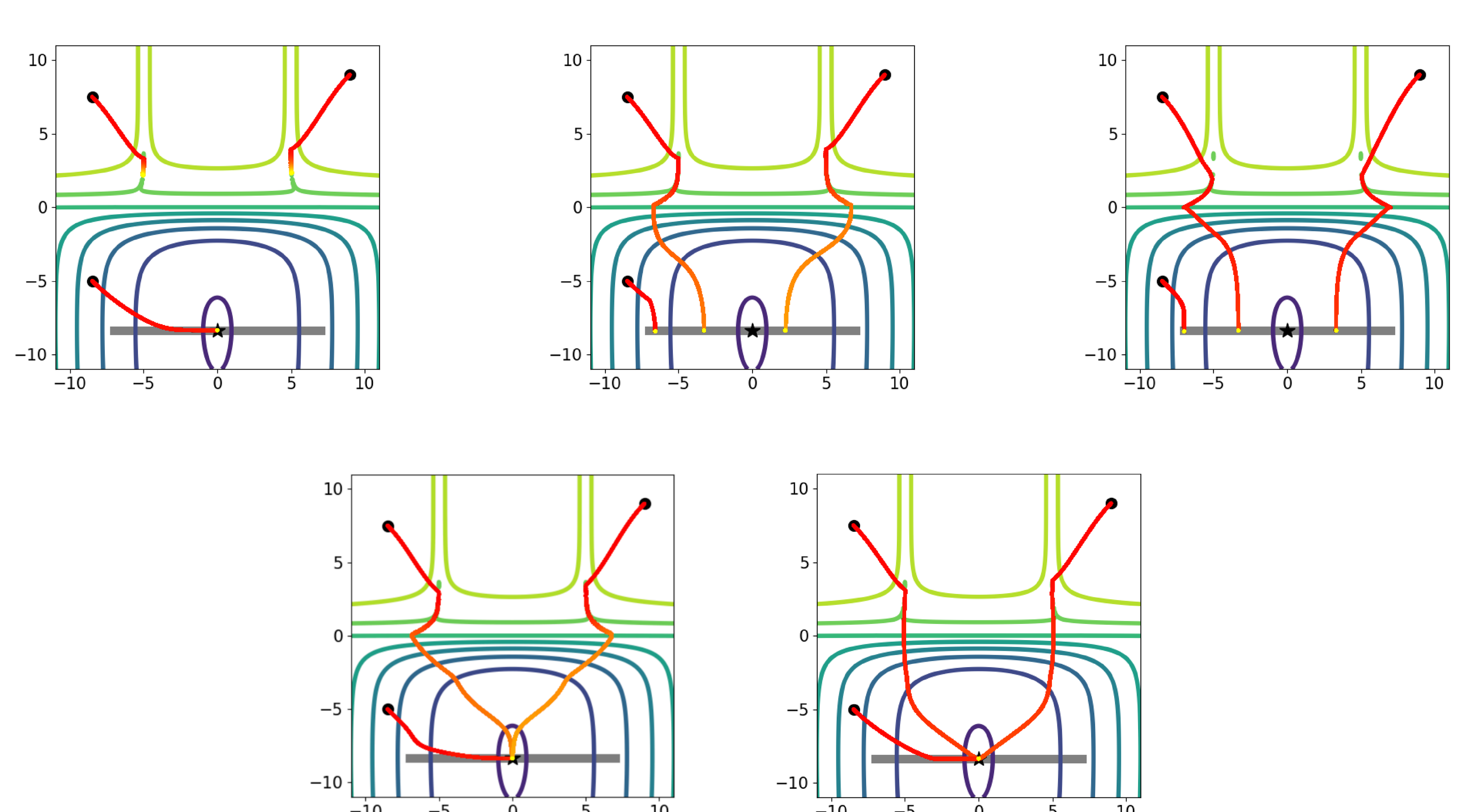

Figure: The plots show the results for SGD, PCGGrad, MGDA, CAGrad, and our method. The grey line represents the Pareto front; the star represents the Pareto point that averages the task objectives; and the dots represent the starting points. Each trajectory changes colour from red to yellow as the number of iterations increases.

References

- J.-A. Désidéri, *Multiple-gradient descent algorithm (MGDA) for multiobjective optimization*, Compt. Rend. Math., 350 (2012), pp. 313–318.
- H. Tanabe, E. Fukuda, and N. Yamashita, *Proximal gradient methods for multiobjective optimization and their applications*, Comput. Optim. Appl., 72 (2019), p. 339–361.

Existing Methods

When $G(x) \equiv 0$,

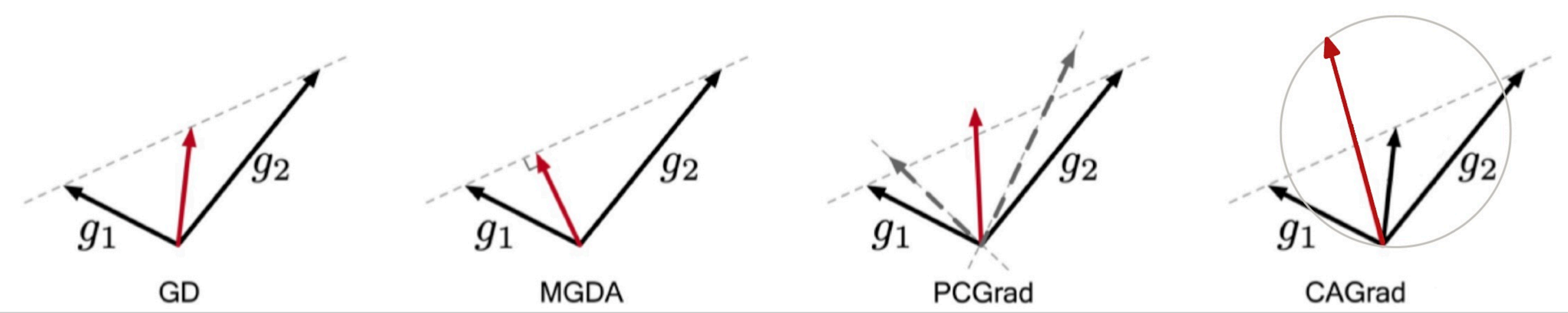

Figure: The update rules of d at iteration k for the above Bi-Objective Optimization Problem are given by the following different methods:
1) GD: $d_k = -\frac{\nabla h_1(x_k)+\nabla h_2(x_k)}{2}$;
2) MGDA [1]: $d_k \in \arg\min_{d\in\mathbb{R}^n} \max_{i\in\{1,2\}}\langle \nabla h_i(x_k), d\rangle \quad s.t. ||d|| \le 1$;
3) PCGrad: $d_k = -\frac{\nabla h_{1\perp 2}(x_k)+\nabla h_{2\perp 1}(x_k)}{2} \quad$ where $\nabla h_{i\perp j} = \nabla h_i - \frac{\nabla h_i^\top \nabla h_j}{\|\nabla h_j\|^2}\nabla h_j$;
4) CAGrad: $d_k \in \min_{d\in\mathbb{R}^n} \max_{i\in\{1,2\}}\langle \nabla h_i(x_k), d\rangle \quad s.t. \|d - \nabla h_0(x_k)\| \le c\|\nabla h_0(x_k)\|$, where $\nabla h_0(x_k) = \frac{1}{2}\sum_{i=1}^{2}\nabla h_i(x_k)$, and $c \ge 1$ to ensure the convergence to Pareto stationary point.

When $G(x) \not\equiv 0$, Tanabe [2] proposed a generalized Proximal Gradient Methods that updates d at iteration k by

$$d_k \in \arg\min_d \{ \max_{i\in\{1,\dots,m\}} [g_i(x_k+d) - g_i(x_k) + \langle \nabla h_i(x_k), d\rangle] + \frac{l}{2}\|d\|^2 \}.$$

Algorithm Framework of CACI-PG

Require: Initial point $x_0 \in \mathbb{R}^n$, step-size sequence $\{\lambda_k\}$, proximal parameter $\ell > 0$, trust-region radius $\{\Delta_k\}$, tolerance $\epsilon > 0$.

Set $k = 0$

while stopping criterion is not met **do**

Step 1: Compute task gradients $\{\nabla h_i(x_k)\}_{i=1}^m$.

Step 2: Compute curvature matrix

$$M_k = \frac{1}{m}\sum_{i=1}^{m} \nabla h_i(x_k)\nabla h_i(x_k)^\top.$$

Step 3: Compute descent direction

$$d_k = \arg\min_{d\in\mathcal{C}_M(x_k)} \left\{ \max_{i=1,\dots,m} [g_i(x_k+d) - g_i(x_k) + \langle \nabla h_i(x_k), d\rangle] + \frac{\ell}{2}\|d\|^2 \right\}$$

where $\mathcal{C}_M(x_k) := \{ d \in \mathbb{R}^n : \|d - \nabla h_0(x_k)\|_{M_k} \le \Delta_k \}$, with $\|x\|_M := \sqrt{x^T M x}$.

Step 4: Update iterate $x_{k+1} = x_k + \lambda_k d_k$.

if $\|d_k\| \le \epsilon$ **then**

break

else

$k = k + 1$

end if

end while

return $x = x_{k+1}$.

Define the function

$$\phi_i(d) := g_i(x_k+d) - g_i(x_k) + \langle \nabla h_i(x_k), d\rangle,$$

$$S(d) := \max_i \phi_i(d) + \frac{\ell}{2}\|d\|^2.$$

Introduce $t \in \mathbb{R}$ so that subproblem in step 3 is equivalent to

$$\begin{aligned} \min_{d,t} \quad & t + \frac{\ell}{2}\|d\|^2, \\ \text{s.t.} \quad & \phi_i(d) - t \le 0, \quad i = 1,\dots,m, \\ & \|d - \nabla h_0(x_k)\|_{M_k}^2 - \Delta_k^2 \le 0. \end{aligned} \tag{4}$$

We then calculate a dual form of the subproblem (4) to save computation.

Department of Applied Mathematics, Hong Kong Polytechnic University, yiyang-amber.li@connect.polyu.hk