

TRƯỜNG ĐẠI HỌC BÁCH KHOA HÀ NỘI
VIỆN TOÁN ỨNG DỤNG VÀ TIN HỌC

—o0o—



PHƯƠNG PHÁP HƯỚNG GIẢM SÂU NHẤT
TỰ THÍCH NGHI CHO HỌC ĐA NHIỆM

ĐỒ ÁN II

Chuyên ngành: TOÁN TIN

Giảng viên hướng dẫn: TS. TRẦN NGỌC THẮNG

Bộ môn: TOÁN TIN

Chữ kí của GVHD

Sinh viên thực hiện: NGUYỄN ANH MINH

Lớp: CTTN Toán Tin - K64

Hà Nội, 3/2023

Nhận xét của giảng viên	Điểm

Hà Nội, ngày ... tháng ... năm 2023

Giảng viên hướng dẫn

TS. Trần Ngọc Thăng

Lời mở đầu

Quy tắc viết đề án và ký hiệu

Trong báo cáo này, chúng tôi sử dụng một số tên viết tắt cho các thuật ngữ như sau:

MTL	Multi-task Learning
PMTL	Pareto Multi-task Learning
MGDA	Multi Gradient Descent Algorithm
MSGD	Multi Steepest Gradient Descent
MOP	Multi objective Optimization Problem

Bảng 1: Bảng thuật ngữ viết tắt

Đồng thời, chúng tôi sẽ sử dụng các ký hiệu sau trong báo cáo:

$\lambda \in \mathbb{R}^m$	vector trọng số
\mathbb{R}^m	không gian véc tơ m chiều
$y^1 < y^2$	$y_k^1 < y_k^2$ với $k = 1, \dots, m$
$y^1 \leq y^2$	$y_k^1 \leq y_k^2$ với $k = 1, \dots, m$ nhưng $y^1 \neq y^2$
\mathbb{R}_+^m	$\{y \in \mathbb{R}^m : y \geq 0\}$
\mathbb{R}_{++}^m	$\{y \in \mathbb{R}^p : y > 0\}$
\mathbb{R}_+	$\{y \in \mathbb{R} : y \geq 0\}$
\mathbb{R}_{++}	$\{y \in \mathbb{R} : y > 0\}$
i.i.d	Biến ngẫu nhiên độc lập có phân phối đồng nhất

Bảng 2: Bảng ký hiệu

Cấu trúc của đề án

Đề án II này sẽ được trình bày gồm 3 chương chính như sau:

- **Chương 1: Cơ sở lý thuyết** sẽ trình bày các kiến thức cơ bản liên quan đến tối ưu lồi và hướng giảm trong bài toán quy hoạch đa mục tiêu.
- **Chương 2: Thuật toán hướng giảm sâu nhất tự thích nghi** sẽ trình bày về thuật toán hướng giảm đề xuất để giải bài toán quy hoạch đa mục tiêu có ràng buộc cũng như không ràng buộc và áp dụng để giải bài toán MTL.
- **Chương 3: Tính toán thử nghiệm và đánh giá kết quả** sẽ trình bày về kết quả thực nghiệm của thuật toán đề xuất cho các bài toán tối ưu đa mục tiêu và kết quả thực nghiệm cho bài toán MTL với hai tác vụ phân loại ảnh với tập dữ liệu Multi-MNIST ([17]).

Dù đã rất cố gắng xong đề án này vẫn không tránh khỏi những hạn chế cần khắc phục. Vì vậy, chúng tôi rất mong quý thầy cô đưa ra những ý kiến góp ý bổ ích để đề án này tiếp tục được phát triển và có những kết quả mới tốt hơn.

Lời cảm ơn

Báo cáo này được thực hiện và hoàn thành tại Trường Đại học Bách Khoa Hà Nội, nằm trong nội dung học phần *Đồ án II* của kì học 2022-1.

Chúng tôi xin được dành lời cảm ơn chân thành tới TS. Trần Ngọc Thăng, là giảng viên đã trực tiếp hướng dẫn và gợi ý cho chúng tôi đề tài rất thú vị này, đồng thời thầy cũng đã giúp đỡ tận tình và có những đóng góp bổ ích để chúng tôi có thể hoàn thành báo cáo này một cách tốt nhất.

Hà Nội, ngày ... tháng 03 năm 2023.

Chúng tôi đồ án

Nguyễn Anh Minh

Mục lục

1	Cơ sở lý thuyết	7
1.1	Một số khái niệm và định nghĩa	7
1.2	Phát biểu bài toán	8
1.2.1	Bài toán quy hoạch đa mục tiêu	8
1.2.2	Bài toán học đa nhiệm	10
1.3	Phương pháp hướng giảm đa mục tiêu	11
1.3.1	Thuật toán hướng giảm sâu nhất đa mục tiêu	11
1.3.2	Thuật toán hướng giảm PMTL	16
2	Thuật toán hướng giảm sâu nhất tự thích nghi	20
2.1	Thuật toán đề xuất cho tối ưu đa mục tiêu không ràng buộc	21
2.2	Thuật toán đề xuất cho tối ưu đa mục tiêu có ràng buộc	27
2.3	Thuật toán đề xuất cho học đa nhiệm	31
3	Tính toán thử nghiệm và đánh giá kết quả	33
3.1	Một số ví dụ về quy hoạch đa mục tiêu	33
3.2	Ví dụ về bài toán học đa nhiệm	39

Chương 1

Cơ sở lý thuyết

1.1 Một số khái niệm và định nghĩa

Trong mục này, chúng tôi sẽ quy ước một số kí hiệu dùng trong đề án, đồng thời giới thiệu một số kiến thức cơ bản về tối ưu đa mục tiêu và định nghĩa hướng giảm trong tối ưu đa mục tiêu.

Định nghĩa 1.1.1 [20] Một hàm khả vi $f : \mathbb{R}^n \rightarrow \mathbb{R}$ được gọi là

i) lồi trên U nếu với mọi $x, y \in U$, $\lambda \in [0, 1]$ có

$$f(\lambda x + (1 - \lambda)y) \leq \lambda f(x) + (1 - \lambda)f(y).$$

ii) giả lồi U nếu với mỗi $x, y \in U$ có

$$\langle \nabla f(x), y - x \rangle \geq 0 \Rightarrow f(y) \geq f(x).$$

iii) tựa lồi trên U nếu với mỗi $x, y \in U$, $\lambda \in [0; 1]$ có

$$f(\lambda x + (1 - \lambda)y) \leq \max \{f(x); f(y)\}.$$

Bổ đề 1.1.2 [6] Một hàm khả vi f được gọi là hàm tựa lồi trên U khi và chỉ khi

$$f(y) \leq f(x) \Rightarrow \langle \nabla f(x), y - x \rangle \leq 0.$$

Tính chất cơ bản: Nếu f là lồi $\Rightarrow f$ giả lồi $\Rightarrow f$ tựa lồi ([20]).

Ảnh của ma trận $M \in \mathbb{R}^{m \times n}$ sẽ được ký hiệu bởi $R(M)$ và $I \in \mathbb{R}^{n \times n}$ ký hiệu cho ma trận đơn vị. Cho hai ma trận $A, B \in \mathbb{R}^{n \times n}$, $B \leq A$ ($B < A$) nghĩa là $A - B$ xác định dương.

Trong phần tiếp theo, ký hiệu chuẩn Euclidean trong \mathbb{R}^n sẽ được ký hiệu bằng $\|\cdot\|$ và $B[x, r]$ ký hiệu cho quả cầu đóng bán kính r với tâm $x \in \mathbb{R}^n$. Chúng ta sẽ dùng ký hiệu $\|\cdot\|$ tương ứng cho chuẩn của không gian ma trận.

Cho điểm $x \in \mathbb{R}^n$, ký hiệu $P_U(x)$ là ảnh của điểm x trên U :

$$P_U(x) := \operatorname{argmin}\{\|z - x\| : z \in U\}. \quad (1.1)$$

Bổ đề 1.1.3 [2] *Ta có tính chất sau:*

- (i) $\|P_U(x) - P_U(y)\| \leq \|x - y\|$ với mọi $x, y \in \mathbb{R}^n$,
- (ii) $\langle y - P_U(x), x - P_U(x) \rangle \leq 0$ với mọi $x \in \mathbb{R}^n, y \in U$.

Bổ đề 1.1.4 [6] *Giả sử ∇f là hàm L -Lipschitz liên tục trên U . Với mọi $x, y \in U$, ta có:*

$$|f(y) - f(x) - \langle \nabla f(x), y - x \rangle| \leq \frac{L}{2} \|y - x\|^2.$$

1.2 Phát biểu bài toán

1.2.1 Bài toán quy hoạch đa mục tiêu

Ta xét bài toán tối ưu đa mục tiêu sau:

$$\operatorname{Min}_{x \in U} F(x), \quad (\operatorname{MOP}(F, U))$$

trong đó tập $U \subset \mathbb{R}^n$ khác rỗng, lồi đóng và hàm vector $F : U \rightarrow \mathbb{R}^m$ là một hàm khả vi trên tập mở chứa U . Ta ký hiệu ma trận Jacobian của F tại x có dạng sau:

$$DF(x) = \begin{bmatrix} \nabla F_1(x)^T \\ \vdots \\ \nabla F_m(x)^T \end{bmatrix} \in \mathbb{R}^{m \times n}$$

Trong đề án này, ta xét ma trận Jacobian DF của F là liên tục Lipschitz, nghĩa là các đạo hàm $\nabla F_j \in \mathbb{R}^n$ của hàm thành phần F_j là liên tục Lipschitz với mọi $j = 1, \dots, m$. Giả sử tập nghiệm của $(\operatorname{MOP}(F, U))$ khác rỗng. Các định nghĩa dưới đây được trích ra từ [8].

Định nghĩa 1.2.1 *Một điểm $x^* \in U$ được gọi là nghiệm hữu hiệu (efficient point) hay một điểm tối ưu Pareto của bài toán $(\operatorname{MOP}(F, U))$ nếu $\nexists y \in U, F(y) \leq F(x^*)$ và $F(y) \neq F(x^*)$, trong đó dấu \leq giữa các vector được hiểu theo nghĩa quan hệ giữa từng hàm thành*

phần. Tức là, $\nexists y \in U : F_i(y) \leq F_i(x^*), \forall i = 1, \dots, m$ và $\exists i_0 \in \{1, \dots, m\} : F_{i_0}(y) < F_{i_0}(x^*)$. Tập các điểm tối ưu Pareto tạo thành Pareto Front.

Xét $\mathbb{R}_+^m := \mathbb{R}_+ \times \dots \times \mathbb{R}_+$, khi đó thứ tự từng phần (partial order) giữa giá trị hàm vector được định nghĩa:

$$F(y) \leq F(z) \iff F(z) - F(y) \in \mathbb{R}_+^m,$$

Do đó, để giải bài toán $(MOP(F, U))$, ta cần tìm điểm cực tiểu theo thứ tự từng phần.

Định nghĩa 1.2.2 Một điểm $x^* \in U$ được gọi là nghiệm hữu hiệu yếu (weakly efficient) hay điểm tối ưu Pareto yếu của bài toán $(MOP(F, U))$ nếu $\nexists y \in U$ sao cho $F(y) < F(x^*)$, trong đó bất đẳng thức $F(y) < F(x^*)$ cũng được hiểu theo nghĩa thứ tự từng phần.

Xét $\mathbb{R}_{++}^m := \mathbb{R}_{++} \times \dots \times \mathbb{R}_{++}$ đồng thời định nghĩa

$$-\mathbb{R}_{++}^m = \{-v : v \in \mathbb{R}_{++}^m\}.$$

Định nghĩa 1.2.3 Một điểm $x^* \in U$ được gọi là điểm hữu hiệu địa phương (locally efficient) (tương ứng hữu hiệu địa phương yếu) của bài toán $(MOP(F, U))$ nếu tồn tại một vùng lân cận $V \subseteq U$ của x^* sao cho điểm x^* là nghiệm hữu hiệu (tương ứng hữu hiệu yếu) của bài toán $(MOP(F, U))$ giới hạn trên V .

Lưu ý:

- Khi U lồi và F là hàm \mathbb{R}_+^m -lồi (tức các hàm thành phần của F lồi) thì mỗi điểm tối ưu Pareto địa phương là điểm tối ưu Pareto toàn cục.
- Mỗi điểm hữu hiệu địa phương cũng là điểm hữu hiệu địa phương yếu.

Định nghĩa 1.2.4 [10] Cho $z \in U$ một vector $v \in \mathbb{R}^n$ được gọi là hướng tiếp xúc của U tại z nếu tồn tại một chuỗi $(z^k)_k \subseteq U$ và một giá trị vô hướng $\lambda \in \mathbb{R}_+$ sao cho

$$\lim_{k \rightarrow \infty} z^k = z \quad \text{và} \quad \lim_{k \rightarrow \infty} \lambda \frac{z^k - z}{\|z^k - z\|} = v$$

Tập tất cả các hướng tiếp xúc U tại z được gọi là nón tiếp xúc của U tại z và sẽ được ký hiệu bởi $T(U, z)$.

Với tập U là lồi đóng thì $T(U, z) := U(z) := \{s \in \mathbb{R}^n | s = u - z \text{ với } u \in U\}$.

Định nghĩa 1.2.5 *Ta định nghĩa một điểm $x^* \in U$ là một điểm dừng Pareto (Pareto stationary point) (hoặc Pareto critical point) của hàm vector F trên U nếu*

$$DF(T(U, x^*)) \cap (-\mathbb{R}_{++}^m) = \emptyset, \quad (1.2)$$

trong đó $DF(x^*)$ là ma trận Jacobian của F tại x^* .

Định nghĩa này là điều kiện cần (nhưng không phải là điều kiện đủ) cho một điểm là điểm hữu hiệu Pareto và được lần đầu sử dụng tại [10] để xác định thuật toán hướng giảm sâu nhất đa mục tiêu. Trong trường hợp $m = 1$, (1.2) trở thành bài toán tối ưu truyền thống trong tối ưu vô hướng (tối ưu đơn mục tiêu).

Bổ đề 1.2.6 *Khi hàm F giả lồi (tức các hàm thành phần F_i , $i = 1, \dots, m$ giả lồi) thì (1.2) là điều kiện cần và đủ cho một điểm là điểm hữu hiệu yếu.*

Thật vậy giả sử ngược lại, tồn tại điểm $y \in U$ sao cho $F(y) < F(x)$. Vì F_i là giả lồi với mỗi $i = 1, \dots, m$ suy ra $\langle \nabla F_i(x), (y - x) \rangle < 0$ và do đó $DF(x)(y - x) \in -\mathbb{R}_{++}^m$ mâu thuẫn với (1.2) (chú ý ta đang xét tập U lồi, đóng).

1.2.2 Bài toán học đa nhiệm

Trong phần này, chúng tôi sẽ mô hình hóa bài toán MTL dưới dạng bài toán MOP dựa trên ý tưởng của Sener et.al [22].

Bài toán MTL bao gồm việc tiến hành m tác vụ với hàm vector mất mát :

$$\min_{\theta} \mathcal{L}(\theta) = (\mathcal{L}_1(\theta), \mathcal{L}_2(\theta), \dots, \mathcal{L}_m(\theta))^T, \quad (\text{MTL})$$

trong đó, $\mathcal{L}_i(\theta)$ là hàm mất mát cho tác vụ thứ i với bộ tham số θ cho các tác vụ. Các thuật toán giải MTL sẽ đi tối ưu hóa đồng thời tất cả các tác vụ. Bài toán (MTL) là một bài toán tối ưu đa mục tiêu, không một nghiệm tối ưu hay bộ tham số tối ưu nào có thể tối ưu đồng thời toàn bộ các tác vụ. Do đó, ta chỉ có thể đưa ra các nghiệm tối ưu Pareto biểu thị cho sự đánh đổi độ ưu tiên giữa các tác vụ.

Định nghĩa tính trội của các tham số trong bài toán MTL: Xét hai bộ tham số θ^a, θ^b trong Ω , θ^a được gọi là trội hơn θ^b ($\theta^a \prec \theta^b$) khi và chỉ khi $\mathcal{L}_i(\theta^a) \leq \mathcal{L}_i(\theta^b)$, $\forall i \in \{1, \dots, m\}$ và $\mathcal{L}_j(\theta^a) < \mathcal{L}_j(\theta^b)$, $\exists j \in \{1, \dots, m\}$.

Định nghĩa điểm tối ưu Pareto cho bài toán MTL: θ^* được gọi là nghiệm tối ưu Pareto và $\mathcal{L}(\theta^*)$ được gọi là giá trị tối ưu Pareto khi và chỉ khi không tồn tại $\hat{\theta} \in \Omega$ sao cho $\hat{\theta} \prec \theta^*$. Tập tất cả các điểm tối ưu Pareto được gọi là tập nghiệm tối ưu Pareto và ảnh của chúng được gọi là đường cong Pareto (Pareto front).

1.3 Phương pháp hướng giảm đa mục tiêu

1.3.1 Thuật toán hướng giảm sâu nhất đa mục tiêu

Phương pháp hướng giảm là công cụ thường được sử dụng trong việc giải các bài toán quy hoạch đa mục tiêu cũng như đơn mục tiêu với các lớp hàm phi tuyến từ lồi đến không lồi và có rất nhiều ứng dụng thực tế ([4], [5]). Tại mỗi bước lặp, phương pháp hướng giảm sẽ xây dựng một dãy điểm sao cho giá trị hàm mục tiêu giảm dần và dãy điểm đó hội tụ đến điểm dừng của thuật toán. Trong phần này, chúng tôi sẽ trình bày về hướng giảm sâu nhất giải bài toán quy hoạch đa mục tiêu trong 2 trường hợp: không ràng buộc và có ràng buộc. chúng tôi sẽ trình bày kiến thức về hướng giảm sâu nhất đa mục tiêu dựa trên bài báo của Fliege et.al [10] và Desideri et.al [7].

Giải bài toán quy hoạch đa mục tiêu không ràng buộc

Ta tiến hành giải bài toán $(\text{MOP}(F, U))$ trong trường hợp $U = \mathbb{R}^n$. Cho $x \in \mathbb{R}^n$, ta định nghĩa hướng giảm sâu nhất $s(x)$ tại x là nghiệm của bài toán tối ưu sau:

$$\begin{cases} \min & \max_{j=1, \dots, m} \nabla F_j(x)^T s + \frac{1}{2} s^T s \\ \text{v.đ.k} & s \in \mathbb{R}^n, \end{cases} \quad (1.3)$$

Bài toán (1.3) luôn có nghiệm tối ưu duy nhất vì hàm $\nabla F_j(x)^T s + \frac{1}{2} s^T s$ là hàm lồi mạnh theo biến s với $j = 1, \dots, m$. Trong trường hợp $m = 1$ hướng $s(x)$ giảm sâu nhất chính là $-\nabla F(x)$.

Việc xây dựng bài toán (1.3) do chúng ta đang tiến hành xấp xỉ

$$\max_{j=1, \dots, m} F_j(x + s) - F_j(x)$$

bằng khai triển Taylor cấp 2 tại x của mỗi hàm F_j . Việc thiết lập bài toán tối ưu (1.3) để tìm hướng giảm sâu nhất cho bài toán quy hoạch đa mục tiêu đã được đề xuất tại [10]. Nghiệm tối ưu cho (1.3) sẽ được ký hiệu bằng $\Theta(x)$. Do đó,

$$\Theta(x) = \inf_{s \in \mathbb{R}^n} \max_{j=1, \dots, m} \nabla F_j(x)^T s + \frac{1}{2} s^T s, \quad (1.4)$$

và

$$s(x) = \arg \min_{s \in \mathbb{R}^n} \max_{j=1, \dots, m} \nabla F_j(x)^T s + \frac{1}{2} s^T s \quad (1.5)$$

Mặc dù bài toán (1.3) là bài toán không trơn, nó vẫn có thể được xem là bài toán tối ưu bậc hai và có thể được giải một cách hiệu quả dựa vào điều kiện KKT (Karush-Kuhn-Tucker), (1.3) tương đương với

$$\begin{cases} \min & g(t, s) = t \\ \text{v.đ.k} & \nabla F_j(x)^T s + \frac{1}{2} s^T s - t \leq 0 \quad (1 \leq j \leq m) \\ & (t, s) \in \mathbb{R} \times \mathbb{R}^n. \end{cases} \quad (1.6)$$

Hàm mục tiêu Lagrangian tương ứng là

$$L((t, s), \lambda) = t + \sum_{j=1}^m \lambda_j \left(\nabla F_j(x)^T s + \frac{1}{2} s^T s - t \right).$$

Dựa vào điều kiện KKT, ta có hệ:

$$\sum_{j=1}^m \lambda_j = 1, \quad \sum_{j=1}^m \lambda_j (\nabla F_j(x) + s) = 0, \quad (1.7)$$

$$\lambda_j \geq 0, \quad \nabla F_j(x)^T s + \frac{1}{2} s^T s \leq t \quad (1 \leq j \leq m), \quad (1.8)$$

$$\lambda_j \left(\nabla F_j(x)^T s + \frac{1}{2} s^T s - t \right) = 0 \quad (1 \leq j \leq m). \quad (1.9)$$

Bài toán (1.6) có nghiệm tối ưu duy nhất, $(\Theta(x), s(x))$ do đây là bài toán tối ưu lồi và giả sử điều kiện Slater thỏa mãn khi đó tồn tại bộ số KKT $\lambda = \lambda(x)$ tương ứng với $s = s(x)$ và $t = \Theta(x)$ thỏa mãn điều kiện (1.7)-(1.9). Cụ thể từ (1.7) ta xác định:

$$s(x) = - \sum_{j=1}^m \lambda_j(x) \nabla F_j(x). \quad (1.10)$$

Hướng giảm sâu nhất được tìm ra bằng bộ trọng số λ sau khi giải hệ KKT đưa bài toán tối ưu đa mục tiêu về bài toán tối ưu vô hướng. Giả sử điều kiện chính quy được thỏa mãn, khi đó nghiệm KKT cho bài toán quy hoạch lồi (1.6) chính là nghiệm của bài toán

tối ưu ban đầu và bài toán đối ngẫu. Từ (1.10), ta có:

$$\begin{aligned}
\Theta(x) &= \sup_{\lambda \geq 0} \inf_{s \in \mathbb{R}^n} L((t, s), \lambda) \\
&= \sup_{\substack{\lambda \geq 0 \\ \sum \lambda_j = 1}} \inf_{s \in \mathbb{R}^n} \sum_{j=1}^m \lambda_j \left(\nabla F_j(x)^T s + \frac{1}{2} s^T s \right) \\
&= \sup_{\substack{\lambda \geq 0 \\ \sum \lambda_j = 1}} \inf_{s \in \mathbb{R}^n} \left(\sum_{j=1}^m \lambda_j(x) \nabla F_j(x)^T \cdot s + \frac{1}{2} s^T s \right) \\
&= \sup_{\substack{\lambda \geq 0 \\ \sum \lambda_j = 1}} -\frac{1}{2} \left\| \sum_{j=1}^m \lambda_j(x) \nabla F_j(x) \right\|^2
\end{aligned} \tag{1.11}$$

Phần tiếp theo, ta sẽ đi khảo sát tính chất của Θ và phân tích quan hệ của nó với $s(x)$ và tính dừng của x .

Bổ đề 1.3.1 Với $x \in U$ thì

$$\Theta(x) = -\frac{1}{2} s(x)^T s(x) \text{ và } \|\Theta(x)\| = \frac{1}{2} \|s(x)\|^2.$$

Bổ đề 1.3.2 [10] Với những điều kiện giả thiết đặt ra thì

1. Với mọi $x \in U$ thì $\Theta(x) \leq 0$.
2. Những điều kiện sau là tương đương
 - (a) Điểm x là điểm không dừng (non-critical).
 - (b) $\Theta(x) < 0$.
 - (c) $s(x) \neq 0$. Cụ thể, $x \in U$ là điểm dừng Pareto khi và chỉ khi $\Theta(x) = 0$.
3. Hàm $s : U \rightarrow \mathbb{R}^n$, cho bởi (1.5) bị chặn trên tập compact và $\theta : U \rightarrow \mathbb{R}$ cho bởi công thức (1.4) liên tục trên U .

Tại mỗi bước, với một điểm không là điểm dừng, ta sẽ tiến hành giải bài toán (1.3) để xác định hướng giảm sâu nhất theo công thức (1.5). Sau đó cỡ bước giảm có thể được tìm bằng thủ tục quay lui hoặc thủ tục tìm chính xác theo tia. Thuật toán tìm hướng giảm sâu nhất giải bài toán quy hoạch đa mục tiêu được đề xuất tại [10] như sau:

Thuật toán 1 Thuật toán hướng giảm sâu nhất giải bài toán quy hoạch đa mục tiêu không ràng buộc

Bước 1. Chọn $\beta \in [0, 1]$, $\sigma \in [0, 1]$, và $x^{(1)} \in \mathbb{R}^n$, đặt $k := 1$.

Bước 2. Nếu x^k là điểm dừng Pareto thì dừng lại.

Bước 3. Tính hướng giảm s^k bằng cách giải xấp xỉ bài toán (1.3) tại $x = x^k$ với sai số σ .

Bước 4. Xác định cỡ bước $t_k \in [0, 1]$ là giá trị lớn nhất của

$$T_k := \{t = 1/2^j \mid j \in \mathbb{N}, F(x^k + ts^k) \leq F(x^k) + \beta t JF(x^k) s^k\}.$$

Bước 5. Đặt $x^{(k+1)} := x^k + t_k v^k$, $k := k + 1$ và quay trở lại Bước 2.

Cỡ bước t_k được xác định trong thuật toán trên dựa vào quy tắc Armijo. Ngoài ra, ta có thể có một số cách thiết lập cỡ bước khác như:

- Cỡ bước cố định: $t_k = h > 0$.
- Cỡ bước giảm theo bước lặp: $t_k = \frac{h}{\sqrt{k+1}}$.
- Thủ tục tìm chính xác theo tia: $t_k = \arg \min_{h \geq 0} F(x^k + hs(x^k))$.

Giải bài toán quy hoạch đa mục tiêu có ràng buộc

Xét bài toán tối ưu với F là hàm đa mục tiêu khả vi trên tập chấp nhận được U và ma trận Jacobian $DF \in \mathbb{R}^{m \times n}$ của hàm F liên tục Lipschitz, nghĩa là đạo hàm $\nabla F_j \in \mathbb{R}^n$ của hàm thành phần F_j là liên tục Lipschitz với mọi $j = 1, \dots, m$. Ta xét bài toán tối ưu:

$$\text{Min}_{x \in U} F(x), \quad (MOP_{Cons}(F, U))$$

trong đó $U := \{x \in \mathbb{R}^n \mid g_i(x) \leq 0, \forall i = 1, \dots, l\}$ với các hàm ràng buộc khả vi $g_i, \forall i = 1, \dots, K$. Giả sử tập nghiệm cho $(MOP_{Cons}(F, U))$ là khác rỗng.

Điều kiện cần để một điểm $x \in U$ là nghiệm tối ưu Pareto địa phương là:

$$DF(T(U, x)) \cap (-\mathbb{R}_{++}^m) = \emptyset, \quad (1.12)$$

tương đương với hệ

$$(\nabla F_i(x))^T s < 0, \quad \forall i = 1, \dots, m, \quad s \in T(U, x), \quad (1.13)$$

không có nghiệm s . Tập các ràng buộc thỏa mãn chặt tại z được ký hiệu bởi $I_0(x)$,

$$I_0(x) := \{j \mid g_j(x) = 0\},$$

khi đó nón tuyến tính (*linearized cone*) tại x , ký hiệu là $C(x)$, được định nghĩa:

$$C(x) := \left\{ s \in \mathbb{R}^n \mid (\nabla g_j(x))^\top s \leq 0, \forall j \in I_0(x) \right\}.$$

Giả sử điều kiện chính quy (regular condition) $C(x) = T(U, x)$ thỏa mãn, khi đó điều kiện (1.12) (và 1.13) tương đương việc hệ

$$\begin{aligned} \nabla F_i(x)^\top s &< 0, \quad \forall i \\ \nabla g_j(x)^\top s &\leq 0 \quad \forall j \in I_0(x) \end{aligned}$$

không có nghiệm $s \in \mathbb{R}^n$. Gọi \hat{A} là ma trận được xác định bởi việc thêm các hàng $(\nabla g_j(x))^\top$ cho tất cả các ràng buộc j thỏa mãn chặt vào dưới ma trận Jacobian $DF(x)$. Điều kiện cần để hệ trên vô nghiệm là

$$R(\hat{A}) \cap -\mathbb{R}_{++}^m = \emptyset.$$

Xét tập chỉ số $I := \{1, \dots, l\}$ với $\varepsilon \in \mathbb{R}_+$ và $x \in \mathbb{R}^n$ sử dụng ý tưởng tại [10], ta định nghĩa tập I_ε như sau:

$$I_\varepsilon(x) := \{j \in I \mid g_j(x) \geq -\varepsilon\}.$$

Sau đó, ta tiến hành giải bài toán tối ưu sau nhằm tìm hướng giảm cho bài toán $(MOP_{Cons}(F, U))$:

$$\begin{aligned} \min \quad & \alpha \\ \text{v.đ.k} \quad & (\nabla F_i(x))^\top s \leq \alpha, \quad i = 1, \dots, m, \\ & (\nabla g_j(x))^\top s \leq \alpha, \quad j \in I_\varepsilon(x), \\ & \|s\|_\infty \leq 1, \end{aligned} \tag{1.14}$$

Ký hiệu $\Theta(x, \varepsilon)$ là giá trị tối ưu tìm được. Dựa theo bổ đề Farkas ta có $\Theta(x, 0) = 0$ với một điểm hữu hiệu x khi và chỉ khi tồn tại vector $\lambda \in \mathbb{R}_+^m \setminus \{0\}, \beta \in \mathbb{R}_+^\ell$ với

$$\sum_{i=1}^m \lambda_i \nabla F_i(x) + \sum_{j=1}^\ell \beta_j \nabla g_j(x) = 0$$

và

$$\sum_{j=1}^\ell \beta_j g_j(x) = 0$$

Với $\lambda = (\lambda_i), \beta = (\beta_i) \in \mathbb{R}_{++}^m$ thì đây là điều kiện đủ cho x thành nghiệm tối ưu Pareto.

Thuật toán 2 Thuật toán hướng giảm sâu nhất giải bài toán quy hoạch đa mục tiêu có ràng buộc

Bước 1: Chọn $\beta \in [0, 1]$, $\varepsilon_1 \in \mathbb{R}_{++}$ và $x^1 \in U$, đặt $k := 1$.

Bước 2: Giải bài toán (1.14) để xác định hướng giảm s^k và giá trị tối ưu $\Theta(x^k, \varepsilon_k)$.

Bước 3: Nếu $\Theta(x^k, \varepsilon_k) \geq -\varepsilon_k$ thì $\varepsilon_{k+1} := \varepsilon_k/2$, $x^{k+1} := x^k$, $k := k + 1$ và chuyển sang **Bước 2**.

Bước 4: Xác định cỡ bước $t_k \in]0, 1]$ sao cho

$$F(x^k + t_k s^k) \leq F(x^k) + \beta t_k JF(x^k) s^k$$

và $x^k + t_k s^k \in U$.

Bước 5: Cập nhật $x^{k+1} := x^k + t_k s^k$, $k := k + 1$ và sang **Bước 2**.

1.3.2 Thuật toán hướng giảm PMTL

Trong phần này, chúng tôi sẽ giới thiệu thuật toán PMTL được đề xuất bởi [17]. Ý tưởng chính của thuật toán dựa trên việc phân rã không gian ảnh thành các bài toán con đồng thời giải bài toán tối ưu đa mục tiêu kết hợp các ràng buộc bất đẳng thức tuyến tính để các nghiệm tìm được trải đều trên Pareto Front.

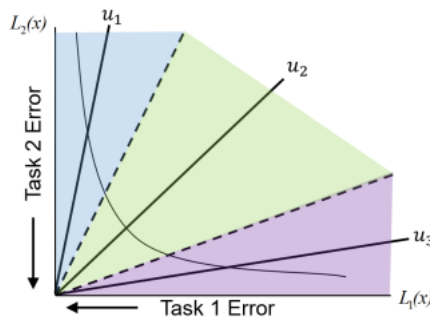
Phân rã bài toán Multi-task Learning

Xuất phát từ ý tưởng của [18], ta phân rã bài toán MTL thành K bài toán con tương ứng với tập vector tham chiếu trải đều $\{\mathbf{u}_1, \mathbf{u}_2, \dots, \mathbf{u}_K\} \in \mathbb{R}_+^m$. Giả sử các hàm thành phần đều dương, bài toán con tương ứng với vector tham chiếu \mathbf{u}_k được định nghĩa:

$$\min_{\theta} \mathcal{L}(\theta) = (\mathcal{L}_1(\theta), \mathcal{L}_2(\theta), \dots, \mathcal{L}_m(\theta))^T, \text{ v.đ.k } \mathcal{L}(\theta) \in \Omega_k, \quad (1.15)$$

trong đó $\Omega_k (k = 1, \dots, K)$ là một vùng trong không gian ảnh:

$$\Omega_k = \{\mathbf{v} \in \mathbb{R}_+^m \mid \langle \mathbf{u}_j, \mathbf{v} \rangle \leq \langle \mathbf{u}_k, \mathbf{v} \rangle, \forall j = 1, \dots, K\}$$



Hình 1.1: Phân rã bài toán tối ưu đa mục tiêu trong thuật toán PMTL

Nghĩa là, $\mathbf{v} \in \Omega_k$ khi và chỉ khi \mathbf{v} hợp với \mathbf{u}_k một góc nhỏ nhất so với K vector tham chiếu còn lại. Bài toán (1.15) tương đương:

$$\begin{aligned} \min_{\theta} \mathcal{L}(\theta) &= (\mathcal{L}_1(\theta), \mathcal{L}_2(\theta), \dots, \mathcal{L}_m(\theta))^T \\ \text{v.đ.k } \mathcal{G}_i(\theta_t) &= \langle \mathbf{u}_j - \mathbf{u}_k, \mathcal{L}(\theta_t) \rangle \leq 0, \forall i = 1, \dots, K, \end{aligned}$$

Như đã chỉ ra trong Hình 1.1 thì các vector tham chiếu chia không gian ảnh thành các vùng, mỗi vùng sẽ thu được một nghiệm tối ưu Pareto.

Tìm điểm khởi tạo

Để giải bài toán (1.15), ta cần tìm một điểm khởi tạo hữu hiệu, thỏa mãn mọi ràng buộc. Cho trước nghiệm ngẫu nhiên θ_r , cách làm đơn giản nhất là tìm θ_0 thỏa mãn:

$$\min_{\theta_0} \|\theta_0 - \theta_r\|^2 \quad \text{v.đ.k } \mathcal{L}(\theta_0) \in \Omega_k.$$

Tuy nhiên điều này dẫn tới việc giải bài toán n chiều có ràng buộc ([17]). Do vậy, ta thực hiện cách làm hiệu quả hơn như sau.

Với nghiệm θ_r , ta xác định tập các ràng buộc chặt $I(\theta_r) = \{i \mid \mathcal{G}_i(\theta_r) \geq 0, i = 1, \dots, K\}$. Ta có thể tìm một hướng giảm d_r như sau:

$$(d_r, \alpha_r) = \arg \min_{d \in R^n, \alpha \in R} \alpha + \frac{1}{2} \|d\|^2, \quad \text{v.đ.k } \nabla \mathcal{G}_i(\theta_r)^T d \leq \alpha, i \in I(\theta_r).$$

và tiến hành cập nhật $\theta_{r_{t+1}} = \theta_{r_t} + \eta_r d_{r_t}$ cho đến khi tìm được một nghiệm chấp nhận được.

Giải các bài toán con

Khi đã có điểm khởi tạo ta sẽ dùng phương pháp gradient để giải các bài toán con. θ^* là nghiệm tối ưu Pareto cho $\mathcal{L}(\theta)$ trên Ω_k nếu $\theta^* \in \Omega_k$ và không tồn tại $\hat{\theta} \in \Omega_k$ sao cho $\hat{\theta} \leq \theta^*$.

Dựa trên [10], ta xác định hướng giảm cho bài toán MOP có ràng buộc tuyến tính dựa trên việc giải bài toán sau:

$$\begin{aligned} (d_t, \alpha_t) &= \arg \min_{d \in R^n, \alpha \in R} \alpha + \frac{1}{2} \|d\|^2 \\ \text{v.đ.k } \nabla \mathcal{L}_p(\theta_t)^T d &\leq \alpha, p = 1, \dots, m. \\ \nabla \mathcal{G}_i(\theta_t)^T d &\leq \alpha, i \in I_\epsilon(\theta_t), \end{aligned} \tag{1.16}$$

Tổng quát hóa công thức hướng giảm

Bằng việc giải bài toán tối ưu có ràng buộc (1.16), ta sẽ xác định được một hướng giảm đa mục tiêu. Tuy nhiên việc giải bài toán (1.16) là không hiệu quả trong trường hợp không gian có số chiều lớn, nhất là trong bài toán MTL. Do đó ta sẽ viết lại bài toán (1.16) dưới dạng đối ngẫu. Dựa trên điều kiện KKT, ta có:

$$d_t = - \left(\sum_{p=1}^m \lambda_p \nabla \mathcal{L}_p(\theta_t) + \sum_{i \in I_e(\mathbf{x})} \beta_i \nabla \mathcal{G}_i(\theta_t) \right), \quad \sum_{p=1}^m \lambda_p + \sum_{i \in I_e(\theta)} \beta_i = 1, \quad (1.17)$$

với $\lambda_i \geq 0$ và $\beta_i \geq 0$ là các nhân tử Lagrange. Khi đó, ta giải bài toán đối ngẫu tương đương:

$$\begin{aligned} \max_{\lambda_p, \beta_i} & -\frac{1}{2} \left\| \sum_{p=1}^m \lambda_p \nabla \mathcal{L}_p(\theta_t) + \sum_{i \in I_e(\mathbf{x})} \beta_i \nabla \mathcal{G}_i(\theta_t) \right\|^2 \\ \text{v.đ.k} & \sum_{p=1}^m \lambda_p + \sum_{i \in I_e(\theta)} \beta_i = 1, \lambda_p \geq 0, \beta_i \geq 0, \forall p = 1, \dots, m, \forall i \in I_e(\theta). \end{aligned} \quad (1.18)$$

Dựa trên việc giải bài toán đối ngẫu (1.18) ta có thể tìm được một hướng giảm hiệu quả và có công thức tại (1.17). Việc giải các bài toán con có thể diễn ra đồng thời dựa trên các vector tham chiếu. Ví dụ với bài toán MTL 2 mục tiêu ta có thể sử dụng $K + 1$ vector tham chiếu $\left\{ \left(\cos\left(\frac{k\pi}{2K}\right), \sin\left(\frac{k\pi}{2K}\right) \right) \mid k = 0, 1, \dots, K \right\}$ để phân chia không gian ảnh thành các bài toán con khác nhau. Do đó ta có thuật toán PMTL như sau:

Thuật toán 3 Thuật toán PMTL

Input. Tập K vector tham chiếu trải đều $\mathbf{u}_1, \dots, \mathbf{u}_K$.

- 1: **Bước 2.** Cập nhật (Tiến hành giải song song cho từng bài toán con)
- 2: **for** $k = 1$ to K **do**
- 3: Khởi tạo bộ tham số ngẫu nhiên θ_r^k .
- 4: Tìm tham số hữu hiệu θ_0^k từ θ_r^k sử dụng phương pháp hướng giảm.
- 5: **for** $t = 1$ to T **do**
- 6: Xác định bộ tham số λ^k, β^k dựa vào việc giải bài toán (1.18) từ đó xây dựng công thức hướng giảm như công thức (1.17)
- 7: Cập nhật tham số $\theta_{t=1}^k = \theta_t^k + \eta d_t^k$.
- 8: **end for**
- 9: **end for**

Output: Bộ tham số tối ưu ứng với K vector tham chiếu.

Nhận xét: Thuật toán PMTL vẫn được sử dụng để giải bài toán $(MOP(F, U))$ với việc thay đổi các hàm mục tiêu và nghiệm x vào công thức (1.17) và (1.18).

Chương 2

Thuật toán hướng giảm sâu nhất tự thích nghi

Trong phần này, chúng tôi sẽ trình bày tổng quan về cách thiết lập cỡ bước trong các thuật toán hướng giảm giải bài toán quy hoạch đơn mục tiêu cũng như đa mục tiêu. Sau đó, chúng tôi sẽ trình bày về thuật toán đề xuất tìm hướng giảm kết hợp cỡ bước tự thích nghi để giải bài toán tối ưu đa mục tiêu trong cả hai trường hợp có ràng buộc và không ràng buộc và ứng dụng vào bài toán MTL. Đồng thời chứng minh độ hội tụ của thuật toán.

Khảo sát cách thiết lập cỡ bước trong các thuật toán hướng giảm

Gần đây, sự phát triển của các mô hình Học máy xử lý dữ liệu lớn với các hàm mất mát không lồi cần có các phương pháp thiết lập cỡ bước trong việc cập nhật tham số bằng hướng giảm một cách hiệu quả ([5], [14]). Việc thiết lập cỡ bước bằng thủ tục chính xác theo tia hoặc thủ tục quay lui tỏ ra không hiệu quả, có chi phí tính toán lớn trong trường hợp hàm phức tạp hoặc hàm vector có số thành phần lớn ([4]). Để tránh cách làm đó, ta có thể thiết lập cỡ bước cố định dựa trên hệ số Lipschitz của hàm. Tuy nhiên, cách làm này có thể gây ra các ước lượng không chính xác và dẫn tới việc làm giảm tốc độ hội tụ của thuật toán ([21]).

Mặc dù Kiwiel đã phát triển phương pháp hướng giảm giải bài toán quy hoạch tựa lồi vào 2001 nhưng kết quả hội tụ khá chậm do việc giảm dần giá trị cỡ bước ([12]). Sau đó, Yu et al và ([23], 2019), Hu et al. ([11], 2020), đã cải tiến cách làm này bằng việc sử dụng cỡ bước cố định nhưng các hàm mục tiêu vẫn phải thỏa mãn điều kiện Holder. Một số

cách tiếp cận khác sử dụng hệ động lực hoặc mạng recurrent neural để giải bài toán quy hoạch giả lồi với tập ràng buộc không bị chặn ([19], [3]), trong đó cỡ bước được giữ cố định và không tự thích nghi.

Cỡ bước tự thích nghi đã được đề xuất bởi Konnov et al ([13]) và Ferreira et al ([9]). Trong [13], chúng tôi đề xuất hướng giảm điều hướng (conditional gradient) kết hợp cỡ bước tự thích nghi để áp dụng cho lớp hàm mục tiêu giả lồi với tập chấp nhận được bị chặn. Phương pháp này tiếp tục được mở rộng trong [9] cho trường hợp tập chấp nhận được không bị chặn nhưng không thể áp dụng cho trường hợp giải bài toán tối ưu có ràng buộc. Để khắc phục các hạn chế trong các bài báo nêu trên, trong báo cáo này, chúng tôi đề xuất một thuật toán hướng giảm với cỡ bước tự thích nghi để giải lớp bài toán với hàm mục tiêu không lồi, trơn và tập chấp nhận được có tính lồi, đóng, không bị chặn.

2.1 Thuật toán đề xuất cho tối ưu đa mục tiêu không ràng buộc

Xét bài toán tối ưu đa mục tiêu không ràng buộc sau:

$$\text{Min}_{x \in \mathbb{R}^n} F(x). \quad (\text{MOP}(F, \mathbb{R}^n))$$

Trong đó, hàm F có tính chất như phần 1.2.1 đã đề cập. Ta sẽ tiến hành giải bài toán trên dựa theo phương pháp hướng giảm như đã được trình bày trong Phần 1.3.1. Trong trường hợp giải bài toán $(\text{MOP}(F, \mathbb{R}^n))$ bằng thuật toán MGDA (thuật toán này dựa trên định nghĩa của điểm dừng Pareto thông qua bao lồi của các vector gradient thành phần được đề xuất bởi Desideri năm 2012 [7]) thì nghiệm thu được sẽ bị tập trung tại một vùng trên Pareto Front ([17]). Minh họa cho việc tập trung nghiệm Pareto sẽ được chỉ ra trong phần thực nghiệm tại Chương 3. Do đó, để các nghiệm thu được trải đều trên mặt Pareto Front, chúng tôi đề xuất việc giải bài toán tối ưu đa mục tiêu không ràng buộc dựa trên phương pháp hướng giảm sâu nhất đa mục tiêu kết hợp với các bất đẳng thức giúp phân tách không gian ảnh thành các bài toán con và giải từng nghiệm tối ưu Pareto của bài toán con đó, tương tự như trong [17]. Phần lý thuyết về hướng giảm sâu nhất đa mục tiêu đã được chúng tôi trình bày ở Phần 1.3.1.

Phiên bản 1 của thuật toán đề xuất giải bài toán quy hoạch đa mục tiêu không ràng buộc

Mô tả thuật toán

Chúng tôi kết hợp phương pháp hướng giảm sâu nhất đa mục tiêu được trình bày ở Phần 1.3.1 với cơ bước tự thích nghi để đề xuất thuật toán giải bài toán (MOP(F, \mathbb{R}^n)) như sau:

Thuật toán 4 Thuật toán hướng giảm sâu nhất tự thích nghi phiên bản 1

Bước 1. Thiết lập $\kappa \in [0, 1]$, $\sigma \in [0, 1]$, $\alpha_1 \in (0, +\infty)$, và $x^1 \in \mathbb{R}^n$. Đặt $k := 1$.

Bước 2. (Vòng lặp chính)

(a) Tìm hướng giảm thông qua việc giải bài toán sau:

$$\min_{\lambda_j} \left\| \sum_{j=1}^m \lambda_j^k \nabla F_j(x^k) \right\|^2$$
$$\text{v.đ.k: } \lambda_j \geq 0, \sum_{j=1}^m \lambda_j^k = 1$$

Từ đó xác định được nghiệm tối ưu:

$$\lambda^k = (\lambda_j^k), s(x^k) = - \left(\sum_{j=1}^m \lambda_j^k \nabla F_j(x^k) \right) \text{ và } \Theta(x^k) = -\frac{1}{2} \|s(x^k)\|^2.$$

(b) Nếu $\Theta(x^k) = 0$ thì chuyển qua **Bước 3**. Ngược lại tiếp tục **Bước 2(c)**.

(c) Đặt $x^{k+1} := x^k + \alpha_k s(x^k)$.

(d) Tính toán cỡ bước:

$$\text{Nếu } \sum_{j=1}^m \lambda_j^k F_j(x^{k+1}) \leq \sum_{j=1}^m \lambda_j^k F_j(x^k) + \sigma \langle s(x^k), x_{temp} - x^k \rangle$$

thì $\alpha_{k+1} = \alpha_k$, ngược lại đặt $\alpha_{k+1} := \kappa \alpha_k$

Bước 3. Đặt $k := k + 1$ và chuyển đến **Bước 2**.

Chứng minh độ hội tụ

Nhận xét 2.1.1 Nếu Thuật toán (4) dừng lại sau bước k thì x^k là nghiệm dừng Pareto của bài toán (MOP(F, \mathbb{R}^n)).

Với $x \in \mathbb{R}^n$, đặt

$$H(x) = \sum_{j=1}^m \lambda_j F_j(x).$$

Ta có:

$$H(x^k) = \sum_{j=1}^m \lambda_j^k F_j(x^k) \text{ và } s(x^k) = -\nabla H(x^k) = -\sum_{j=1}^m \lambda_j^k \nabla F_j(x^k).$$

Chứng minh: Với $U = \mathbb{R}^n$, ta có: $x = P_U(x)$. Với $x^k = P_U(x^k - \alpha_k \nabla H(x^k))$ sử dụng (1.1.3)-(ii), ta có:

$$\langle z - x^{k+1}, x^k - \alpha_k \nabla H(x^k) - x^{k+1} \rangle \leq 0 \quad \forall z \in U. \quad (2.1)$$

Nếu $x^{k+1} = x^k$ thì

$$\langle \nabla H(x^k), z - x^k \rangle \geq 0 \quad \forall z \in U. \quad (2.2)$$

Nếu $U = \mathbb{R}^n$ thì $s(x^k) = -\nabla H(x^k) = 0$, nghĩa là x^k là điểm dừng Pareto. Thêm điều kiện F là hàm giả lồi từ Bổ đề (1.2.6), ta có nhận xét x^k là nghiệm hữu hiệu yếu của $(\text{MOP}(F, \mathbb{R}^n))$.

Trong trường hợp thuật toán sinh ra dãy nghiệm vô hạn, ta sẽ chứng minh dãy nghiệm đó hội tụ về nghiệm Pareto của bài toán $(\text{MOP}(F, \mathbb{R}^n))$.

Định lý 2.1.2 *Giả sử chuỗi $\{x^k\}$ sinh ra bởi Thuật toán 4. Khi đó, điểm hội tụ của dãy $\{x^k\}$ là điểm dừng Pareto của bài toán (2.9).*

Áp dụng Bổ đề (1.1.4), ta có:

$$H(x^{k+1}) \leq H(x^k) + \langle \nabla H(x^k), x^{k+1} - x^k \rangle + \frac{L}{2} \|x^{k+1} - x^k\|^2. \quad (2.3)$$

Từ (2.10) chọn $z = x^k \in U$, ta có:

$$\langle \nabla H(x^k), x^{k+1} - x^k \rangle \leq -\frac{1}{\alpha_k} \|x^{k+1} - x^k\|^2. \quad (2.4)$$

Kết hợp (2.3) với (2.4), ta có:

$$H(x^{k+1}) \leq H(x^k) - \sigma \langle \nabla H(x^k), x^k - x^{k+1} \rangle - \left(\frac{1 - \sigma}{\alpha_k} - \frac{L}{2} \right) \|x^{k+1} - x^k\|^2. \quad (2.5)$$

Ta chứng minh $\{\alpha_k\}$ không tiến dần tới 0 hay nói cách khác cỡ bước chỉ được cập nhật với số lần hữu hạn. Thật vậy, giả sử ngược lại $\alpha_k \rightarrow 0$. Từ (2.5), tồn tại $k_0 \in \mathbb{N}$ thỏa mãn

$$H(x^{k+1}) \leq H(x^k) - \sigma \langle \nabla H(x^k), x^k - x^{k+1} \rangle \quad \forall k \geq k_0.$$

Dựa trên cách xây dựng α_k , từ bất đẳng thức cuối suy ra $\alpha_k = \alpha_{k_0}$ với mọi $k \geq k_0$. Vô lý, do đó tồn tại $k_1 \in \mathbb{N}$ sao cho với mọi $k \geq k_1$, ta có $\alpha_k = \alpha_{k_1}$ và

$$H(x^{k+1}) \leq H(x^k) - \sigma \langle \nabla H(x^k), x^k - x^{k+1} \rangle. \quad (2.6)$$

Từ (2.11), ta có $\langle \nabla H(x^k), x^k - x^{k+1} \rangle \geq 0$ suy ra các tính chất sau:

$$\sum_{k=0}^{\infty} \langle \nabla H(x^k), x^k - x^{k+1} \rangle < \infty; \quad \sum_{k=0}^{\infty} \|x^{k+1} - x^k\|^2 < \infty. \quad (2.7)$$

Từ (2.10), với mọi $z \in U$, ta có

$$\begin{aligned}\|x^{k+1} - z\|^2 &= \|x^k - z\|^2 - \|x^{k+1} - x^k\|^2 + 2\langle x^{k+1} - x^k, x^{k+1} - z \rangle \\ &\leq \|x^k - z\|^2 - \|x^{k+1} - x^k\|^2 + 2\alpha_k \langle \nabla H(x^k), z - x^{k+1} \rangle.\end{aligned}\quad (2.8)$$

Cho \bar{x} là điểm hội tụ của $\{x^k\}$. Tồn tại một chuỗi con $\{x^{k_i}\} \subset \{x^k\}$ sao cho $\lim_{i \rightarrow \infty} x^{k_i} = \bar{x}$. Từ (2.8), đặt $k = k_i$ và lấy giới hạn $i \rightarrow \infty$. Chú ý $\|x^k - x^{k+1}\| \rightarrow 0$, ∇H mặt khác ∇H liên tục nên ta có:

$$\langle \nabla G(\bar{x}), z - \bar{x} \rangle \geq 0 \quad \forall z \in U.$$

Từ đó, \bar{x} là điểm dừng Pareto của $(\text{MOP}(F, \mathbb{R}^n))$.

Tiến hành thực nghiệm thuật toán như ở Phần 3, ta thấy nghiệm tìm được bởi thuật toán 4 tập trung tại một vùng trên Pareto Front. Do đó để giúp nghiệm trải đều trên Pareto Front ta tiến hành xây dựng phiên bản 2 của thuật toán 4.

Phiên bản 2 của thuật toán đề xuất giải bài toán quy hoạch đa mục tiêu không ràng buộc

Cơ sở lý thuyết của thuật toán

Xét tập K vector tham chiếu $\{\mathbf{u}_1, \mathbf{u}_2, \dots, \mathbf{u}_K\} \in \mathbb{R}_+^m$, tiến hành phân tách không gian ảnh thành các vùng không gian con Ω_k ($k = 1, \dots, K$), với

$$\Omega_k = \{\mathbf{v} \in \mathbb{R}_+^m \mid \langle \mathbf{u}_i, \mathbf{v} \rangle \leq \langle \mathbf{u}_k, \mathbf{v} \rangle, \forall i = 1, \dots, K\}.$$

Ta tiến hành giải bài toán tối ưu không ràng buộc trên từng vùng Ω_k thông qua việc kết hợp các ràng buộc bất đẳng thức \mathcal{G}_i giúp phân tách không gian. Ta cần giải bài toán sau:

$$\begin{aligned}\min_{x \in \mathbb{R}^n} F(x) &= (F_1(x), F_2(x), \dots, F_m(x)) \\ \text{v.đ.k } \mathcal{G}_i(x) &= \langle \mathbf{u}_i - \mathbf{u}_k, F(x) \rangle \leq 0, \forall i = 1, \dots, K,\end{aligned}\quad (2.9)$$

Khi đó, bài toán $(\text{MOP}(F, \mathbb{R}^n))$ sẽ trở thành bài toán $(\text{MOP}(F, U))$ với tập $U = \{x \in \mathbb{R}^n \mid \mathcal{G}_i(x) \leq 0, i = 1, \dots, K\}$.

Tìm điểm khởi tạo

Để giải bài toán (2.9), ta cần tìm một điểm khởi tạo chấp nhận được thỏa mãn mọi ràng buộc. chúng tôi sử dụng phương pháp tìm điểm chấp nhận được dựa trên hướng

giảm dựa trên ý tưởng của [17]. Trước khi giải bài toán tối ưu, ta sẽ tìm điểm khởi tạo dựa trên các bước sau:

- **Bước 1:** Khởi tạo ngẫu nhiên một điểm $x_0 \in \mathbb{R}^n$.
- **Bước 2:** Nếu x_0 là điểm chấp nhận được thì dừng lại. Ngược lại chuyển xuống **Bước 3**.
- **Bước 3:** Ta xác định tập các ràng buộc chặt $I(x_0) = \{i \mid \mathcal{G}_i(x_0) \geq 0, i = 1, \dots, K\}$. Ta có thể tìm một hướng giảm d_r như sau:

$$(d_r, \alpha_r) = \arg \min_{d \in \mathbb{R}^n, \alpha \in R} \alpha + \frac{1}{2} \|d\|^2, \text{ v.đ.k } \nabla \mathcal{G}_i(x_0)^T d \leq \alpha, i \in I(x_0).$$

- **Bước 4:** Tiến hành cập nhật $x_0 = x_0 + \eta_r d_r$ và quay trở lại **Bước 2**.

Sau khi tìm được điểm khởi tạo ta sẽ tiến hành đến với thuật toán chính giải bài toán quy hoạch đa mục tiêu trong các trường hợp được trình bày trong phần dưới. Dựa vào nội dung lý thuyết đã trình bày về hướng giảm sâu nhất cho bài toán quy hoạch đa mục tiêu tại Phần 1.3.1, chúng tôi đề xuất thuật toán như sau:

Thuật toán 5 Thuật toán hướng giảm sâu nhất tự thích nghi phiên bản 2

Bước 1. Thiết lập $\kappa \in [0, 1], \sigma \in [0, 1], \alpha_1 \in (0, +\infty)$, và $x_r^1 \in \mathbb{R}^n$. Đặt $k := 1$.

Bước 2. Tiến hành tìm điểm khởi tạo chấp nhận được x^1 từ x_r^1 .

Bước 3. (Vòng lặp chính)

(a) Tìm hướng giảm thông qua việc giải bài toán sau:

$$\min_{\lambda_j, \beta_i} \left\| \sum_{j=1}^m \lambda_j^k \nabla F_j(x^k) + \sum_{i \in I_\epsilon(x^k)} \beta_i^k \nabla \mathcal{G}_i(x^k) \right\|^2$$

$$\text{v.đ.k: } \lambda_j \geq 0, \beta_i \geq 0, \sum_{j=1}^m \lambda_j^k + \sum_{i \in I_\epsilon(x^k)} \beta_i^k = 1$$

Trong đó, $I_\epsilon(x^k) := \{i \in 1, \dots, K | \mathcal{G}_i(x^k) \geq -\epsilon\}$.

Từ đó xác định được nghiệm tối ưu:

$$\lambda^k = (\lambda_j^k), \beta^k = (\beta_i^k), s(x^k) = - \left(\sum_{j=1}^m \lambda_j^k \nabla F_j(x^k) + \sum_{i \in I_\epsilon(x)} \beta_i^k \nabla \mathcal{G}_i(x^k) \right) \text{ và } \Theta(x^k) = -\frac{1}{2} \|s(x^k)\|^2.$$

(b) Nếu $\Theta(x^k) = 0$ thì chuyển qua **Bước 4**. Ngược lại tiếp tục **Bước 2(c)**.

(c) Đặt $x_{temp} := x^k + \alpha_k s(x^k)$.

(d) Tính toán cỡ bước:

$$\text{Đặt } \eta_j^k := \lambda_j^k + \sum_{i \in I_\epsilon(x)} \beta_i^k$$

$$\text{Nếu } \sum_{j=1}^m \eta_j^k F_j(x_{temp}) \leq \sum_{j=1}^m \eta_j^k F_j(x^k) + \sigma \langle s(x^k), x_{temp} - x^k \rangle$$

thì $\alpha_{k+1} = \alpha_k$, ngược lại đặt $\alpha_{k+1} := \kappa \alpha_k$

Bước 3. Đặt $x^{k+1} := x^k + \alpha_{k+1} s(x^k)$, $k := k + 1$ và chuyển đến **Bước 2**.

Bước 4. Sử dụng toán tử chiếu P_U , đưa ra nghiệm $x_{out} = P_U(x^k)$.

Chứng minh độ hội tụ

Nhận xét 2.1.3 Nếu Thuật toán (5) dừng lại sau bước k và $x_{out} = P_U(x^k) = x^k$ thì x_{out} là nghiệm dừng Pareto của bài toán (2.9).

Với $x \in U$, đặt

$$H(x) = \sum_{j=1}^m \eta_j F_j(x).$$

Ta có:

$$H(x^k) = \sum_{j=1}^m \eta_j^k F_j(x^k) \text{ và } s(x^k) = -\nabla H(x^k) = -\sum_{j=1}^m \eta_j^k \nabla F_j(x^k).$$

Chứng minh:

Với $x_{out} = P_U(x^k - \alpha_k \nabla H(x^k))$ sử dụng (1.1.3)-(ii), ta có:

$$\langle z - x_{out}, x^k - \alpha_k \nabla H(x^k) - x_{out} \rangle \leq 0 \quad \forall z \in U. \quad (2.10)$$

Nếu $x_{out} = x^k$ thì

$$\langle \nabla H(x^k), z - x^k \rangle \geq 0 \quad \forall z \in U. \quad (2.11)$$

Nếu $U = \mathbb{R}^n$ thì $s(x^k) = -\nabla H(x^k) = 0$, nghĩa là x^k là điểm dừng Pareto. Thêm điều kiện F là hàm giả lồi từ Bổ đề (1.2.6), ta có nhận xét x^k là nghiệm hữu hiệu yếu của (2.9).

Kết quả thử nghiệm ở Phần 3 cho thấy độ hiệu quả của thuật toán 5.

2.2 Thuật toán đề xuất cho tối ưu đa mục tiêu có ràng buộc

Phiên bản 1 của thuật toán đề xuất giải bài toán quy hoạch đa mục tiêu có ràng buộc

Chúng tôi kết hợp phương pháp hướng giảm sâu nhất đa mục tiêu được trình bày ở Phần 1.3.1 với cỡ bước tự thích nghi, cùng quá trình tìm điểm khởi tạo để giải bài toán $(MOP_{Cons}(F, U))$ trong đó $U := \{x \in \mathbb{R}^n | g_i(x) \leq 0, \forall i = 1, \dots, l\}$ với các hàm ràng buộc khả vi $g_i, \forall i = 1, \dots, l$.

Tìm điểm khởi tạo

Tương tự như trong Phần 2.1, chúng tôi đề xuất cách tìm điểm khởi tạo chấp nhận được như sau

- **Bước 1:** Khởi tạo ngẫu nhiên một điểm $x_0 \in \mathbb{R}^n$.
- **Bước 2:** Nếu x_0 là điểm hữu hiệu thì dừng lại. Ngược lại chuyển xuống **Bước 3**.
- **Bước 3:** Ta xác định tập các ràng buộc chặt

$I_\epsilon(x_0) = \{i \mid g_i(x_0) \geq -\epsilon\}$. Ta có thể tìm một hướng giảm d_r như sau:

$$(d_r, \alpha_r) = \arg \min_{d \in \mathbb{R}^n, \alpha \in \mathbb{R}} \alpha + \frac{1}{2} \|d\|^2, \text{ v.đ.k } g_i(x_0)^T d \leq \alpha, i \in I(x_0).$$

- **Bước 4:** Tiến hành cập nhật $x_0 = P_U(x_0 + \eta_r d_r)$ và quay trở lại **Bước 2**.

Sau khi đã tìm được điểm khởi tạo hữu hiệu, ta đến với phần thuật toán chính sau

Thuật toán 6 Thuật toán hướng giảm sâu nhất tự thích nghi phiên bản 1

Bước 1. Thiết lập $\kappa \in [0, 1], \sigma \in [0, 1], \alpha_1 \in (0, +\infty)$, và $x_r^1 \in \mathbb{R}^n$. Đặt $k := 1$.

Bước 2. Tiến hành tìm điểm khởi tạo chấp nhận được x^1 từ x_r^1 .

Bước 3. (Vòng lặp chính)

(a) Tìm hướng giảm thông qua việc giải bài toán sau:

$$\min_{\lambda_j, \beta_i} \left\| \sum_{j=1}^m \lambda_j^k \nabla F_j(x^k) + \sum_{i \in I_\epsilon(x^k)} \beta_i^k \nabla g_i(x^k) \right\|^2$$

$$\text{v.đ.k: } \lambda_j \geq 0, \beta_i \geq 0, \sum_{j=1}^m \lambda_j^k + \sum_{i \in I_\epsilon(x^k)} \beta_i^k = 1$$

Trong đó, $I_\epsilon(x^k) = \{i \in 1, \dots, l \mid g_i(x^k) \geq -\epsilon\}$.

Từ đó xác định được nghiệm tối ưu:

$$\lambda^k = (\lambda_j^k), \beta^k = (\beta_i^k), s(x^k) = - \left(\sum_{j=1}^m \lambda_j^k \nabla F_j(x^k) + \sum_{i \in I_\epsilon(x^k)} \beta_i^k \nabla g_i(x^k) \right)$$

$$\text{và } \Theta(x^k) = -\frac{1}{2} \|s(x^k)\|^2.$$

(b) Nếu $\Theta(x^k) = 0$ thì chuyển sang **Bước 5**. Ngược lại tiếp tục **Bước 3(c)**.

(c) Đặt $x_{temp} := x^k + \alpha_k s(x^k)$.

(d) Tính toán cỡ bước:

$$\text{Nếu } \sum_{j=1}^m \lambda_j^k F_j(x_{temp}) + \sum_{i \in I_\epsilon(x^k)} \beta_i^k g_i(x^k) \leq \sum_{j=1}^m \lambda_j^k F_j(x^k) + \sum_{i \in I_\epsilon(x^k)} \beta_i^k g_i(x^k) + \sigma \langle s(x^k), x_{temp} - x^k \rangle$$

thì $\alpha_{k+1} = \alpha_k$, ngược lại đặt $\alpha_{k+1} := \kappa \alpha_k$.

Bước 4. Đặt $x^{k+1} := x^k + \alpha_{k+1} s(x^k)$, $k := k + 1$ và chuyển đến **Bước 3**.

Bước 5. Sử dụng toán tử chiếu P_U , đưa ra nghiệm $x_{out} = P_U(x^k)$.

Chứng minh độ hội tụ

Với $x \in U$, đặt

$$H(x) = \sum_{j=1}^m \lambda_j F_j(x) + \sum_{i \in I_\epsilon(x)} \beta_i^k g_i(x).$$

Ta có:

$$H(x^k) = \sum_{j=1}^m \lambda_j^k F_j(x^k) + \sum_{i \in I_\epsilon(x^k)} \beta_i^k g_i(x^k)$$

$$s(x^k) = -\nabla H(x^k) = - \sum_{j=1}^m \lambda_j^k \nabla F_j(x^k) - \sum_{i \in I_\epsilon(x^k)} \beta_i^k \nabla g_i(x^k).$$

Bằng cách chứng minh tương tự Phần 2.1, ta có định lý sau:

Định lý 2.2.1 Giả sử chuỗi $\{x^k\}$ sinh ra bởi Thuật toán 6, trong trường hợp thuật toán kết thúc sau dừng lại sau k bước và $x_{out} = P_U(x^k) = x^k$ thì x^k là nghiệm dừng Pareto của

bài toán $(MOP_{Cons}(F, U))$.

Phiên bản 2 của thuật toán đề xuất giải bài toán quy hoạch đa mục tiêu có ràng buộc

Cơ sở lý thuyết của thuật toán

Tương tự như trong Phần 2.1, chúng tôi đề xuất việc thêm các ràng buộc bất đẳng thức \mathcal{G}_i giúp phân tách không gian ảnh và tìm điểm khởi tạo hữu hiệu trước khi giải bài toán $(MOP_{Cons}(F, U))$. Khi đó, bài toán ta cần giải được phát biểu như sau

$$\begin{aligned} \min_{x \in U} F(x) &= (F_1(x), F_2(x), \dots, F_m(x)) \\ \text{v.đ.k } \mathcal{G}_i(x) &= \langle \mathbf{u}_j - \mathbf{u}_k, F(x) \rangle \leq 0, \forall i = 1, \dots, K, \end{aligned}$$

Khi đó, ta sẽ bổ sung các ràng buộc $\mathcal{G}_i, i = 1, \dots, K$ vào tập chấp nhận được U (đã được định nghĩa ở Phần 1.3.1).

Tìm điểm khởi tạo

Tương tự như trong Phần 2.1, chúng tôi đề xuất cách tìm điểm khởi tạo chấp nhận được như sau

- **Bước 1:** Khởi tạo ngẫu nhiên một điểm $x_0 \in \mathbb{R}^n$.
- **Bước 2:** Nếu x_0 là điểm hữu hiệu thì dừng lại. Ngược lại chuyển xuống **Bước 3**.
- **Bước 3:** Ta xác định tập các ràng buộc chặt
 $I_\epsilon(x_0) = \{i, p \mid \mathcal{G}_i(x_0) \geq 0, i = 1, \dots, K \text{ và } g_p(x) \geq -\epsilon\}$. Ta có thể tìm một hướng giảm d_r như sau:

$$\begin{aligned} (d_r, \alpha_r) &= \arg \min_{d \in \mathbb{R}^n, \alpha \in \mathbb{R}} \alpha + \frac{1}{2} \|d\|^2, \\ \text{v.đ.k } \nabla \mathcal{G}_i(x_0)^T d &\leq \alpha, i \in I(x_0) \\ \nabla g_p(x_0)^T d &\leq \alpha, p \in I(x_0) \end{aligned}$$

- **Bước 4:** Tiến hành cập nhật $x_0 = P_U(x_0 + \eta_r d_r)$ và quay trở lại **Bước 2**.

Sau khi đã tìm được điểm khởi tạo hữu hiệu, ta đến với phần thuật toán chính sau

Thuật toán 7 Thuật toán hướng giảm sâu nhất đa mục tiêu phiên bản 2

Bước 1. Thiết lập $\kappa \in [0, 1], \sigma \in [0, 1], \alpha_1 \in (0, +\infty)$, và $x^1 \in \mathbb{R}^n$. Đặt $k := 1$.

Bước 2. (Vòng lặp chính)

(a) Tìm hướng giảm thông qua việc giải bài toán sau:

$$\begin{aligned} \min_{\lambda_j, \beta_i} & \left\| \sum_{j=1}^m \lambda_j^k \nabla F_j(x^k) + \sum_{p \in I_\epsilon(x^k)} \beta_p^k \nabla g_p(x^k) + \sum_{i \in I_\epsilon(x^k)} \gamma_i^k \nabla \mathcal{G}_i(x^k) \right\|^2 \\ \text{v.đ.k: } & \lambda_j \geq 0, \beta_i \geq 0, \gamma_i \geq 0, \sum_{j=1}^m \lambda_j^k + \sum_{i \in I_\epsilon(x^k)} \beta_p^k + \sum_{p \in I_\epsilon(x^k)} \gamma_i^k = 1 \end{aligned}$$

Trong đó, $I_\epsilon(x^k) := \{j \in 1, \dots, l; p \in 1, \dots, K \mid \mathcal{G}_i(x^k) \geq 0 \text{ và } g_p(x^k) \geq -\epsilon\}$.

Từ đó xác định được nghiệm tối ưu:

$$\lambda^k = (\lambda_j^k), \beta^k = (\beta_i^k), s(x^k) = - \left(\sum_{j=1}^m \lambda_j^k \nabla F_j(x^k) + \sum_{i \in I_\epsilon(x^k)} \beta_p^k \nabla g_p(x^k) + \sum_{p \in I_\epsilon(x^k)} \gamma_i^k \nabla \mathcal{G}_i(x^k) \right)$$

$$\text{và } \Theta(x^k) = -\frac{1}{2} \|s(x^k)\|^2.$$

(b) Nếu $\Theta(x^k) = 0$ thì chuyển sang **Bước 4**. Ngược lại tiếp tục **Bước 2(c)**.

(c) Đặt $x_{temp} := x^k + \alpha_k s(x^k)$.

(d) Tính toán cỡ bước:

$$\text{Đặt } \eta_j^k := \lambda_j^k + \sum_{i \in I_\epsilon(x)} \gamma_i^k$$

$$\text{Nếu } \sum_{j=1}^m \eta_j^k F_j(x_{temp}) + \sum_{p \in I_\epsilon(x^k)} \beta_p^k g_p(x^k) \leq \sum_{j=1}^m \eta_j^k F_j(x^k) + \sum_{p \in I_\epsilon(x^k)} \beta_p^k g_p(x^k) + \sigma \langle s(x^k), x_{temp} - x^k \rangle$$

thì $\alpha_{k+1} = \alpha_k$, ngược lại đặt $\alpha_{k+1} := \kappa \alpha_k$.

Bước 3. Đặt $x^{k+1} := x^k + \alpha_{k+1} s(x^k)$, $k := k + 1$ và chuyển đến Bước 2.

Bước 4. Sử dụng toán tử chiếu P_U , đưa ra nghiệm $x_{out} = P_U(x^k)$.

Chứng minh độ hội tụ

Với $x \in U$, đặt

$$H(x) = \sum_{j=1}^m \eta_j F_j(x) + \sum_{p \in I_\epsilon(x)} \beta_p^k g_p(x).$$

Ta có:

$$H(x^k) = \sum_{j=1}^m \eta_j^k F_j(x^{k+1}) + \sum_{p \in I_\epsilon(x^k)} \beta_p^k g_p(x^k)$$

$$s(x^k) = -\nabla H(x^k) = -\sum_{j=1}^m \eta_j^k \nabla F_j(x^{k+1}) - \sum_{p \in I_\epsilon(x^k)} \beta_p^k \nabla g_p(x^k).$$

Bằng cách chứng minh tương tự Phần 2.1, ta có định lý sau:

Định lý 2.2.2 *Giả sử chuỗi $\{x^k\}$ sinh ra bởi Thuật toán 7, trong trường hợp thuật toán kết thúc sau dừng lại sau k bước và $x_{out} = P_U(x^k) = x^k$ thì x^k là nghiệm dừng Pareto của bài toán $(MOP_{Cons}(F, U))$.*

2.3 Thuật toán đề xuất cho học đa nhiệm

Tương tự trong [17], chúng tôi sẽ sử dụng K vector tham chiếu $\{\mathbf{u}_1, \mathbf{u}_2, \dots, \mathbf{u}_K\}$ để biểu thị cho việc đánh đổi cho độ ưu tiên của các tác vụ trong bài toán MTL. Đồng thời sử dụng các ràng buộc bất đẳng thức tuyến tính $\mathcal{G}_i(\theta_t) = \langle \mathbf{u}_j - \mathbf{u}_k, \mathcal{L}(\theta_t) \rangle \leq 0, \forall i = 1, \dots, K$ để phân tách không gian ảnh thành các bài toán con. Sau đó, chúng tôi đề xuất sử dụng cỡ bước tự thích nghi để cập nhật cho *learnig rate* của mạng Neural. Thuật toán đề xuất cho MTL được trình bày chi tiết dưới đây.

Thuật toán 8 Thuật toán đề xuất cho MTL

Input: Tập các vector tham chiếu $\{\mathbf{u}_1, \mathbf{u}_2, \dots, \mathbf{u}_K\}$.

- 1: Thiết lập $\kappa \in [0, 1], \sigma \in [0, 1], \alpha_1 \in (0, +\infty)$.
- 2: **for** $k = 1$ to K **do**
- 3: Khởi tạo bộ siêu tham số θ_r^k cho mạng Neural.
- 4: Tìm bộ siêu tham số hữu hiệu θ_0^k từ θ_r^k sử dụng phương pháp hướng giảm.
- 5: **for** $t = 1$ to T **do**
- 6: Tìm hướng giảm thông qua việc giải bài toán sau:

$$\begin{aligned} \min_{\lambda_j, \beta_i} & \left\| \sum_{j=1}^m \lambda_j^k \nabla \mathcal{L}_j(\theta^k)^T + \sum_{i \in I_\epsilon(\theta^k)} \beta_i^k \nabla \mathcal{G}_i(\theta^k) \right\|^2 \\ \text{v.đ.k: } & \lambda_j \geq 0, \beta_i \geq 0, \sum_{j=1}^m \lambda_j^k + \sum_{i \in I_\epsilon(\theta^k)} \beta_i^k = 1 \end{aligned}$$

- 7: Trong đó, $I_\epsilon(\theta^k) := \{i \in 1, \dots, K \mid \mathcal{G}_i(\theta^k) \geq -\epsilon\}$.
- 8: Từ đó xác định hướng giảm: $s(\theta^k) = - \left(\sum_{j=1}^m \lambda_j^k \nabla \mathcal{L}_j(\theta^k) + \sum_{i \in I_\epsilon(\theta)} \beta_i^k \nabla \mathcal{G}_i(\theta^k) \right)$.
- 9: Đặt $\theta_{temp} := \theta^k + \alpha_k s(\theta^k)$.
- 10: Tính toán cỡ bước:
- 11: Đặt $\eta_j^k := \lambda_j^k + \sum_{i \in I_\epsilon(\theta)} \beta_i^k$
- 12: Nếu $\sum_{j=1}^m \eta_j^k \mathcal{L}_j(\theta_{temp}) \leq \sum_{j=1}^m \eta_j^k \mathcal{L}_j(\theta^k) + \sigma \langle s(\theta^k), \theta_{temp} - \theta^k \rangle$
- 13: thì $\alpha_{k+1} = \alpha_k$, ngược lại đặt $\alpha_{k+1} := \kappa \alpha_k$.
- 14: Cập nhật siêu tham số $\theta^{k+1} := \theta^k + \alpha_{k+1} s(\theta^k)$, $k := k + 1$.
- 15: **end for**
- 16: **end for**

Output: Bộ siêu tham số tối ưu cho từng bài toán con $\{\theta_T^k \mid k = 1, \dots, K\}$.

Chương 3

Tính toán thử nghiệm và đánh giá kết quả

Trong phần này, chúng tôi sẽ kế thừa và xây dựng một số bài toán quy hoạch đa mục tiêu có ràng buộc và không ràng buộc khác nhau và tiến hành lập trình để kiểm thử thuật toán. Đồng thời áp dụng thuật toán đề xuất giải bài toán MTL phân loại ảnh với bộ dữ liệu Multi-Mnist được xây dựng tại [17].

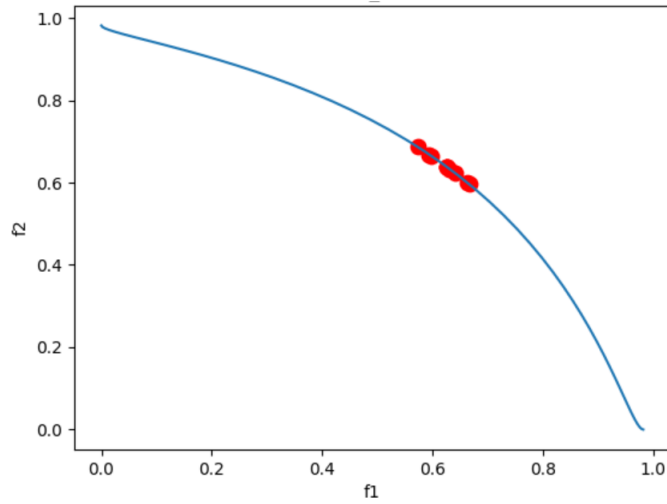
3.1 Một số ví dụ về quy hoạch đa mục tiêu

Bài toán 1 (Andreani et al [1])

Xét bài toán tối ưu đa mục tiêu lồi không ràng buộc:

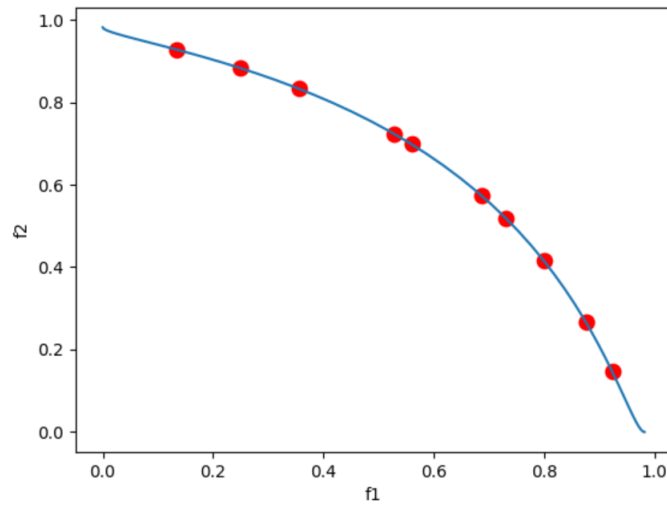
$$\begin{aligned} \text{Min } F(x) &= \left\{ \frac{1}{25}x_1^2 + \frac{1}{100} \left(x_2 - \frac{9}{2}\right)^2, \frac{1}{25}x_2^2 + \frac{1}{100} \left(x_1 - \frac{9}{2}\right)^2 \right\} \\ \text{v.d.k: } x &\in \mathbb{R}^n \end{aligned}$$

Khi sử dụng Thuật toán 4 để giải Bài toán 1, ta thu được kết quả như sau:



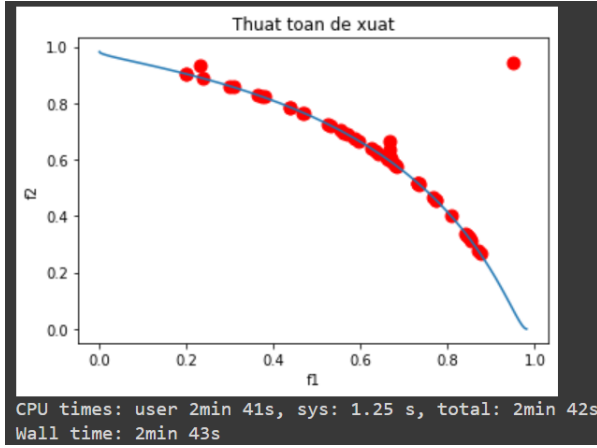
Hình 3.1: Phiên bản 1 của Thuật toán đề xuất giải Bài toán 1

Các nghiệm Pareto thu được tập trung ở một vùng trên Pareto Front. Khi tiến hành thử nghiệm Thuật toán 5 với cùng bài toán, với số điểm khởi tạo 10 điểm ta thu được kết quả như sau:

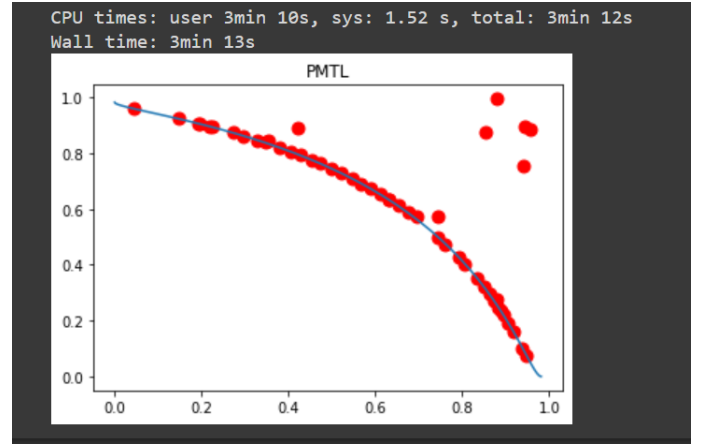


Hình 3.2: Phiên bản 2 của thuật toán đề xuất giải Bài toán 1

Với đường màu xanh Pareto Front như hình, ta thấy các nghiệm tìm được đều thuộc Pareto Front và có trải đều trên toàn mặt. Đây là điểm cải tiến so với phiên bản 1, hay thuật toán MGDA khi nghiệm tìm được chỉ tập trung tại một vùng trên Pareto Front [17]. So sánh thời gian chạy của phiên bản 2 của thuật toán đề xuất với thuật toán PMTL [17], ta thu được kết quả như sau:



Hình 3.3: Thời gian giải bài toán của thuật toán đề xuất



Hình 3.4: Thời gian giải bài toán của thuật toán PMTL

Ta thấy thuật toán đề xuất có thời gian chạy nhanh hơn so với thuật toán PMTL (2 phút 43s so với 3 phút 13s) đồng thời các nghiệm tìm được hội tụ về Pareto Front tốt hơn so với thuật toán PMTL.

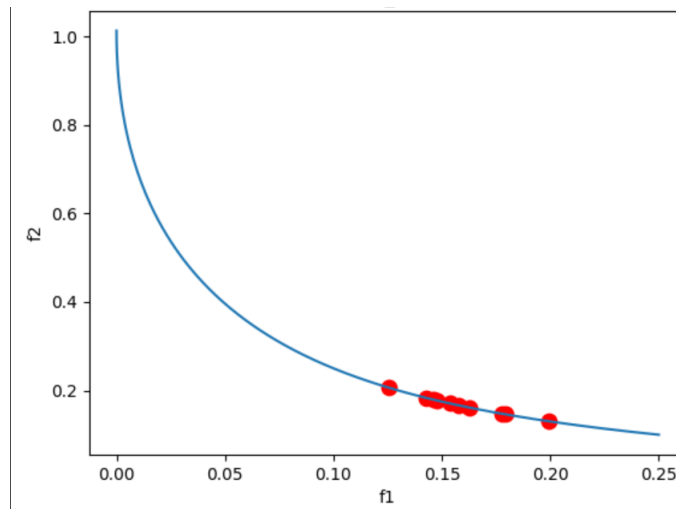
Bài toán 2: Toy example (Lin et al [17])

Xét bài toán tối ưu đa mục tiêu không lỗi không ràng buộc:

$$\text{Min } F(x) = \left\{ 1 - \exp^{-\sum_{i=1}^d (x_i - \frac{1}{d})^2}, 1 - \exp^{-\sum_{i=1}^d (x_i + \frac{1}{d})^2} \right\}$$

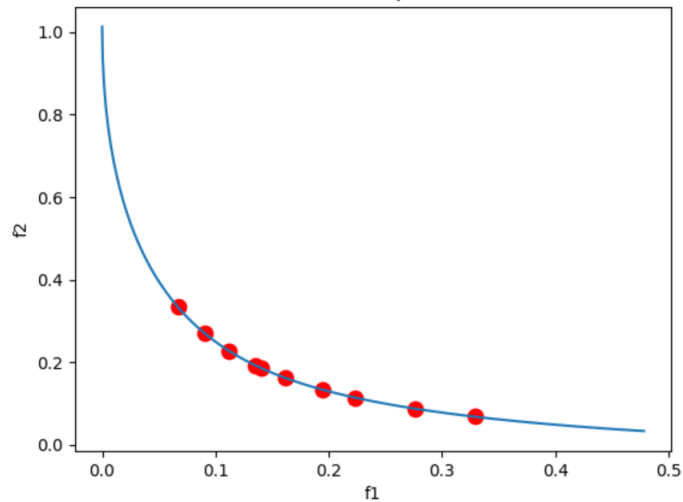
v.đ.k: $x \in \mathbb{R}^2$

Tiến hành thử nghiệm với $d = 20$. Khi sử dụng Thuật toán 4 giải Bài toán 2, ta thu được kết quả như sau:



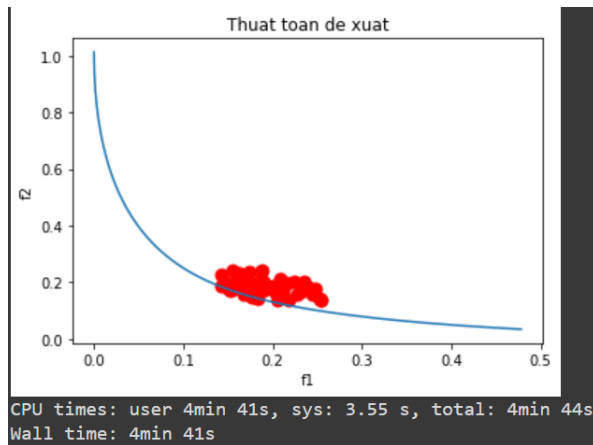
Hình 3.5: Phiên bản 1 của Thuật toán đề xuất giải Bài toán 2

Các nghiệm Pareto thu được tập trung ở một vùng trên Pareto Front. Kết quả thử nghiệm thuật toán 5 với 10 điểm khởi tạo ngẫu nhiên cho kết quả nghiệm Pareto như sau:

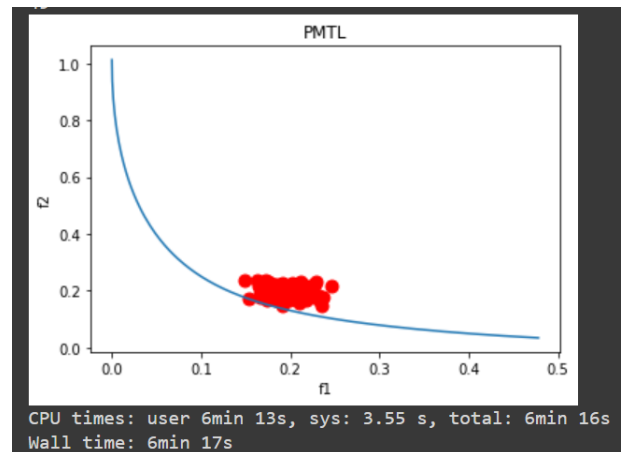


Hình 3.6: Thuật toán đề xuất cho Bài toán 2

Với đường màu xanh Pareto Front như hình, ta thấy các nghiệm tìm được đều thuộc Pareto Front và có trải đều trên toàn mặt. So sánh thời gian chạy của phiên bản 2 của thuật toán đề xuất với thuật toán PMTL [17], ta thu được kết quả như sau:



Hình 3.7: Thời gian giải bài toán của thuật toán đề xuất



Hình 3.8: Thời gian giải bài toán của thuật toán PMTL

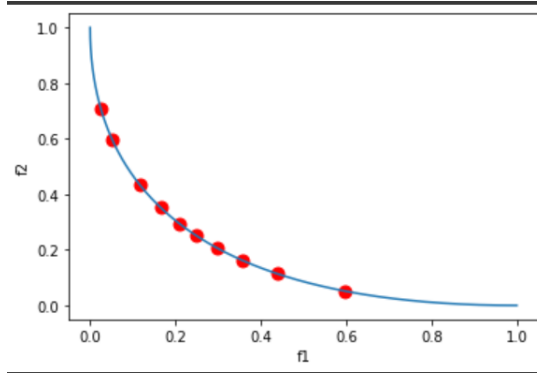
Ta thấy thuật toán đề xuất có thời gian chạy nhanh hơn so với thuật toán PMTL (4 phút 41s so với 6 phút 17s) đồng thời các nghiệm tìm được hội tụ về Pareto Front tốt hơn so với thuật toán PMTL.

Bài toán 3: Xét bài toán tối ưu đa mục tiêu có ràng buộc khoảng như sau:

$$\text{Min } F(x) = \left\{ \frac{x_1^2 + x_2^2}{50}, \frac{(x_1 - 5)^2 + (x_2 - 5)^2}{50} \right\}$$

v.đ.k: $x_i \in [0, 5], i = 1, 2$

Kết quả thử nghiệm thuật toán 7 với 10 điểm khởi tạo ngẫu nhiên cho kết quả nghiệm Pareto tìm được như sau:



Hình 3.9: Thuật toán đề xuất cho Bài toán 3

Với đường màu xanh Pareto Front như hình, ta thấy các nghiệm tìm được đều thuộc Pareto Front và có trải đều trên toàn mặt.

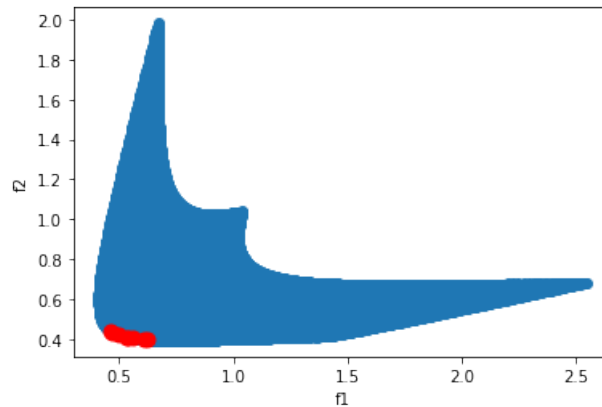
Bài toán 4. Xét bài toán tối ưu không lồi có ràng buộc sau:

$$\text{Min } F(x) = \left\{ \frac{x_1^2 + x_2^2 + 3}{1 + 2x_1 + 8x_2}, \frac{x_2^2 + x_1^2 + 3}{1 + 2x_2 + 8x_1} \right\}$$

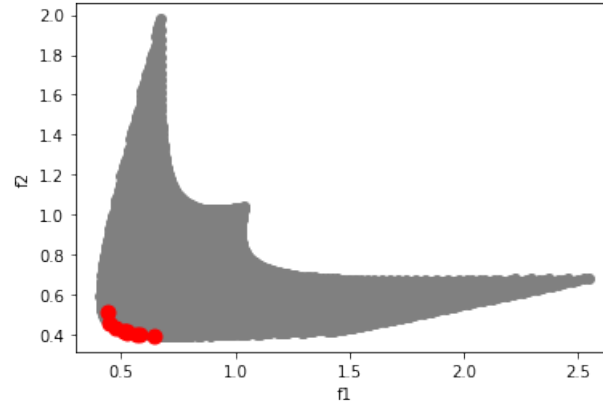
v.đ.k $x \in U,$

trong đó $U = \{x = (x_1, x_2)^\top \in \mathbb{R}^2 | g_1(x) = -x_1^2 - 2x_1x_2 \leq -4; x_1, x_2 \geq 0\}.$

Tiến hành thử nghiệm thuật toán 6 với 20 điểm khởi tạo, ta thu được kết quả như sau:



Ta thấy các nghiệm tìm được tập trung tại 1 vùng trên Pareto Front. Tiến hành thử nghiệm thuật toán 7. Kết quả thử nghiệm phiên bản 2 của thuật toán đề xuất với 20 điểm khởi tạo ngẫu nhiên cho kết quả nghiệm Pareto tìm được như sau:



Hình 3.10: Thuật toán đề xuất cho Bài toán 4

Với đường màu xanh Pareto Front như hình, ta thấy hầu hết các nghiệm tìm được đều thuộc Pareto Front và có trải đều trên toàn mặt.

Bài toán 5

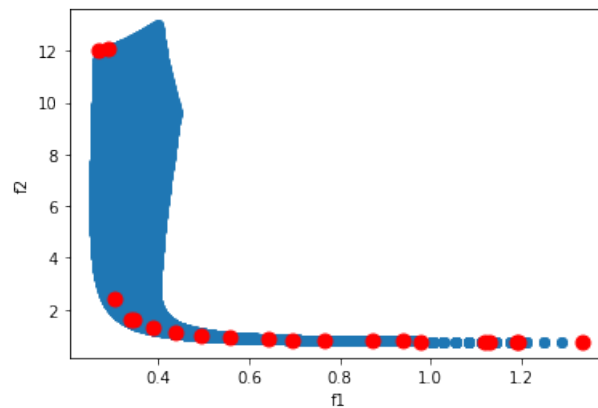
Xét bài toán tối ưu đa mục tiêu có ràng buộc với hàm mục tiêu phân thức như sau:

$$\text{Min } F(x) = \left\{ \frac{x_1 + 1}{-x_1^2 + 3x_1 - x_2^2 + 3x_2 + 3.50}, \frac{x_1^2 - 2x_1 + x_2^2 - 8x_2 + 20.00}{x_2} \right\}$$

v.đ.k $x_1, x_2 \in U$,

Trong đó $U = \{2x_1 + x_2 \leq 6, 3x_1 + x_2 \leq 8, x_1 - x_2 \leq 1, x_1, x_2 \geq 1\}$

Kết quả thử nghiệm thuật toán 7 với 20 điểm khởi tạo ngẫu nhiên cho kết quả nghiệm Pareto tìm được như sau:



Hình 3.11: Thuật toán đề xuất cho Bài toán 5

Với đường màu xanh Pareto Front như hình, ta thấy các nghiệm tìm được hầu hết đều thuộc Pareto Front, các điểm khác sẽ gần tiến tới Pareto Front.

Nhận xét: Ta có thể thấy thuật toán đề xuất hoạt động tốt, cho được nghiệm tối ưu Pareto trong các bài toán thử nghiệm.

3.2 Ví dụ về bài toán học đa nhiệm

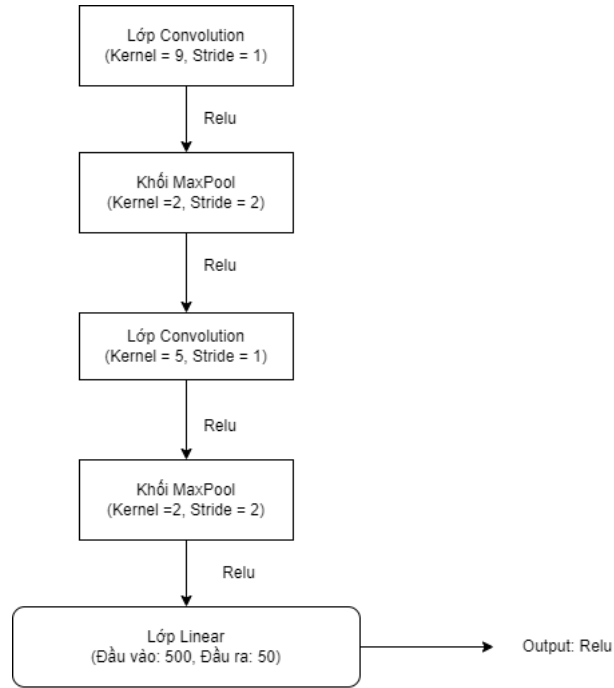
Trong phần này, chúng tôi sẽ tiến hành cài đặt thuật toán đề xuất và đánh giá độ hiệu quả của thuật toán với bài toán học hai tác vụ phân loại ảnh với bộ dữ liệu *Multi-MNIST* được cung cấp tại [17]. Đồng thời, ta sẽ đi so sánh độ hiệu quả của thuật toán đề xuất với thuật toán gốc của Lin et al [17].

Multi-MNIST là bộ dữ liệu được xây dựng dựa trên việc ghép 2 chữ số trong bộ dữ liệu MNIST [15] vào 1 ảnh. Bài toán MTL phân loại ảnh sẽ học đồng thời hai hàm mất mát biểu thị cho việc dự đoán 2 chữ số, 1 ở góc trên bên trái và 1 ở góc dưới bên phải. Hình ảnh minh họa cho bộ dữ liệu ở dưới đây.



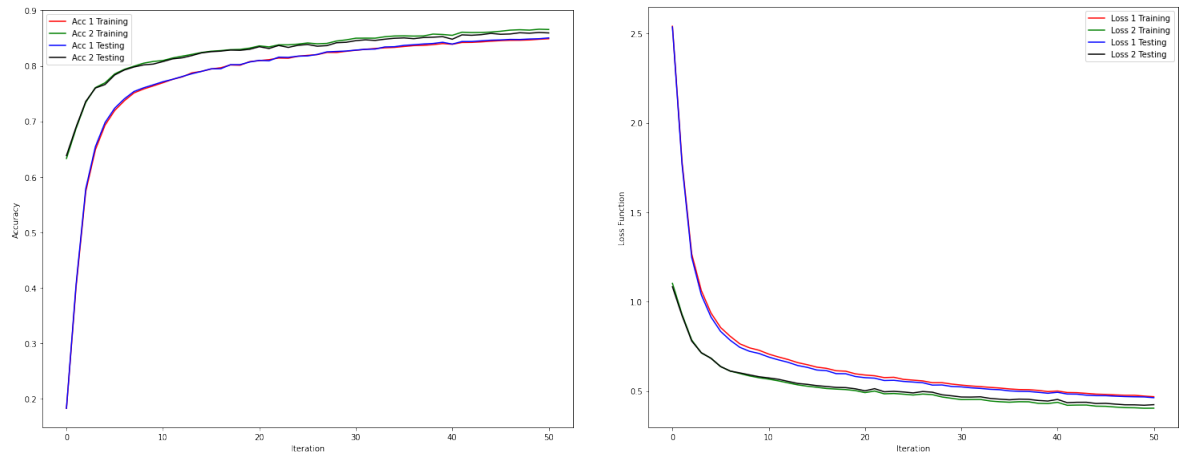
Hình 3.12: Bộ dữ liệu Multi-MNIST

chúng tôi tiến hành xây dựng model LeNet tương tự như trong bài [16] gồm các lớp như sau:



Hình 3.13: Model xây dựng

Tiến hành quá trình huấn luyện với 5 vector tham chiếu $[\cos(\frac{k\pi}{2K}), \sin(\frac{k\pi}{2K})]$, $k = 0, 1, 2, 3, 4$ với mỗi lần 50 epoch. Ta minh họa kết quả với vector tham chiếu $u_1 = (1, 0)$ như sau:

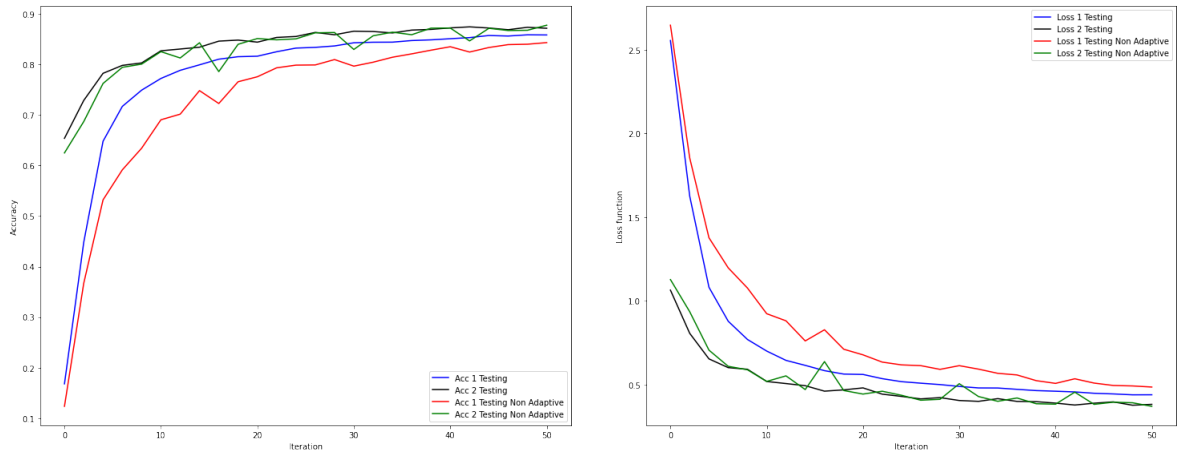


Hình 3.14: Minh họa kết quả thử nghiệm

Kết quả thực nghiệm cho thấy 2 tác vụ phân loại ảnh đều được học đồng thời và tương đối hiệu quả. Bảng so sánh giá trị của Accuracy trên tập Test của thuật toán đề xuất với PMTL sẽ được minh họa dưới đây:

Vector tham chiếu	Accuracy Task 1 Proposal	Accuracy Task 2 Proposal	Accuracy Task 1 PMTL	Accuracy Task 2 PMTL
u_1	0.85 ± 0.05	0.87 ± 0.05	0.84 ± 0.05	0.87 ± 0.05
u_2	0.88 ± 0.05	0.88 ± 0.05	0.88 ± 0.05	0.89 ± 0.05
u_3	0.90 ± 0.05	0.88 ± 0.05	0.91 ± 0.05	0.88 ± 0.05
u_4	0.90 ± 0.05	0.88 ± 0.05	0.91 ± 0.05	0.87 ± 0.05
u_5	0.91 ± 0.05	0.84 ± 0.05	0.92 ± 0.05	0.86 ± 0.05

Ta có thể thấy độ hiệu quả của thuật toán đề xuất khá tương đồng với thuật toán PMTL. Hình ảnh so sánh quá trình huấn luyện của thuật toán đề xuất (đường màu xanh dương cho tác vụ 1 và đường màu đen cho tác vụ 2) và thuật toán PMTL (đường màu đỏ cho tác vụ 1 và đường màu xanh lá cho tác vụ 2) trong trường hợp vector tham chiếu u_1 như sau:



Hình 3.15: So sánh thuật toán đề xuất và PMTL

Ta có thể thấy quá trình huấn luyện với cơ bước Adaptive cho kết quả hàm mất mát giảm đều hơn và do vậy độ chính xác cũng tăng đều liên tục.

Kết luận và hướng phát triển đề tài

Đề án này đã trình bày được các kiến thức liên quan đến phương pháp giải bài toán quy hoạch đa mục tiêu dựa trên phương pháp hướng giảm. Đồng thời, chúng tôi đề xuất một thuật toán hướng giảm với cỡ bước tự thích để giải các bài toán quy hoạch đa mục tiêu có ràng buộc và không ràng buộc cũng như áp dụng thuật toán vào giải một bài toán MTL phân loại ảnh. Một số kết quả mà chúng tôi đạt được trong quá trình hoàn thiện đề án này như sau:

- Trình bày nội dung lý thuyết về các thuật toán hướng giảm trong việc giải bài toán quy hoạch đa mục tiêu.
- Đề xuất thuật toán hướng giảm sâu nhất đa mục tiêu kết hợp cỡ bước tự thích nghi và đặt thêm các ràng buộc bất đẳng thức tuyến tính để trải đều nghiệm ra toàn bộ Pareto Front.
- Trình bày cách chứng minh độ hội tụ của thuật toán.
- Code thuật toán đề xuất và đạt được kết quả như kỳ vọng.
- Thuật toán đề xuất cho bài toán MTL đạt được kết quả tương đương thuật toán PMTL.

Hướng phát triển đề tài

Trong thời gian tới, chúng tôi sẽ tiếp tục nghiên cứu đề tài và tiến hành thực hiện các ý tưởng sau:

- Xây dựng thêm các ví dụ tối ưu đa mục tiêu lồi, không lồi, có ràng buộc và không ràng buộc để thực nghiệm thuật toán đề xuất.

-
- Áp dụng thuật toán đề xuất giải bài toán MTL cho nhiều bộ dữ liệu khác nhau như Multi-Fashion, Fashion-Mnist ([17]), ...
 - Nghiên cứu các loại cỡ bước khác nhau và so sánh độ hiệu quả với thuật toán đề xuất.

Tài liệu tham khảo

- [1] Roberto Andreani et al. “Bilevel optimization with a multiobjective problem in the lower level”. In: *Numerical Algorithms* 81.3 (2019), pp. 915–946.
- [2] Heinz H Bauschke, Patrick L Combettes, et al. *Convex analysis and monotone operator theory in Hilbert spaces*. Vol. 408. Springer, 2011.
- [3] Wei Bian et al. “Neural network for nonsmooth pseudoconvex optimization with general convex constraints”. In: *Neural Networks* 101 (2018), pp. 1–14.
- [4] SP Boyd and L Vandenberghe. *edition 7. Convex optimization*. 2009.
- [5] Volkan Cevher, Stephen Becker, and Mark Schmidt. “Convex optimization for big data: Scalable, randomized, and parallel algorithms for big data analytics”. In: *IEEE Signal Processing Magazine* 31.5 (2014), pp. 32–43.
- [6] John E Dennis Jr and Robert B Schnabel. *Numerical methods for unconstrained optimization and nonlinear equations*. SIAM, 1996.
- [7] Jean-Antoine Désidéri. “Multiple-gradient descent algorithm (MGDA) for multiobjective optimization”. In: *Comptes Rendus Mathématique* 350.5-6 (2012), pp. 313–318.
- [8] Matthias Ehrgott. *Multicriteria optimization*. Vol. 491. Springer Science & Business Media, 2005.
- [9] OP Ferreira and WS Sosa. “On the Frank–Wolfe algorithm for non-compact constrained optimization problems”. In: *Optimization* 71.1 (2022), pp. 197–211.
- [10] Jörg Fliege and Benar Fux Svaiter. “Steepest descent methods for multicriteria optimization”. In: *Mathematical methods of operations research* 51 (2000), pp. 479–494.
- [11] Yaohua Hu, Jiawen Li, and Carisa Kwok Wai Yu. “Convergence rates of subgradient methods for quasi-convex optimization problems”. In: *Computational Optimization and Applications* 77 (2020), pp. 183–212.
- [12] Krzysztof C Kiwiel. “Convergence and efficiency of subgradient methods for quasiconvex minimization”. In: *Mathematical programming* 90 (2001), pp. 1–25.
- [13] IV Konnov. “Simplified versions of the conditional gradient method”. In: *Optimization* 67.12 (2018), pp. 2275–2290.

-
- [14] Guanghui Lan. *First-order and stochastic optimization methods for machine learning*. Vol. 1. Springer, 2020.
 - [15] Yann LeCun, Corinna Cortes, and CJ Burges. “MNIST handwritten digit database”. In: *ATT Labs [Online]*. Available: <http://yann.lecun.com/exdb/mnist> 2 (2010).
 - [16] Yann LeCun et al. “Gradient-based learning applied to document recognition”. In: *Proceedings of the IEEE* 86.11 (1998), pp. 2278–2324.
 - [17] Xi Lin et al. “Pareto multi-task learning”. In: *Advances in neural information processing systems* 32 (2019).
 - [18] Hai-Lin Liu, Fangqing Gu, and Qingfu Zhang. “Decomposition of a multiobjective optimization problem into a number of simple multiobjective subproblems”. In: *IEEE transactions on evolutionary computation* 18.3 (2013), pp. 450–455.
 - [19] Na Liu, Jun Wang, and Sitian Qin. “A one-layer recurrent neural network for nonsmooth pseudoconvex optimization with quasiconvex inequality and affine equality constraints”. In: *Neural Networks* 147 (2022), pp. 1–9.
 - [20] Olvi L Mangasarian. *Nonlinear programming*. SIAM, 1994.
 - [21] Yurii Nesterov. *Introductory lectures on convex optimization: A basic course*. Vol. 87. Springer Science & Business Media, 2003.
 - [22] Ozan Sener and Vladlen Koltun. “Multi-task learning as multi-objective optimization”. In: *Advances in neural information processing systems* 31 (2018).
 - [23] Carisa Kwok Wai Yu et al. “Abstract convergence theorem for quasi-convex optimization problems with applications”. In: *Optimization* (2018).