

---

# Appendix for Multi-Task Learning with User Preferences: Gradient Descent with Controlled Ascent in Pareto Optimization

---

Debabrata Mahapatra<sup>1</sup> Vaibhav Rajan<sup>1</sup>

Equations (1) to (36) and figures 1 to 8 are from the main paper.

## A. Proofs

**Theorem 1.** *If all the objective functions are differentiable, then for any direction  $d \in \mathbb{R}^n$  with  $d^T d_{\text{bal}} > 0$ , there exists a step size  $\eta_0 > 0$ , such that*

$$\mu_r(l(\theta - \eta d)) \leq \mu_r(l(\theta)), \quad \forall \eta \in [0, \eta_0]. \quad (37)$$

*Proof.* Taylor's expansion of the differentiable objective function  $l_j$  can be written with Peano's form of remainder as

$$l_j(\theta - \eta d) = l_j(\theta) - \eta d^T g_j + o(\eta), \quad (38)$$

where  $g_j = \partial l_j / \partial \theta$ , and the asymptotic notation little- $o(\eta)$  represents a function that approaches 0 faster than  $\eta$ . In particular, for every  $\epsilon > 0$ , there exists an  $\eta_0 > 0$  such that

$$\left| \frac{o(\eta)}{\eta} \right| < \epsilon, \quad \text{for } |\eta| < \eta_0. \quad (39)$$

In this proof, we will be using two additional properties of  $o(\eta)$ , viz.  $c \times o(\eta) = o(\eta)$  for any constant  $c$ , and  $o(\eta) + o(\eta) = o(\eta)$ .

For brevity of notations, we expediently denote  $l_j(\theta - \eta d)$  as  $l_j^\eta$ , while  $l_j(\theta)$  as simply  $l_j$ . The non-uniformity of  $l(\theta)$  can now be written as

$$\begin{aligned} \mu_r(l) &= \frac{\sum_{j=1}^m r_j l_j \log(r_j l_j)}{L_r} + \log\left(\frac{m}{L_r}\right) \\ &= A + B, \end{aligned} \quad (40)$$

$$\text{where } L_r = \sum_{j=1}^m r_j l_j. \quad (41)$$

Similarly, the non-uniformity of  $l(\theta - \eta d)$  can be written as

$$\begin{aligned} \mu_r(l^\eta) &= \frac{\sum_{j=1}^m r_j l_j^\eta \log(r_j l_j^\eta)}{L_r^\eta} + \log\left(\frac{m}{L_r^\eta}\right) \\ &= A^\eta + B^\eta, \end{aligned} \quad (42)$$

$$\text{where } L_r^\eta = \sum_{j=1}^m r_j l_j^\eta. \quad (43)$$

Applying the Taylor's expansion (38) to  $L_r^\eta$  and using (41) one can write

$$L_r^\eta = L_r - \eta d^T \sum_{j=1}^m r_j g_j + o(\eta). \quad (44)$$

In the above, we used the property  $c \times o(\eta) = o(\eta)$ . Using  $L_r^\eta$  of (44), the second term  $B^\eta$  in (42) can be analysed as

$$-B^\eta = \log\left(L_r - \eta d^T \sum_{j=1}^m r_j g_j + o(\eta)\right) - \log(m) \quad (45)$$

$$\geq \log(L_r) - \frac{\eta d^T \sum_{j=1}^m r_j g_j - o(\eta)}{L_r^\eta} - \log(m) \quad (46)$$

$$\Rightarrow B^\eta \leq B + \frac{\eta d^T \sum_{j=1}^m r_j g_j}{L_r^\eta} - \frac{o(\eta)}{L_r^\eta}. \quad (47)$$

For the inequality in (46), we use the following property

$$\log(a + b) \geq \log(a) + \frac{b}{a + b}, \quad \text{for } a > \max(0, -b), \quad (48)$$

where  $a = L_r$ ,  $b = -\eta d^T \sum_{j=1}^m r_j g_j + o(\eta)$ , and  $a + b = L_r^\eta$ . In this case, the usage of (48) is valid, because the preference vector  $r$  and all the objective functions are always non-negative.

Next, before we simplify the first term  $A^\eta$  in (42), let us express  $\log(l_j^\eta)$  as

$$\log(l_j - \eta d^T g_j + o(\eta)) \leq \log(l_j) - \frac{\eta d^T g_j}{l_j} + o(\eta). \quad (49)$$

The above inequality is due to another property of the log function,

$$\log(a + b) \leq \log(a) + \frac{b}{a}, \quad \text{for } a > \max(0, -b), \quad (50)$$

where  $a = l_j$  and  $b = -\eta d^T g_j + o(\eta)$ ; since,  $l_j$  is not a function of  $\eta$ , we can write  $o(\eta)/l_j = o(\eta)$ . Using (49), we can now write the first term  $A^\eta$  as

$$\begin{aligned} A^\eta &\leq \frac{\sum_{j=1}^m r_j (l_j - \eta d^T g_j) \log(r_j l_j)}{L_r^\eta} \\ &\quad - \frac{\eta d^T \sum_{j=1}^m r_j g_j}{L_r^\eta} + \frac{o(\eta)}{L_r^\eta}, \end{aligned} \quad (51)$$

where all the second or higher order terms of  $\eta$  are subsumed in  $o(\eta)$ .

Next, we upper bound the non-uniformity of  $l(\theta - \eta d)$  by adding (47) and (51)

$$\mu_r(l^n) \leq \frac{L_r}{L_r^\eta} A + B + \frac{o(\eta)}{L_r^\eta} - \frac{\eta d^T \sum_{j=1}^m r_j g_j \log(r_j l_j)}{L_r^\eta}. \quad (52)$$

Note that the  $o(\eta)$  in (47) may not cancel out with the one in (51), so we keep it in equation (52). The first term of (52) can be further simplified by using (44)

$$\frac{L_r}{L_r^\eta} A = A + \frac{\eta d^T \sum_{j=1}^m r_j g_j A}{L_r^\eta} + \frac{o(\eta)}{L_r^\eta}. \quad (53)$$

Finally, we can relate the non-uniformity of  $l$  and  $l^n$  by using (53) in (52) as

$$\mu_r(l^n) \leq \mu_r(l) + \frac{o(\eta)}{L_r^\eta} + \frac{\eta d^T \sum_{j=1}^m r_j g_j (A - \log(r_j l_j))}{L_r^\eta}. \quad (54)$$

By adding and subtracting  $B$  with  $A - \log(r_j l_j)$  we obtain

$$A + B - (\log(r_j l_j) + B) = \mu_r(l) - \log(m \hat{l}_j). \quad (55)$$

Using the expression for  $d_{bal}$  in (12) along with (54) and (55), one can write

$$\mu_r(l) - \mu_r(l^n) \geq \frac{\eta d^T d_{bal} - o(\eta)}{L_r^\eta}. \quad (56)$$

We know that  $d^T d_{bal} > 0$  from the statement of the theorem 1. As the final step, we use the property of  $o(\eta)$  as mentioned in (39) by treating  $d^T d_{bal}$  as  $\epsilon$ , and conclude that there exists a step size  $\eta_0 > 0$  such that

$$\left| \frac{o(\eta)}{\eta} \right| < d^T d_{bal}, \forall \eta \in [0, \eta_0], \quad (57)$$

and hence  $\mu_r(l) - \mu_r(l^n) \geq 0$ ; the equality holds when  $d_{bal} = 0 \in \mathbb{R}^n$ .  $\square$

**Claim 1.** *The adjustment vector  $a$  is perpendicular to the multi-objective vector  $l$  in the objective space  $\mathbb{R}^m$*

$$a^T l = 0. \quad (58)$$

*Proof.* We expand (58) as

$$\begin{aligned} a^T l &= \sum_{j=1}^m r_j \left( \log(m \hat{l}_j) - \mu_r(l) \right) \times l_j \\ &= \sum_{j=1}^m r_j l_j \left( (1 - \hat{l}_j) \log(\hat{l}_j) - \sum_{j' \neq j} \hat{l}_{j'} \log(\hat{l}_{j'}) \right). \end{aligned}$$

We use the fact that  $\sum_{j=1}^m \hat{l}_j = 1$ , and further expand as

$$\begin{aligned} a^T l &= \sum_{j=1}^m r_j l_j \left( \sum_{j' \neq j} \hat{l}_{j'} \log(\hat{l}_j) - \sum_{j' \neq j} \hat{l}_{j'} \log(\hat{l}_{j'}) \right) \\ &= \sum_{j=1}^m r_j l_j \left( \sum_{j' \neq j} \hat{l}_{j'} \log\left(\frac{\hat{l}_j}{\hat{l}_{j'}}\right) \right). \end{aligned}$$

In the inner summation we can now add the term for  $j = j'$  as  $\log\left(\frac{\hat{l}_j}{\hat{l}_{j'}}\right) = \log(1) = 0$  and write the above expression as

$$a^T l = \frac{1}{\sum_{j=1}^m r_j l_j} \sum_{j=1}^m \sum_{j'=1}^m r_j l_j r_{j'} l_{j'} \log\left(\frac{\hat{l}_j}{\hat{l}_{j'}}\right)$$

The double summation in the numerator can be written as the inner product of a symmetric and a skew-symmetric matrix which is equal to 0.  $\square$

**Lemma 1.** *If  $\mu_r(l^t) = 0$ , then the non-dominating direction  $d_{nd}$  becomes a descent direction, i.e.*

$$d_{nd}^T g_j \geq 0, \quad \forall j \in [m]. \quad (59)$$

*Proof.* If  $l^t$  lies in the  $r^{-1}$  ray, i.e.  $\mu_r(l^t) = 0$ , then the adjustment vector  $a$  is zero. This means  $J$  is empty (see (17)). As a result,  $\mathbb{1}_J = 0$  and all the constraints of (24b) becomes

$$c_j^T \beta^* = g_j^T G \beta^* = g_j^T d_{nd} \geq 0, \quad \forall j \in [m]. \quad (60)$$

Thus,  $d_{nd}$  becomes a descent direction.  $\square$

**Lemma 2.** *Let  $\gamma^* = a^T C \beta^*$  be the maximum value of the LP problem (24) when  $\mu_r(l^t) > 0$ .*

$$\text{If } \gamma^* > 0, \text{ then } d_{nd}^T d_{bal} > 0. \quad (61)$$

$$\text{If } \gamma^* \leq 0, \text{ then } d_{nd}^T g_j \geq 0, \forall j \in [m]. \quad (62)$$

*Proof.* Proving (61), i.e. when  $\gamma^* > 0$ , is a matter of simply rewriting its formula,

$$\begin{aligned} \gamma^* &= a^T C \beta^* \\ \implies \gamma^* &= a^T G^T G \beta^* \\ \implies \gamma^* &= d_{bal}^T d_{nd}. \end{aligned}$$

As a result,  $\gamma^* > 0 \implies d_{bal}^T d_{nd} > 0$ .

Now, let us analyze the case  $\gamma^* \leq 0$ . A negative value of  $d_{bal}^T d_{nd}$  means that there is no gradient  $g_j$ ,  $j \in [m]$ , for which  $d_{bal}^T g_j > 0$ . As a result,  $J$  is empty and  $\mathbb{1}_J = 0$ . This is similar to the case of  $\mu_r(l^t) = 0$  and (60). So  $d_{nd}$  becomes a descent direction.  $\square$

**Lemma 3.** *The set of dominating multi-objective points is a subset of the admissible set*

$$\mathcal{V}_{\leq l^t} \subset \mathcal{A}_{l^t}^r. \quad (63)$$

*Proof.*

$$\begin{aligned} l &\in \mathcal{V}_{\leq l^t} \\ \implies l &\leq l^t \\ \implies r \odot l &\leq r \odot l^t \\ \implies r \odot l &\leq \lambda^t \mathbf{1} \\ \implies l &\leq \check{l}^t \\ \therefore l &\in \mathcal{A}_{l^t}^r. \end{aligned}$$

□

**Theorem 2.** *There exists a step size  $\eta_0 > 0$ , such that for every  $\eta \in [0, \eta_0]$ , the multi-objective value of new solution point  $\theta^{t+1} = \theta^t - \eta d_{nd}$  lies in the  $t^{\text{th}}$  admissible set*

$$l(\theta^{t+1}) \in \mathcal{A}_{l^t}^r. \quad (64)$$

*Proof.* From Lemma 1 and 2, we know that  $d_{nd}$  can either be a descent direction (62, 60), or have positive angle with the balancing direction (61). So we divide our analysis into these cases.

When  $d_{nd}$  is a descent direction,  $g_j^T d_{nd} \geq 0$  for all  $j \in [m]$ . As a result, by using the Taylor's expansion with Peano form of remainder (38) and the property of little- $o$  notation (39), one can deduce that for every  $j \in [m]$  there exists a step size  $\eta_{0j} > 0$  such that

$$l(\theta^t - \eta d_{nd}) = l_j^{t+1} \leq l_j^t$$

for all  $\eta \in [0, \eta_{0j}]$ . If we choose  $\eta_0 = \min_j \{\eta_{0j}\}$ , then for all  $\eta \in [0, \eta_0]$

$$\begin{aligned} l(\theta^t - \eta d_{nd}) &= l_j^{t+1} \leq l_j^t, \quad \forall j \in [m] \\ \implies l^{t+1} &\in \mathcal{V}_{\leq l^t} \end{aligned} \quad (65)$$

From Lemma 3 and (65) we conclude that  $l^{t+1} \in \mathcal{A}_{l^t}^r$  for all  $\eta \in [0, \eta_0]$ .

Now let us analyse the case when  $d_{nd}$  is not a descent direction. Let  $J^+ = \{j \mid g_j^T d_{nd} \geq 0\}$  be the index set for descending objectives and  $J^- = [m] - J$  for ascending ones. So, there exists an  $\eta_{0j} > 0$  for all  $j \in J^+$  such that

$$l(\theta^t - \eta d_{nd}) = l_j^{t+1} \leq l_j^t$$

for all  $\eta \in [0, \eta_{0j}]$ . Let  $\eta_0^{J^+} = \min_j \{\eta_{0j} \mid j \in J^+\}$ , and  $\tilde{\eta}_0 = \min\{\eta_0^{J^+}, \eta_0^{J^-}\}$ . Then for all  $\eta \in [0, \tilde{\eta}_0]$ , and

$$l^{t+1} = l(\theta^t - \eta d_{nd})$$

$$\begin{aligned} \mu_r(l^{t+1}) &\leq \mu_r(l^t), \\ \text{and } l_j^{t+1} &\leq l_j^t, \quad \forall j \in J^+. \\ \implies r_j l_j^{t+1} &\leq r_j l_j^t \leq \lambda^t \\ \implies l_j^{t+1} &\leq \check{l}_j^t, \quad \forall j \in J^+ \end{aligned}$$

Note that the constraints in (24c) ensures that  $J^* \subset J^+$ . If all the other objectives in  $J^-$  also satisfy

$$r_j l_j^{t+1} \leq \lambda^t, \quad \forall \eta \in [0, \tilde{\eta}_0]$$

then  $\tilde{\eta}_0$  can be used as the step size as it is. If this is not the case, i.e. there exists some  $j' \in J^-$  such that

$$r_{j'} l_{j'}(\theta^t - \tilde{\eta}_0 d_{nd}) > \lambda^t,$$

then continuity of the objective functions ensures that there must exists some  $\eta_{0j'} < \tilde{\eta}_0$  such that

$$r_{j'} l_{j'}^{t+1} \leq \lambda^t, \quad \forall \eta \in [0, \eta_{0j'}].$$

Moreover  $\eta_{0j'} > 0$ , because  $r_{j'} l_{j'}^t = r_{j'} l(\theta^t) < \lambda^t$ . So choosing  $\eta_0 = \min_{j'} \{\eta_{0j'}\}$  we finally get

$$\begin{aligned} r \odot l(\theta^t - \eta d_{nd}) &= r \odot l^{t+1} \leq \lambda^t r \\ \implies l^{t+1} &\leq \check{l}^t \\ \therefore l^{t+1} &\in \mathcal{A}_{l^t}^r \end{aligned}$$

for all  $\eta \in [0, \eta_0]$ . □

**Claim 2.** *Let  $\theta^* \in \mathcal{P}$  be a regular Pareto optimal solution. If the set of exact Pareto optimal solutions  $\mathcal{P}_r$  is nonempty, then the non-dominating direction  $d_{nd} = G\beta^*$  found by the LP (24) is  $0 \in \mathbb{R}^n$  if and only if  $\theta^* \in \mathcal{P}_r$ .*

*Proof.* If  $\theta^* \in \mathcal{P}_r$ , then  $\mu_r(l^*) = \mu_r(l(\theta^*)) = 0$ , hence the adjustments  $a$  are also 0. From Lemma 1 we know that  $d_{nd}$  is a descent direction, i.e.

$$g_j^T d_{nd} \geq 0, \quad \forall j \in [m]. \quad (66)$$

Let us say  $d_{nd} \neq 0$ . Then in the next iteration we can reach to a point in  $\mathcal{V}_{\leq l^*}$ , which violates our assumption that  $l^*$  is Pareto Optimal. So  $d_{nd} = 0$ .

If  $\theta^* \notin \mathcal{P}_r$ , then  $\mu_r(l^*) \neq 0$ , hence the adjustments  $a$  are also nonzero. As  $\theta^* \in \mathcal{P}$  is a Pareto optimal solution,  $0 \in \mathcal{CH}_{\theta^*}$ , due to the criticality condition. As a result, the convex cone of  $\{g_1, \dots, g_m\}$  forms a linear subspace spanned by  $g_j$ s. The balancing direction  $d_{bal} = Ga$  also belongs to this linear subspace, and hence  $d_{bal} \in \mathcal{CH}_{\theta^*}$ . Further, due to the regularity of  $\theta^*$ , i.e. rank of  $G$  is  $m - 1$ , we know that  $d_{bal}$  is not  $0 \in \mathbb{R}^n$ . Therefore

$$\max_{d \in \mathcal{CH}_{\theta^*}} d_{bal}^T d = d_{bal}^T d_{nd} > 0, \quad (67)$$

which means  $d_{nd} \neq 0$ .

Note that even if  $\theta^*$  is not regular, i.e. rank of  $G$  is less than  $m - 1$ , this claim can be true as long as the adjustment vector  $a$  is not in the Null space of  $G$ .  $\square$

## B. Extensions and Variants of EPO Search

### B.1. Restricting Trajectory of Pure Descent

In practice, we choose a fixed step size for every iteration. In such a scenario, the indicator  $\mathbb{1}_{\mu_r^t}$  in (24a) for non-zero  $\mu_r(l^t)$  becomes a strict condition to enter into the pure descent mode. Although eventually the iterations converge, the multi-objective values could fluctuate around the  $r^{-1}$  ray before reaching the EPO. This is because, the fixed step size may not allow the multi-objective value to reach exactly onto the  $r^{-1}$  ray, i.e.  $\mu_r(l^t) = 0$ , and hence perform pure descent mode.

One approach to address this is to relax the indicator's condition from  $\mu_r^t > 0$  to  $\mu_r^t > \epsilon$  for some small positive  $\epsilon$ , so that pure descent can occur whenever  $\mu_r^t \leq \epsilon$ . In other words, whenever the iterate  $l^t$  lies inside the cone

$$\mathcal{M}_\epsilon^r = \{l \in \mathbb{R}_+^m \mid \mu_r(l) \leq \epsilon\}, \quad (68)$$

the algorithm performs a pure descent step.

Even after relaxing the indicator condition for practical implementation with fixed step size, the fluctuations near the  $r^{-1}$  ray may not be avoided, as shown in fig. 4a. This is because a descent direction  $d \in \mathcal{CH}_{\theta^*}$  that maximizes  $d^T G = \beta^T G \mathbf{1}$  may not decrease the iterate  $l^t$  along the  $r^{-1}$  ray. To find a descent direction that will take  $r^{-1}$  ray into consideration, we add another constraint in the LP in the pure descent mode

$$\beta^* = \arg \max_{\beta \in S^m} \beta^T C \mathbf{1} \quad (69)$$

$$\text{s.t. } \beta^T c_j \geq 0, \quad \forall j \in [m] \quad (70)$$

$$\beta^T C a \geq \left( \max_j \{c_j^T a\} \right)^-, \quad (71)$$

where  $(x)^- = x$  if  $x < 0$ , and 0 otherwise. The restriction (71) essentially means that we want  $\beta^{*T} C a = d_{nd}^T d_{bal}$  to be non-negative if there exists at least one gradient  $g_j$  making a non-negative angle with  $d_{bal}$ , i.e.  $g_j^T d_{bal} = c_j^T a \geq 0$ . When there is no such  $g_j$ , we want  $d_{nd}^T d_{bal}$  to be at least as high as the  $\max_j \{g_j^T d_{bal}\}$ . The result of applying this restriction is shown in fig. 4b. The experimental setup for this is described in section 5.1.

**Distinction from Pareto MTL.** The above restricted descent approach appears to be similar to Pareto MTL Lin et al. (2019) (described in section 3.2), where in the first phase one finds a solution  $\theta_r^0 \in \Omega_k$  such that  $\theta_r^* \in \Omega_k$ , and in

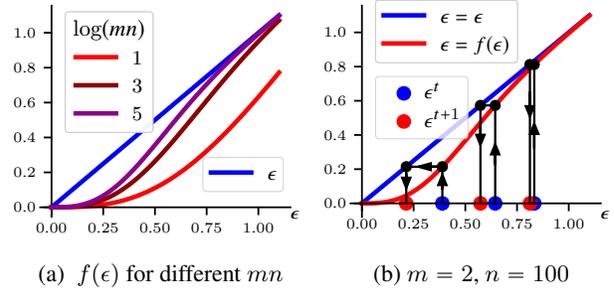


Figure 9. Illustration of annealing schedule for updating the threshold  $\epsilon^t$  when  $\mu_r(l^t) \leq \epsilon^t$  using  $f(\epsilon)$  as defined in (72).

the second phase one does pure descent. Their construction of  $\Omega_k$  is such that  $l(\Omega_k)$  is also a cone.

However, their method does not guarantee that the outcome of second phase  $\theta^*$  also lies in  $\Omega_k$ . Because while descending, the multi-objective value may go outside the cone  $l(\Omega_k)$ . On the other hand, our method guarantees that the multi-objective value of the final solution will be inside the cone  $\mathcal{M}_\epsilon^r$ . Because, if in some iteration  $l^t \notin \mathcal{M}_\epsilon^r$ , then the indicator  $\mathbb{1}_{\mu_r^t}$  is activated to balance the multi-objective values and bring it back inside the cone  $\mathcal{M}_\epsilon^r$  in the subsequent iterations.

Moreover, the angular fineness of their cone  $l(\Omega_k)$ , which dictates the accuracy of the final solution, is dependent on how many reference vectors  $u_k, k = 1, \dots, K$  are used, which increases exponentially with the number of objectives  $m$ . On other hand, the angular fineness of our cone  $\mathcal{M}_\epsilon^r$  can be set by merely choosing a small value of  $\epsilon$ .

### B.2. Annealed EPO Search

We observe that when  $\epsilon = 0$ , the cone  $\mathcal{M}_\epsilon^r$  becomes the  $r^{-1}$  ray itself. But making  $\epsilon = 0$  from the beginning makes it difficult for the iterate  $l^t$  to enter this degenerate cone when using a fixed step size, as discussed in the previous section. However, when the iterate is already inside an  $\epsilon$  neighbourhood of  $l^* = l(\theta_r^*)$  in the objective space, we can decrease the value of  $\epsilon$  to bring it further close to  $l^*$ . Reducing the value of threshold  $\epsilon$  can be performed repeatedly whenever  $l^t \in \mathcal{M}_\epsilon^r$ . This strategic decrease in  $\epsilon$  can be interpreted as an annealing procedure, wherein the cone  $\mathcal{M}_\epsilon^r$  gradually becomes thinner and eventually degenerates to be the  $r^{-1}$  ray. We choose the following annealing schedule

$$f(\epsilon) = \epsilon - \epsilon \exp(-\log(mn)\epsilon^2) \quad (72)$$

to update the threshold as  $\epsilon \leftarrow f(\epsilon)$  whenever the non-uniformity  $\mu_r^t$  of iterate  $l^t$  is less than  $\epsilon$ . The schedule  $f(\epsilon)$  is illustrated in fig. 9.

In the beginning of annealing process,  $\epsilon$  need not be a small value. In other words, the permissible region for pure descent could be a large cone  $\mathcal{M}_\epsilon^r$  in the objective space. But whenever every pure descent occurs, the threshold  $\epsilon$  is updated as  $l^t \in \mathcal{M}_\epsilon^r$ , i.e.  $\mu_r(l^t) \leq \epsilon$ . On the other hand, in balance mode  $\epsilon$  is kept unchanged. Notice also that the rate of decrease for high valued  $\epsilon$  is less as compared to the low valued ones. This facilitates more descent steps in the beginning by letting the iterate  $l^t$  move closer towards a local Pareto Front.

We split the LP in (24) into two different LP problems, such that only one of them is solved in every step depending on the position of the iterate  $l^t$ . If  $l^t \notin \mathcal{M}_\epsilon^r$ , then we solve

$$\begin{aligned} \beta^* &= \arg \max_{\beta \in \mathcal{S}^m} \beta^T C a & (73) \\ \text{s.t. } & \beta^T c_j \geq a^T c_j \mathbb{1}_J, \quad \forall j \in \bar{J} - J^* \\ & \beta^T c_j \geq 0, \quad \forall j \in J^* \end{aligned}$$

for balancing the multi-objective values. Whereas if  $l^t \in \mathcal{M}_\epsilon^r$ , then we solve

$$\begin{aligned} \beta^* &= \arg \max_{\beta \in \mathcal{S}^m} \beta^T C r \odot l^t & (74) \\ \text{s.t. } & \beta^T c_j \geq 0, \quad \forall j \in [m] & (75) \end{aligned}$$

in order to perform a pure descent step. Notice that the objective in (74) becomes the original objective  $\beta^T C \mathbf{1}$  when  $\epsilon = 0$ . The intuition behind generalizing the objective is to find a descent direction that will take the preference  $r$  into account while decreasing the objective values. Because, when  $\epsilon \neq 0$ , there is still scope for balancing the objectives, and maximizing  $\beta^T C r \odot l^t$  is an attempt to implicitly achieve that.

Although the annealed EPO Search is conceptually appealing – gradual reduction the angular reason of cone  $\mathcal{M}_\epsilon^r$  – we did not find any significant difference in search trajectories as compared to the plain EPO search of (24).

## C. Experimental Results

### C.1. Additional Properties of EPO Search

#### C.1.1. FINDING THE BEST POSSIBLE SOLUTION

We test our algorithm when the preferred solution does not exist, i.e.  $\mathcal{P}_r$  is empty. We shift the loss functions in (36) as  $l_1 \leftarrow l_1 + s$  and  $l_2 \leftarrow l_2 + s$  with a scalar value  $s > 0$ , so that 0 will not be the optimal value for any objective. As a result, not all rays in the positive quadrant will intersect the Pareto Front. We choose  $s = 1.4$  and use the same experimental setup (preference vectors  $r$ , step size and number of iterations) as in fig. 6. In 2 out of 4 cases, the exact Pareto optimal w.r.t. the preference vector does not exist. The result is shown in fig. 10. As discussed after

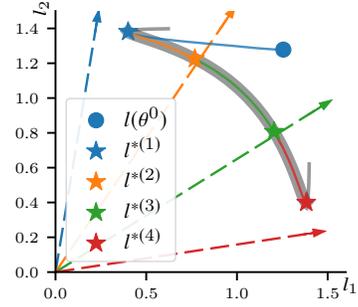


Figure 10. EPO search finds the solutions with minimum non-uniformity (in the sense of definition (9)), when the preference specific Pareto optimal does not exist.

claim 2, the EPO search finds a solution that is closest to the  $r^{-1}$  ray, when it doesn't intersect the Pareto front.

#### C.1.2. TRACING THE PARETO FRONT

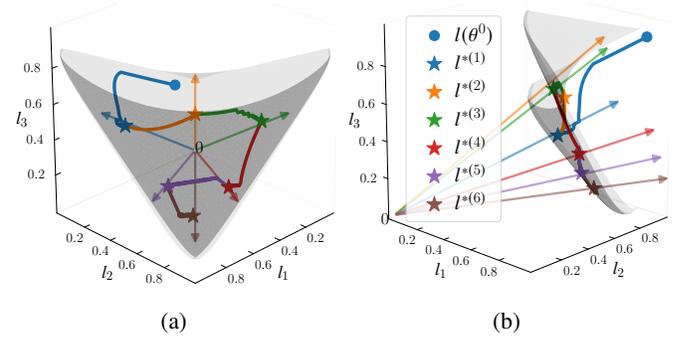


Figure 11. Tracing the Pareto Front of the 3 objectives in (76) using EPO search. Both figures plot the same data from different view points: azimuth angles for 11a and 11b are  $45^\circ$  and  $-45^\circ$  respectively, while both have the same elevation of  $25^\circ$ .

The previous experiment also shows how EPO search can be used to traverse the Pareto Front. We use the  $\theta^*$  of one preference to initialize EPO search for another preference. This technique can be used to trace curves, hence find many Pareto solutions, in high dimensional Pareto Fronts as well. To visually verify this, we construct the following three loss functions:

$$l_1(\theta) = 1 - \exp\left(-\left\|\theta - \frac{\mathbf{1}}{\sqrt{n}}\right\|_2^2\right), \quad (76a)$$

$$l_2(\theta) = 1 - \exp\left(-\left\|\theta + \frac{\mathbf{1}}{\sqrt{n}}\right\|_2^2\right), \quad (76b)$$

$$l_3(\theta) = 1 - \exp\left(-\left\|\theta + \frac{\tilde{\mathbf{1}}}{\sqrt{n}}\right\|_2^2\right), \quad (76c)$$

where  $\tilde{\mathbf{1}} \in \mathbb{R}^n$  with  $-1$  in odd indices and  $1$  in even indices. We then choose few evenly spread preference vectors in

$\mathbb{R}^3$  and solve for them sequentially. The curves generated while tracing the 2 dimensional Pareto front is shown fig. 11. Note that the  $r^{-1}$  ray corresponding to second preference (orange) doesn't intersect the Pareto Front (dark surface), so EPO search finds the closest solution on the border.

## C.2. Synthetic Experiments: Many Objectives

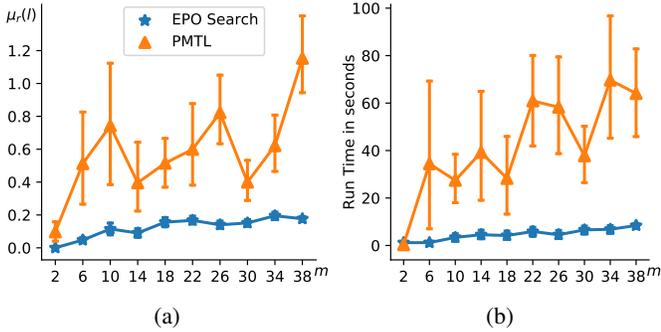


Figure 12. Comparison for how the algorithms scale with increasing number of objective functions. Fig. 12a is for the quality of preference specific optimal solution, and fig. 12b is for run time for 200 iterations.

We test how our algorithm scale with increasing number of objectives and compare that with Pareto MTL. We create  $m$  loss functions as

$$l_j(\theta) = 1 - \exp\left(-\|\theta - \hat{\theta}^j\|_2^2\right), \quad j \in [m], \quad (77)$$

where the entries of  $\hat{\theta}^j \in \mathbb{R}^n$  are sampled uniformly in  $[-1/n, 1/n]$ . For every  $m$ , we run both the algorithms for 20 different  $n$ , dimension of solution space, randomly sampled within 20 and 100. We randomly select a preference vector in  $\mathbb{R}_+^m$  for every  $(m, n)$  pair. In addition to a preference vector, the PMTL algorithm requires  $K$  reference vectors, which, according to the authors, should be increased exponentially with the increasing  $m$ . However, for a fair comparison, we provide  $K = 2m + 2$  (number of constraints in EPO search) reference vectors, which are again randomly  $\mathbb{R}_+^m$ .

We use the non-uniformity  $\mu_r$  in (9) as a measure of the quality of preference-specific Pareto optimal solutions found by the algorithms; result shown in fig. 12a. Clearly, EPO search scales better with increasing number of objectives as compared to the Pareto MTL method. For every  $(m, n)$  pair, both the algorithms were run for 200 iterations with equal step size using the same computing infrastructure (see sec. C.4). We used GNU's Linear Programming kit (glp, 2012) for solving the LP of EPO search. The comparison of overall run time (in seconds) is shown in 12b.

## C.3. Multi-Label Classification

We test our algorithm for multi-class classification of music into 6 emotions (Trohidis et al., 2011): amazed-surprised (E1), happy-pleased (E2), relaxing-calm (E3), quiet-still (E4), sad-lonely (E5), and angry-fearful (E6). A music is

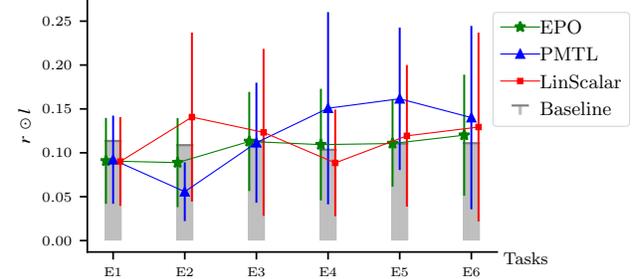


Figure 13. Relative loss profile for emotion recognition of music features.

represented using 72 features. There are 391 examples for training and 202 for testing. We used a feed forward neural network of 4 layers (of sizes  $72 \rightarrow 36 \rightarrow 18 \rightarrow 9 \rightarrow 6$ ). Sigmoid Binary Cross Entropy (SBCE) is used to compute the class-specific loss for every output unit of this network. Similar to the experiment in fig 8, we use the relative loss profiles (RLP), i.e.  $r \odot l$ , for comparison in fig. 13. The fact that linear scalarization has an almost uniform RLP suggests that the Pareto Front might be bordering a convex neighborhood in the objective space. Our algorithm has better or equal performance (in terms of relative loss), except for E2, as compared to Pareto MTL.

## C.4. Hyperparameters used and Computing Infrastructure Details

All the experiments were carried out in the same machine with an Intel Xeon Gold 6130 CPU (2.10GHz). Additionally, for the MTL experiments in figure 7, 8 and 13, we have used a NVIDIA Tesla V100 GPU.

Figures	$m$	$n$	$l$	$N$	$\eta$
1, 4, 10, 6 top row	2	20	eq (36)	80	0.1
6a bottom	2	20	eq (36)	80	0.4
6b bottom	2	20	eq (36)	200	0.4
7	2	31912	CE	100	$10^{-3}$
8	8	6896	MSE	100	$10^{-4}$
11	3	20	eq (76)	200	0.05
12	2 - 38	20 - 100	eq (77)	200	$1/n$
13	6	6595	SBCE	200	$10^{-3}$

Table 1. Hyper-parameters used in different experiments, referred using the corresponding figure number.

Details of the loss functions and hyper parameters used in all the experiments are provided in table 1 :  $m$  is number

of objectives/tasks;  $n$  is number of network parameters or dimension of solution space;  $l$  is the loss function for each task/objective;  $N$  is the number of epochs for MTL or iterations for MOO (synthetic experiments); and  $\eta$  is the learning rate for MTL or step size for MOO. In the loss functions, Cross Entropy is abbreviated as CE, Mean Square Error as MSE, and Sigmoid Binary Cross Entropy as SBCE. Both experiments in figure 8 and 13 uses tanh activation function in the MTL network, whereas the experiment in 7 uses ReLU activation function.

### References

- GLPK (GNU linear programming kit). In *Linear Programming and Algorithms for Communication Networks*, pp. 25–29. CRC Press, aug 2012.
- Lin, X., Zhen, H.-L., Li, Z., Zhang, Q.-F., and Kwong, S. Pareto multi-task learning. In *Advances in Neural Information Processing Systems (NeurIPS)*, pp. 12037–12047. 2019.
- Trohidis, K., Tsoumakas, G., Kalliris, G., and Vlahavas, I. Multi-label classification of music by emotion. *EURASIP Journal on Audio, Speech, and Music Processing*, 2011 (1):4, 2011. ISSN 1687-4722.