

# Injecting Imbalance Sensitivity for Multi-Task Learning

Zhipeng Zhou<sup>1</sup>, Liu Liu<sup>2,†</sup>, Peilin Zhao<sup>2</sup>, Wei Gong<sup>1,†</sup>

<sup>1</sup>University of Science and Technology of China

<sup>2</sup>Tencent AI Lab

zzp1994@mail.ustc.edu.cn, {leonliuliu, masonzhao}@tencent.com  
weigong@ustc.edu.cn

## Abstract

Multi-task learning (MTL) has emerged as a promising approach for deploying deep learning models in real-life applications. Recent studies have proposed optimization-based learning paradigms to establish task-shared representations in MTL. However, our paper empirically argues that these studies, specifically gradient-based ones, primarily emphasize the conflict issue while neglecting the potentially more significant impact of imbalance/dominance in MTL. In line with this perspective, we enhance the existing baseline method by injecting imbalance-sensitivity through the imposition of constraints on the projected norms. To demonstrate the effectiveness of our proposed IMGrad descent method, we evaluate it on multiple mainstream MTL benchmarks, encompassing supervised learning tasks as well as reinforcement learning. The experimental results consistently demonstrate competitive performance.

## 1 Introduction

Real-life scenarios often involve the need to handle multiple distinct tasks concurrently, typically achieved by designing task-specific models to ensure satisfactory performance. However, this approach becomes impractical as the number of tasks grows, as it would require significant computational resources and memory. To address this challenge and establish an efficient multi-task learning (MTL) framework, recent research has focused on developing a single model capable of performing well on all target tasks.

Currently, research on MTL can be broadly categorized into two frameworks: architecture-based [Liu *et al.*, 2019; Ye and Xu, 2022; Gao *et al.*, 2019; Chen *et al.*, 2023] and optimization-based approaches [Sener and Koltun, 2018; Yu *et al.*, 2020a; Liu *et al.*, 2021a; Zhou *et al.*, ; Liu *et al.*, 2023]. The former emphasizes the design of efficient parameter sharing architectures for multiple tasks, whereas the latter typically employs a fixed architecture and focuses on developing optimization strategies to extract task-shared representa-

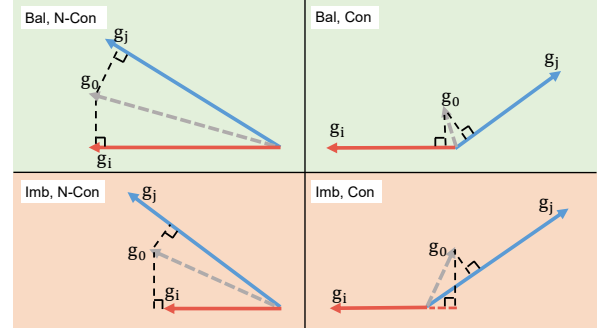


Figure 1: Illustration of imbalance and conflicting issue in multi-task learning. ‘Bal’ and ‘Imb’ represent balanced and imbalanced, while ‘N-Con’ and ‘Con’ represent non-conflicting and conflicting.

tions. In this paper, we exclusively introduce and compare our method with optimization-based approaches, as our proposed method falls within this framework.

In the realm of optimization-based methods, particularly those involving gradient manipulation, a shared paradigm is commonly followed, where task gradients are combined to achieve Pareto optimality for individual tasks. Despite the high performance demonstrated by these methods, the literature has predominantly overlooked the significance of the inherent imbalance nature among individuals (see **Definition 2**). This oversight can be attributed to the greater emphasis placed on addressing the conflict issue. However, it is important to note that the conflict issue alone may not be the fundamental obstacle hindering optimization in MTL. As illustrated in Figure 1, a naïve linear scalarization (LS) strategy ( $g_{mean}$ ) effectively improves all individuals when they are balanced, regardless of conflicts. But it proves ineffective when both imbalance and conflict coexist, underscoring the importance of addressing conflicts that arise solely from imbalances. Furthermore, imbalanced task gradients can introduce optimization preferences and lead to imbalanced progress even in the absence of conflicts [Liu *et al.*, 2023]. Although previous solutions, such as IMTL [Liu *et al.*, 2021b] and Nash-MTL [Navon *et al.*, 2022] illustrated in Table 1, have somewhat mitigated the imbalance/dominance issue, they neither explicitly provide evidence to demonstrate the importance of the imbalance issue nor consider both conflict and imbalance issues simultaneously.

<sup>†</sup> Corresponding authors. Work done when Z. Zhou works as an intern in Tencent AI Lab.

|                     | GD | GradDrop | MGDA | PCGrad | IMTL | CAGrad | Nash-MTL | MoCo | IMGrad |
|---------------------|----|----------|------|--------|------|--------|----------|------|--------|
| Conflict-averse     | ✗  | ✗        | ✓    | ✓      | ✗    | ✓      | ✗        | ✓    | ✓      |
| Imbalance-sensitive | ✗  | ✗        | ✗    | ✗      | ✓    | ✗      | ✓        | ✗    | ✓      |

Table 1: Conflict-averse and imbalance-sensitive comparison for mainstream optimization-based MTL. Note that those which are imbalance-sensitive mean that their solution can tackle the imbalance issue.

In this paper, we begin by empirically highlighting the significance of the imbalance issue in MTL and elucidate the advantages of incorporating imbalance sensitivity into baseline methods as our primary motivation. Subsequently, we enhance the well-established baseline method by injecting imbalance sensitivity through the imposition of constraints on the projected norms. Convergence and speedup analysis are provided in the **Appendix**<sup>1</sup>. In a nutshell, we summarize our contributions as three-fold:

- We place significant emphasis on and empirically identify that the primary challenge in optimization-based MTL lies more in the aspect of imbalance rather than conflict. To the best of our knowledge, we are the first to explicitly assert this claim.
- To introduce the imbalance sensitivity into the existing paradigm, we integrate the projected norm constraint into the objectives. This incorporation allows for a dynamic equilibrium between Pareto property (see **Definition 3**) and convergence (two decoupled objectives), thereby enhancing the combined gradients and optimization trajectories.
- The extensive experimental results present compelling evidence that IMGrad consistently enhances its baselines and surpasses the current advanced gradient manipulation methods in a diverse range of evaluations, e.g., supervised learning tasks, and reinforcement learning benchmarks.

## 2 Related Work

Currently, MTL approaches can be broadly categorized into two groups: architecture-based and optimization-based methods. Architecture-based approaches encompass various paradigms, including hard parameter sharing [Heuer *et al.*, 2021; Kokkinos, 2017], soft parameter sharing [Yang and Hospedales, 2016; Gao *et al.*, 2019], modulation and adapters [He *et al.*, 2021; Liu *et al.*, 2022], and mixture of experts (MoE) [Chen *et al.*, 2023; Fan *et al.*, 2022], etc. On the other hand, optimization-based MTL methods primarily focus on learning paradigms rather than structural designs or parameter sharing strategies. These methods aim to optimize all individual tasks to extract task-shared representations.

One classical optimization-based MTL approach is MGDA [Sener and Koltun, 2018], which seeks a combined gradient with minimal norm using the Frank-Wolfe algorithm [Jaggi, 2013]. PCGrad [Yu *et al.*, 2020a] addresses the conflict issue by projecting individual gradients onto orthogonal directions with respect to others. CAGrad [Liu *et al.*, 2021a] considers preserving both the Pareto property

and global optimization, ultimately striving for a balance between the two objectives using a hyper-parameter. Nash-MTL [Navon *et al.*, 2022] negotiates to reach an agreement on a joint direction of parameter update, enabling all individual tasks to achieve more balanced progress. MoCo [Fernando *et al.*, 2023] tackles the problem of biased gradient directions in previous solutions by developing tracking parameters for correction. Our method falls within the realm of optimization-based MTL, with a specific focus on addressing the issue of imbalance-sensitivity, which is largely lacking in the aforementioned solutions.

**Discussion with Counterparts:** To the best of our knowledge, IMTL [Liu *et al.*, 2021b], Nash-MTL [Navon *et al.*, 2022], and FAMO [Liu *et al.*, 2023] are three recent works that explicitly consider the imbalance issue. However, all three works fail to provide evidence demonstrating the importance of the imbalance issue. Moreover, none of these approaches possess conflict-averse properties. Thus, there is still room for improvement. Although Nash-MTL appears to be designed to avoid conflicts, its practical implementation does not achieve this goal. Please refer to the **Appendix** for more discussion.

## 3 Preliminary

### 3.1 Setup of Optimization-based MTL

As mentioned, optimization-based MTL approaches operate under the assumption that the model consists of a task-shared backbone network alongside task-specific branches. Consequently, the primary objective of these approaches is to devise gradient combination strategies that optimize the backbone network to yield benefits across all tasks. Let us consider a scenario where there are  $K \geq 2$  tasks available, each associated with a differentiable loss function  $\mathcal{L}_i(\Theta)$ , where  $\Theta$  represents the task-shared parameters. The goal of optimization-based MTL is to search for the optimal  $\Theta^* \in \mathbb{R}^m$  that minimizes the losses for all tasks. However, it is widely recognized that a simplistic linear scalar strategy,  $\mathcal{L}_0(\Theta) = \frac{1}{K} \sum_{i=1}^K \mathcal{L}_i(\Theta)$ , fails to achieve satisfactory performance due to the conflict and imbalance issue.

### 3.2 Pareto Concept

Formally, let us assume the weighted loss as  $\mathcal{L}_\omega = \sum_{i=1}^K \omega_i \mathcal{L}_i(\Theta)$ , where  $\omega \in \mathcal{W}$  and  $\mathcal{W}$  represents the probability simplex on  $[K]$ . A point  $\Theta'$  is said to Pareto dominate  $\Theta$  if and only if  $\forall i, \mathcal{L}_i(\Theta') \leq \mathcal{L}_i(\Theta)$ . Consequently, the Pareto optimal situation arises when no  $\Theta'$  can be found that satisfies  $\forall i, \mathcal{L}_i(\Theta') \leq \mathcal{L}_i(\Theta)$  for the given point  $\Theta$ . All points that meet these conditions are referred to as Pareto sets, and their solutions are known as Pareto fronts. Another concept, known as Pareto stationary, requires  $\min_{\omega \in \mathcal{W}} \|g_\omega\| = 0$ , where  $g_\omega$

<sup>1</sup>Refer to <https://arxiv.org/abs/2503.08006> for the Appendix.

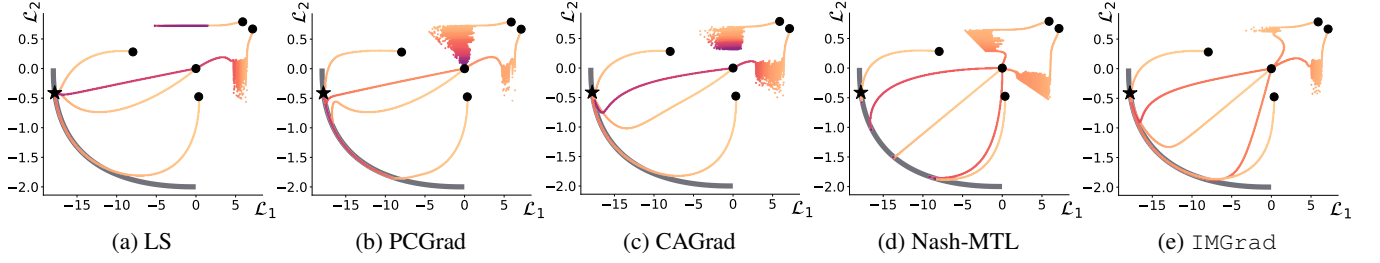


Figure 2: Comparison of MTL approaches on the imbalanced synthetic two-task benchmark. • and ★ represent the starting point and global optimum, respectively, and grey line — represents the Pareto front. Two objectives are extremely imbalanced weighted, i.e.,  $(0.9 * \mathcal{L}_1, 0.1 * \mathcal{L}_2)$ . Please refer to the **Appendix** for more optimization trajectories under various pre-defined task weights.

represents the weighted gradient  $\omega^\top G$ , and  $G$  is the gradients matrix whose each row is an individual gradient. We also provide some definitions here for ease of description.

**Definition 1 (Gradient Similarity).** Denote  $\phi_{ij}$  as the angle between two task gradients  $g_i$  and  $g_j$ , then we define the gradient similarity as  $\cos \phi_{ij}$  and the gradients as conflicting when  $\cos \phi_{ij} < 0$ .

**Definition 2 (Imbalance of Individuals).** Assume the gradient owns the maximal norm in  $G$  is  $g_{max}$ , and the corresponding minimal one is  $g_{min}$ . We define the imbalance ratio of  $G$  as  $r = \frac{\|g_{max}\|}{\|g_{min}\|}$ . If  $r > 1$ , we call it's imbalanced.

**Definition 3 (Pareto Property).** For each training step, the combined optimization direction strives to promote all individuals simultaneously (or at the very least, not cause detriment), i.e. for  $\forall i$ , the gradient similarity between  $g_i$  and the combined gradient  $g_\omega$  satisfies  $\cos \phi_{\omega i} \geq 0$ . When this condition is not met, it is referred to as **Pareto failure**.

## 4 Motivation and Observation

A substantial body of previous studies [Sener and Koltun, 2018; Liu *et al.*, 2021a; Yu *et al.*, 2020a; Navon *et al.*, 2022] have primarily focused on addressing the conflict issue rather than the imbalance issue. In this section, we aim to provide empirical insights into the significance of imbalance and elucidate how imbalance-sensitivity can bring benefits to current popular optimization-based MTL paradigms. Based on these insights, we naturally deduce our design in the next section.

### 4.1 Why Does Imbalance Matter More?

To begin, we conducted experiments on the CityScapes dataset [Cordts *et al.*, 2016] to statistically analyze the imbalance ratios of representative optimization-based MTL methods (e.g., PCGrad [Yu *et al.*, 2020a], CAGrad [Liu *et al.*, 2021a], Nash-MTL [Navon *et al.*, 2022]). The results of these experiments are presented in the **Appendix**. From the depicted results, it is evident that all the methods exhibit significant imbalance during training, which poses a substantial challenge when attempting to optimize all individuals simultaneously, thereby underscoring the importance of addressing the imbalance issue.

Secondly, to demonstrate the higher priority of imbalance issue, we show the toy example results that present imbalance and conflict among gradients in the following cases:

- Conflict (✓); Imbalance (✗): In Figure 3, we manually create scenarios where conflict exists but imbalance is absent. By closely examining the center trajectories in Figure 3 (d)(e), we observe that all methods can easily reach the optimal point when imbalance is absent, regardless of the presence of conflicts. This observation suggests that the sole existence of conflicts has limited impact on optimization, emphasizing the importance of addressing the imbalance issue.
- Conflict (✗); Imbalance (✓): Simulating an optimization trajectory without conflicts among individuals can indeed be challenging. Therefore, we adopt the setting from Nash-MTL [Navon *et al.*, 2022] to handcraft an imbalance-dominated optimization scenario. The resulting trajectories are depicted in Figure 2. It is evident that all the compared approaches fail to converge at the desired global optimum from all initial starts under the extreme imbalance circumstances, though most of them reach the Pareto front. Additionally, the trajectories at the sides in Figure 3 (d)(e) also highlight the issue of progress hindered by imbalance. Specifically, CAGrad fails to reach the global optimum compared to IMGrad despite undergoing the same number of optimization steps.

### 4.2 The Impacts of the Imbalance Issue

In Table 1, we list and compare mainstream optimization-based MTL approaches. The table focuses on two key properties: conflict-averse and imbalance-sensitive properties. It is observed that most MTL approaches possess the conflict-averse property due to their design nature. However, only a few approaches are imbalance-sensitive<sup>2</sup>, and currently, there are no methods that possess both properties simultaneously. Furthermore, we analyze two imbalance-deduced issues that occur and impede past solutions during optimization: Pareto failure and imbalanced individual progress.

**Pareto Failure:** As shown in Figure 4 (a)(b), CAGrad exhibits a certain probability of failing to preserve the conflict issue due to its inherent compromise between conflict-averse and convergence. This compromise is inevitably influenced by the issue of imbalance. As illustrated in Figure 6, CAGrad tends to prioritize the combined gradient that deviates from the

<sup>2</sup>We provide imbalance-sensitive analysis for IMTL, Nash-MTL, and FAMO in the **Appendix**.

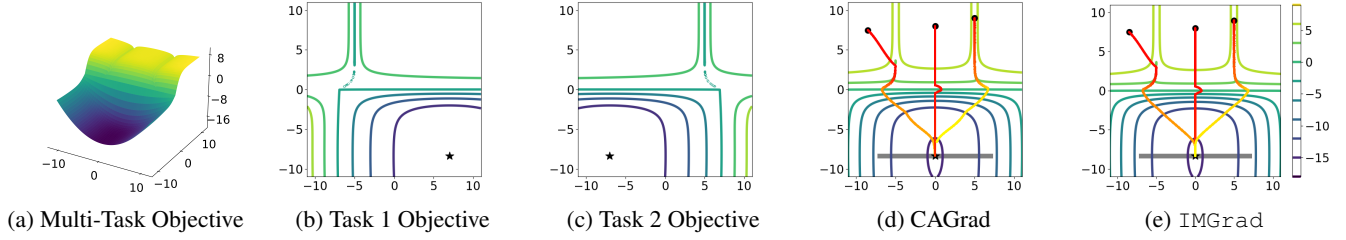


Figure 3: Comparison of MTL approaches on the toy examples. We use the tool provided CAGrad to generate the synthetic toy examples with two objective shown in (b) and (c). In this case, both objective are equally weighted.

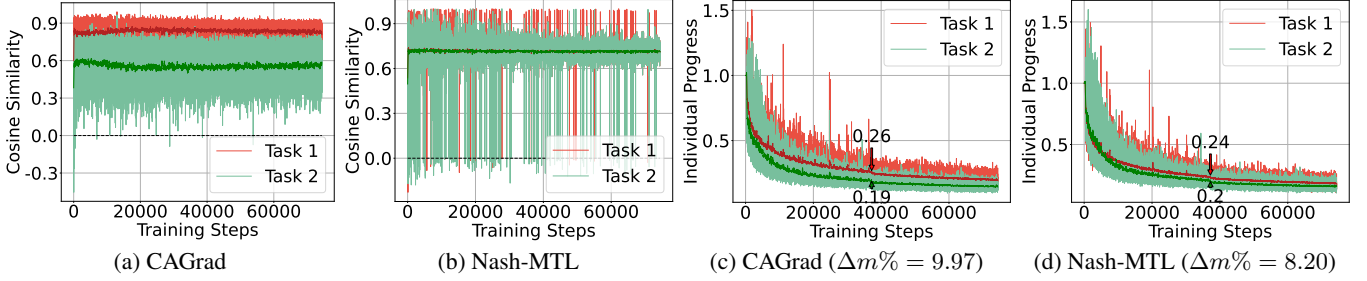


Figure 4: Individual gradient similarity and progress analysis of MTL algorithms on CityScapes. (a)-(c) show the gradient similarities between individuals and the combined gradient; (d)-(e) present the progress of individuals during optimization.

individual with the least norm when encountering imbalanced scenarios, leading to potential conflicts. Surprisingly, although Nash-MTL imposes a strong constraint for the Pareto property, i.e.,  $\forall i, -\varphi_i(\omega) \leq 0, \varphi_i(\omega) = \log(\omega_i) + \log(\mathbf{g}_i^\top \mathbf{G}\omega)$ ,  $\mathbf{G} = [\mathbf{g}_1, \mathbf{g}_2, \dots, \mathbf{g}_K]$ , it often fails to achieve such a guarantee. This failure can be attributed to the presence of negative terms in  $\mathbf{g}_i^\top \mathbf{G}$ , indicating conflicts between  $\mathbf{g}_i$  and  $\mathbf{g}_j$ . Consequently, this leads to infeasible errors in the *cvxpy* [Diamond and Boyd, 2016] implementation, and the Nash-MTL algorithm chooses to skip the current step when such errors occur. As a result, Nash-MTL frequently encounters Pareto failures due to the co-existence of imbalance and conflict, as depicted in Figure 4 (b). IMGrad demonstrates a tendency to acquire a combined gradient that effectively preserves the Pareto property as the imbalance ratio increases.

**Imbalanced Individual Progress:** We employ an individual progress metric proposed by [Chen *et al.*, 2018], which is defined as follows:

$$r_i(t) = \mathcal{L}_i(t) / \mathcal{L}_i(0) \quad (1)$$

where  $\mathcal{L}_i(t)$  represents the individual loss value at  $t$  time. As depicted in Figure 4 (c)(d), Nash-MTL demonstrates a narrower gap in terms of individual progress compared to CAGrad. This can be attributed to the more balanced combination employed by Nash-MTL, as indicated by the cosine similarity in (a)(b). Consequently, Nash-MTL exhibits superior overall performance, characterized by a smaller  $\Delta m\%$ . Specifically,  $\Delta m\%$  is widely adopted to evaluate the overall degradation compared to independently trained models, which are considered as the reference oracles. Its formal definition can be found in the **Performance Evaluation** section.

Unfortunately, none of the above methods get rid of both Pareto failure and imbalanced individual progress, primarily due to their limited focus on the imbalance issue.

### 4.3 Benefits of Integrating Imbalance-Sensitivity

The toy results depicted in Figure 2 and Figure 3 demonstrate that among the methods evaluated, only our proposed IMGrad, which incorporates imbalance-sensitivity, consistently arrives at the optimal point from all initial starts.

To further elucidate the advantages of imbalance-sensitivity in optimization-based MTL, we have implemented a naïve method called *Adaptive Threshold*. This baseline selectively applies optimization-based MTL approaches only when the imbalance ratio surpasses a specific threshold. The results of this implementation on CityScapes are presented in Figure 5 (a). It is evident that all baselines exhibit varying performance as the imbalance ratio fluctuates, emphasizing the significance of imbalance-sensitivity. Notably, all baselines outperform their respective vanilla versions under specific threshold conditions, providing additional evidence of the effectiveness of injecting imbalance-sensitivity.

Additionally, we have conducted a series of control group experiments to further support our findings. Similarly, we only apply optimization-based MTL when the gradient similarity falls below a certain threshold. As depicted in Figure 5 (b), all baselines demonstrate relatively stable performance compared to those in (a) and fail to outperform the vanilla version, except for MGDA (which itself performs worse than LS). This outcome further reinforces the claim that imbalance matters more.



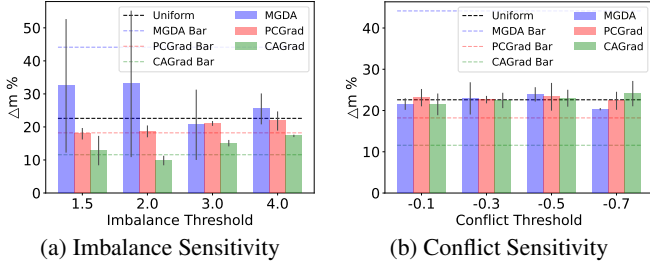


Figure 5: Imbalance and conflict sensitivity examination.

## 5 Principal Design

In this section, taking CAGrad as the baseline, we present the principal design of IMGrad, encompassing its formulation in the objective function and the practical implementation. And we provide convergence and speedup analysis in the **Appendix**.

### 5.1 Injecting Imbalance-Sensitivity

As a widely adopted baseline, CAGrad strikes a balance between Pareto property and globe convergence, and its dual objective is formulated as follows:

$$\max_{d \in \mathbb{R}^m} \min_{\omega \in \mathcal{W}} g_{\omega}^{\top} d \quad \text{s.t.} \quad \|d - g_0\| \leq c \|g_0\| \quad (2)$$

where  $d$  represents the combined gradient, while  $g_0$  denotes the averaged gradient, and  $c$  is the hyper-parameter.

To alleviate the imbalance-deduced Pareto failures or individual progress issue as illustrated in Figure 4, a logical approach is to maximize the projected norm of the combined gradient across all individuals. To achieve this, we incorporate a stronger constraint ( $g_i^{\top} d - \|g_i\|^2$ ) into Eqn. 2, which encourages projected norms that surpass individual norms. This formulation is reflected in our objective presented in Eqn. 3, and subsequently, we derive the corresponding Lagrangian equations in Eqn. 4.

$$\max_{d \in \mathbb{R}^m} \min_{\omega \in \mathcal{W}} g_{\omega}^{\top} d - \mu (g_{\omega}^{\top} d - \|g_{\omega}\|^2) \quad \text{s.t.} \quad \|d - g_0\| \leq c \|g_0\| \quad (3)$$

$$\begin{aligned} \max_{d \in \mathbb{R}^m} \min_{\lambda \geq 0, \omega \in \mathcal{W}} & g_{\omega}^{\top} d - \lambda (\|d - g_0\|^2 - \phi) / 2 \\ & - \mu (g_{\omega}^{\top} d - \|g_{\omega}\|^2), \quad \lambda > 0, \mu > 0 \end{aligned} \quad (4)$$

The strong duality property holds for the aforementioned objective, as supported by convex programming principles and the fulfillment of Slater's condition. Consequently, we interchange the positions of the minimum and maximum operators:

$$\begin{aligned} \min_{\lambda \geq 0, \omega \in \mathcal{W}} \max_{d \in \mathbb{R}^m} & (1 - \mu) g_{\omega}^{\top} d \\ & - \frac{\lambda}{2} (\|d - g_0\|^2 - \phi) + \mu \|g_{\omega}\|^2 \end{aligned} \quad (5)$$

With  $\lambda, \omega$  fixing, the optimal  $d$  is achieved when  $d = g_0 + \frac{(1-\mu)g_{\omega}}{\lambda}$ . Substitute the optimal  $d$  into Eqn. 5, yielding the

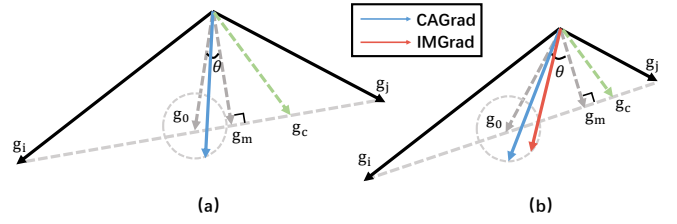


Figure 6: Multi-objective optimization Comparison between CAGrad and IMGrad. Here we suppose the angles between  $g_i$  and  $g_j$  in (a) and (b) are same.  $g_m$  can be obtained via MGDA.

following problem:

$$\begin{aligned} \min_{\lambda \geq 0, \omega \in \mathcal{W}} & (1 - \mu) g_{\omega}^{\top} g_0 + \mu \|g_{\omega}\|^2 \\ & + \frac{(1 - \mu)^2}{2\lambda} \|g_{\omega}\|^2 + \frac{\lambda}{2} \phi \end{aligned} \quad (6)$$

After optimizing out the  $\lambda$  we have

$$\min_{\omega \in \mathcal{W}} (1 - \mu) g_{\omega}^{\top} g_0 + \mu \|g_{\omega}\|^2 + (1 - \mu) \sqrt{\phi} \|g_{\omega}\| \quad (7)$$

where  $\lambda = (1 - \mu) \|g_{\omega}\| / \phi^{1/2}$ , and finally we have the optimization objective in Eqn. 8. By solving this objective, we can obtain  $g_{\omega}$  and have  $d = g_0 + \frac{\phi^{1/2}}{\|g_{\omega}\|} g_{\omega}$ .

$$\min_{\omega \in \mathcal{W}} (1 - \mu) \underbrace{(g_{\omega}^{\top} g_0 + \sqrt{\phi} \|g_{\omega}\|)}_{\text{CAGrad}} + \underbrace{\mu \|g_{\omega}\|^2}_{\sim \text{MGDA}} \quad (8)$$

Upon careful examination of Eqn. 8, it becomes evident that the final objective can be decomposed into two distinct components: CAGrad and MGDA. As depicted in Figure 6 (a), the gradient obtained by solving the practical objective in Eqn. 10, denoted as  $g_c$  (represented by the green dotted line), predominantly resides within the region bounded by  $g_m$  and  $g_j$ . However, in the case of an extreme imbalance scenario, as illustrated in Figure 6 (b), the corresponding  $g_c$  tends to lean towards the dominant gradient  $g_i$ , thereby increasing the risk of conflicting with  $g_j$  and resulting in Pareto failures. When confronted with such a situation characterized by varying imbalances, it is desirable for  $\mu$  to adaptively adjust  $g_c$  to consistently avoid Pareto failures while still promoting individual progress when the imbalance is less pronounced. Consequently, we establish a connection between  $\mu$  and the gradient imbalances, effectively controlling the constraint ( $g_i^{\top} d - \|g_i\|^2$ ) adaptively based on the imbalance circumstances.

Multiple alternatives exist for quantifying the imbalance ratio among individuals<sup>3</sup>. We here choose to compute  $\cos \theta$  to represent the imbalance ratio (see negative correlation between imbalance ratio and  $\cos \theta$  in the **Appendix**), where  $\theta$  denotes the angle between  $g_0$  and  $g_m$ . As a result, Eqn. 8 can be re-written as:

$$\min_{\omega \in \mathcal{W}} (1 - \cos \theta) (g_{\omega}^{\top} g_0 + \sqrt{\phi} \|g_{\omega}\|) + \cos \theta \|g_{\omega}\|^2 \quad (9)$$

<sup>3</sup>Please refer to the **Appendix** for additional alternatives.

**Simplification:** As a matter of fact, CAGrad itself contains decoupled components in its practical objective:

$$\min_{\omega \in \mathcal{W}} \underbrace{g_{\omega}^{\top} g_0}_{\text{Push Away from } g_0} + \underbrace{\sqrt{\phi} \|g_{\omega}\|}_{\sim \text{MGDA}} \quad (10)$$

where  $g_{\omega}^{\top} g_0$  tends to push away from  $g_0$  and  $\sqrt{\phi} \|g_{\omega}\|$  plays the role of MGDA does. Thus we can simplify the Eqn. 9 as:

$$\min_{\omega \in \mathcal{W}} (1 - \cos \theta) g_{\omega}^{\top} g_0 + \cos \theta \sqrt{\phi} \|g_{\omega}\|^2 \quad (11)$$

## 5.2 Augment Nash-MTL with Imbalance Sensitivity

As stated in the previous **Pareto Failure** analysis, while Nash-MTL effectively addresses the imbalance issue and appears to be naturally conflict-averse, its implementation often leads to frequent Pareto failures. To address this problem, let's first examine its decoupled objective:

$$\min_{\omega} \underbrace{\sum_i g_i^{\top} G \omega}_{\text{Push Away from } g_0} + \underbrace{\varphi(\omega)}_{\text{Strike balance among individuals}} \quad (12)$$

s.t.  $\forall i, -\varphi_i(\omega) \leq 0, \omega_i > 0$

where  $\varphi_i(\omega) = \log(\omega_i) + \log(g_i^{\top} G \omega)$ ,  $G = [g_1, g_2, \dots, g_K]$ .  $\sum_i g_i^{\top} G \omega$  tends to push away from  $g_0$  and  $\varphi(\omega)$  strives balance among individuals. Intuitively, we expect to preserve the Pareto property when encountering extremely imbalanced scenarios; therefore,  $\sum_i g_i^{\top} G \omega$  should be given more weight:

$$\min_{\omega} (1 - \cos \theta) \sum_i g_i^{\top} G \omega + \cos \theta \varphi(\omega) \quad (13)$$

With the proper assumption of H-Lipschitz on gradients, we can still avoid Pareto failure with the derived weights among individuals from the last step. In a word, we augment Nash-MTL by injecting imbalance sensitivity to reduce Pareto failures. Please refer to the **Appendix** for more details.

## 5.3 Implementation

We implement our approach with Python 3.8, PyTorch 1.4.0 and cvxpy 1.3.1, while all experiments are carried out on Tesla V100 GPUs<sup>4</sup>. We follow the setting and general implementation of [Liu *et al.*, 2021a], and the toy example generation is borrowed from [Navon *et al.*, 2022; Senushkin *et al.*, 2023]. See more implementation details in the **Appendix**.

## 6 Performance Evaluation

Following the evaluation protocol in [Navon *et al.*, 2022] and taking it as the baseline, we conduct experiments under the supervised learning and reinforcement learning scenarios. Specifically, two scene understanding and one image classification benchmarks are involved in supervised learning, and the classical MT10 benchmark is adopted for reinforcement learning. The examination of Pareto failures, individual task progress, a sensitivity analysis of  $\mu$ , the verification of negative correlation between imbalance ratio and  $\cos \theta$ , speed analysis,

and more visualizations are also provided in the **Appendix**, please refer them for more details.

**Evaluation metric.** In addition to reporting individual performance, we also incorporate a widely used metric,  $\Delta m\%$  [Maninis *et al.*, 2019], which evaluates the overall degradation compared to independently trained models that are considered as the reference oracles. The formal definition of  $\Delta m\%$  is given as:  $\Delta m\% = \frac{1}{K} \sum_{k=1}^K (-1)^{\delta_k} (M_{m,k} - M_{b,k}) / M_{b,k}$ .  $M_{m,k}$  and  $M_{b,k}$  represent the metric  $M_k$  for the compared method and the independent model, respectively. The value of  $\delta_k$  is assigned as 1 if a higher value is better for  $M_k$ , and 0 otherwise.

| Method      | Segmentation    |           | Depth          |           | $\Delta$ m% ↓ |
|-------------|-----------------|-----------|----------------|-----------|---------------|
|             | (Higher Better) |           | (Lower Better) |           |               |
|             | mIoU            | Pix. Acc. | Abs. Err.      | Rel. Err. |               |
| Independent | 74.01           | 93.16     | 0.0125         | 27.77     | -             |
| LS          | 75.18           | 93.49     | 0.0155         | 46.77     | 22.60         |
| RLW         | 74.57           | 93.41     | 0.0158         | 47.79     | 24.37         |
| DWA         | 75.24           | 93.52     | 0.0160         | 44.37     | 21.43         |
| MGDA        | 68.84           | 91.54     | 0.0309         | 33.50     | 44.14         |
| GradDrop    | 75.27           | 93.53     | 0.0157         | 47.54     | 23.67         |
| PCGrad      | 75.13           | 93.48     | 0.0154         | 42.07     | 18.21         |
| CAGrad      | 75.16           | 93.48     | 0.0141         | 37.60     | 11.58         |
| IMTL        | 75.33           | 93.49     | 0.0135         | 38.41     | 11.04         |
| Nash-MTL    | 75.41           | 93.66     | 0.0129         | 35.02     | 6.82          |
| MoCo        | 75.42           | 93.55     | 0.0149         | 34.19     | 9.90          |
| FAMO        | 74.54           | 93.29     | 0.0145         | 32.59     | 8.13          |
| IMGrad      | 75.13           | 93.45     | 0.0128         | 34.95     | 6.61          |

Table 2: **Scene understanding** (*CityScapes*, 2 tasks). We report MTAN model performance averaged over 3 random seeds.

## 6.1 Supervised Learning

Customary evaluation in supervised learning for MTL involves assessing the ability of MTL approaches to handle multiple scene understanding and classification tasks. For scene understanding tasks, we follow previous studies [Liu *et al.*, 2021a; Liu *et al.*, 2021b; Navon *et al.*, 2022] and employ a Multi-Task Attention Network (MTAN) [Liu *et al.*, 2019] as the fundamental architecture for all MTL approaches. Our experiments are conducted on two well-established datasets: NYUv2 [Silberman *et al.*, 2012] and CityScapes [Cordts *et al.*, 2016]. To ensure fair comparisons, we adopt the same training strategy as described in prior works [Liu *et al.*, 2021a; Navon *et al.*, 2022]. Specifically, models are trained for 200 epochs using the Adam optimizer, with an initial learning rate of  $1e-4$ , which decays to  $5e-5$  after 100 epochs. For the image classification task, we utilize a 9-layer convolutional neural network (CNN) as the backbone, with linear layers serving as task-specific heads, and conduct experiments on CelebA [Liu *et al.*, 2015]. The model is trained using the Adam optimizer for 15 epochs, with an initial learning rate of  $3.0e-4$  and a batch size of 256.

**NYUv2.** NYUv2 is a widely used indoor scene understanding dataset for MTL benchmarking, encompassing three tasks:

<sup>4</sup>Code is available at <https://github.com/zzpustc/IMGrad>.

| Method      | Segmentation    |           | Depth          |         | Surface Normal |        |                  |       |       | $\Delta$ m% $\downarrow$ |
|-------------|-----------------|-----------|----------------|---------|----------------|--------|------------------|-------|-------|--------------------------|
|             | (Higher Better) |           | (Lower Better) |         | Angle Distance |        | Within $t^\circ$ |       |       |                          |
|             |                 |           |                |         | (Lower Better) |        | (Higher Better)  |       |       |                          |
|             | mIoU            | Pix. Acc. | Abs Err        | Rel Err | Mean           | Median | 11.25            | 22.5  | 30    |                          |
| Independent | 38.30           | 63.76     | 0.68           | 0.28    | 25.01          | 19.21  | 30.14            | 57.20 | 69.15 | -                        |
| LS          | 39.29           | 65.33     | 0.55           | 0.23    | 28.15          | 23.96  | 22.09            | 47.50 | 61.08 | 5.46                     |
| RLW         | 37.17           | 63.77     | 0.58           | 0.24    | 28.27          | 24.18  | 22.26            | 47.05 | 60.62 | 7.67                     |
| DWA         | 39.11           | 65.31     | 0.55           | 0.23    | 27.61          | 23.18  | 24.17            | 50.18 | 62.39 | 3.49                     |
| MGDA        | 30.47           | 59.90     | 0.61           | 0.26    | 24.88          | 19.45  | 29.18            | 56.88 | 69.36 | 1.47                     |
| GradDrop    | 39.39           | 65.12     | 0.55           | 0.23    | 27.48          | 22.96  | 23.38            | 49.44 | 62.87 | 3.61                     |
| PCGrad      | 38.06           | 64.64     | 0.56           | 0.23    | 27.41          | 22.80  | 23.86            | 49.83 | 63.14 | 3.83                     |
| CAGrad      | 39.79           | 65.49     | 0.55           | 0.23    | 26.31          | 21.58  | 25.61            | 52.36 | 65.58 | 0.29                     |
| IMTL        | 39.35           | 65.60     | 0.54           | 0.23    | 26.02          | 21.19  | 26.20            | 53.13 | 66.24 | -0.59                    |
| Nash-MTL    | 40.13           | 65.93     | 0.53           | 0.22    | 25.26          | 20.08  | 28.40            | 55.47 | 68.15 | -4.04                    |
| MoCo        | 40.30           | 66.07     | 0.56           | 0.21    | 26.67          | 21.83  | 25.61            | 51.78 | 64.85 | 0.16                     |
| FAMO        | 38.88           | 64.90     | 0.55           | 0.22    | 25.06          | 19.57  | 29.21            | 56.61 | 68.98 | -4.10                    |
| IMGrad      | 40.20           | 66.19     | 0.52           | 0.22    | 25.15          | 19.94  | 28.69            | 55.80 | 68.44 | -4.57                    |

Table 3: **Scene understanding** (NYUv2, 3 tasks). We report MTAN model performance averaged over 3 random seeds.

semantic segmentation, depth estimation, and surface normal prediction. The results, presented in Table ??, show that IMGrad surpasses the previous SOTA in terms of  $\Delta m\%$ , highlighting the effectiveness of incorporating imbalance sensitivity. IMGrad also achieves best performance on segmentation and depth tasks without much promise on other tasks.

**CityScapes.** The CityScapes dataset is used for MTL evaluation, focusing on semantic segmentation and depth estimation tasks. Following the previous experimental setup, we utilize a coarser version that categorizes segmentation into 7 classes. The results, presented in Table ??, indicate that IMGrad exhibits a similar trend to its performance on NYUv2 and achieves SOTA results in terms of  $\Delta m\%$ .

| MT10     |                              | CelebA   |                           |
|----------|------------------------------|----------|---------------------------|
| Method   | Success $\pm$ SEM $\uparrow$ | Method   | $\Delta m\%$ $\downarrow$ |
| LS       | 0.49 $\pm$ 0.070             | LS       | 4.15                      |
| STL SAC  | 0.90 $\pm$ 0.032             | SI       | 7.20                      |
| MTL SAC  | 0.49 $\pm$ 0.073             | RLW      | 1.46                      |
| MH SAC   | 0.54 $\pm$ 0.047             | DWA      | 3.20                      |
| SM       | 0.73 $\pm$ 0.043             | UW       | 3.23                      |
| CARE     | 0.84 $\pm$ 0.051             | MGDA     | 14.85                     |
| PCGrad   | 0.72 $\pm$ 0.022             | PCGrad   | 3.17                      |
| CAGrad   | 0.83 $\pm$ 0.045             | CAGrad   | 2.48                      |
| Nash-MTL | 0.91 $\pm$ 0.031             | Nash-MTL | 2.84                      |
| FAMO     | 0.83 $\pm$ 0.050             | FAMO     | 1.21                      |
| IMGrad   | 0.93 $\pm$ 0.068 (+0.10)     | IMGrad   | 1.27                      |

Table 4: **Reinforcement learning** (MT10, 10 tasks) and **image classification** (CelebA, 40-task).

**CelebA.** CelebA is a widely used face attributes dataset containing over 200,000 images annotated with 40 attributes. Recently, it has been adopted as a 40-task MTL benchmark to

evaluate a model’s ability to handle a large number of tasks. The results, presented in Table ??, are averaged over three random seeds. While IMGrad does not achieve the best performance, it consistently ranks among the top methods, underscoring the importance of imbalance sensitivity.

## 6.2 Reinforcement Learning

Reinforcement learning is another domain where MTL is often essential, as it seeks to acquire a policy capable of succeeding across various manipulation tasks. To evaluate the generalizability of our proposed method, we use CAGrad as the baseline and conduct experiments on the MT10 environment from the Meta-World benchmark [Yu *et al.*, 2020b]. The results, presented in Table ??, report the average success rate on the validation set over 10 random seeds. Consistent with the improvements observed in supervised learning evaluations, IMGrad enhances CAGrad by over 0.10, achieving SOTA performance on this benchmark. It is worth noting that Nash-MTL does not provide an official implementation for reinforcement learning benchmarks. As a result, we did not augment it for evaluation in this context.

## 7 Conclusion

In this paper, we begin by empirically demonstrating the significance of addressing the imbalance issue in optimization-based MTL. We assert that incorporating imbalance-sensitivity is crucial for avoiding Pareto failures and promoting balanced individual progress. Building upon this motivation, we propose IMGrad, a method derived from a projection norm constraint, which is further simplified as an adaptive balancer between decoupled objectives. Through extensive experiments, we validate the effectiveness of our proposed approach. We believe that our explicit emphasis on the imbalance issue, rather than the conflict issue, provides valuable insights for the future development of optimization-based MTL.

## Acknowledgements

We thank anonymous reviewers for their valuable comments.

## References

- [Chen *et al.*, 2018] Zhao Chen, Vijay Badrinarayanan, Chen-Yu Lee, and Andrew Rabinovich. Gradnorm: Gradient normalization for adaptive loss balancing in deep multitask networks. In *International conference on machine learning*, pages 794–803. PMLR, 2018.
- [Chen *et al.*, 2023] Zitian Chen, Yikang Shen, Mingyu Ding, Zhenfang Chen, Hengshuang Zhao, Erik G Learned-Miller, and Chuang Gan. Mod-squad: Designing mixtures of experts as modular multi-task learners. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11828–11837, 2023.
- [Cordts *et al.*, 2016] Marius Cordts, Mohamed Omran, Sebastian Ramos, Timo Rehfeld, Markus Enzweiler, Rodrigo Benenson, Uwe Franke, Stefan Roth, and Bernt Schiele. The cityscapes dataset for semantic urban scene understanding. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3213–3223, 2016.
- [Diamond and Boyd, 2016] Steven Diamond and Stephen Boyd. Cvxpy: A python-embedded modeling language for convex optimization. *The Journal of Machine Learning Research*, 17(1):2909–2913, 2016.
- [Fan *et al.*, 2022] Zhiwen Fan, Rishov Sarkar, Ziyu Jiang, Tianlong Chen, Kai Zou, Yu Cheng, Cong Hao, Zhangyang Wang, et al. M<sup>3</sup>vit: Mixture-of-experts vision transformer for efficient multi-task learning with model-accelerator co-design. *Advances in Neural Information Processing Systems*, 35:28441–28457, 2022.
- [Fernando *et al.*, 2023] Heshan Devaka Fernando, Han Shen, Miao Liu, Subhajit Chaudhury, Keerthiram Murugesan, and Tianyi Chen. Mitigating gradient bias in multi-objective learning: A provably convergent approach. In *The Eleventh International Conference on Learning Representations*, 2023.
- [Gao *et al.*, 2019] Yuan Gao, Jiayi Ma, Mingbo Zhao, Wei Liu, and Alan L Yuille. Nddr-cnn: Layerwise feature fusing in multi-task cnns by neural discriminative dimensionality reduction. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 3205–3214, 2019.
- [He *et al.*, 2021] Junxian He, Chunting Zhou, Xuezhe Ma, Taylor Berg-Kirkpatrick, and Graham Neubig. Towards a unified view of parameter-efficient transfer learning. *arXiv preprint arXiv:2110.04366*, 2021.
- [Heuer *et al.*, 2021] Falk Heuer, Sven Mantowsky, Saqib Bukhari, and Georg Schneider. Multitask-centernet (mcn): Efficient and diverse multitask learning using an anchor free approach. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 997–1005, 2021.
- [Jaggi, 2013] Martin Jaggi. Revisiting frank-wolfe: Projection-free sparse convex optimization. In *International conference on machine learning*, pages 427–435. PMLR, 2013.
- [Kokkinos, 2017] Iasonas Kokkinos. Ubertnet: Training a universal convolutional neural network for low-, mid-, and high-level vision using diverse datasets and limited memory. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 6129–6138, 2017.
- [Liu *et al.*, 2015] Ziwei Liu, Ping Luo, Xiaogang Wang, and Xiaoou Tang. Deep learning face attributes in the wild. In *Proceedings of the IEEE international conference on computer vision*, pages 3730–3738, 2015.
- [Liu *et al.*, 2019] Shikun Liu, Edward Johns, and Andrew J Davison. End-to-end multi-task learning with attention. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 1871–1880, 2019.
- [Liu *et al.*, 2021a] Bo Liu, Xingchao Liu, Xiaojie Jin, Peter Stone, and Qiang Liu. Conflict-averse gradient descent for multi-task learning. *Advances in Neural Information Processing Systems*, 34:18878–18890, 2021.
- [Liu *et al.*, 2021b] Liyang Liu, Yi Li, Zhanghui Kuang, J Xue, Yimin Chen, Wenming Yang, Qingmin Liao, and Wayne Zhang. Towards impartial multi-task learning. In *International Conference on Learning Representations*, 2021.
- [Liu *et al.*, 2021c] Liyang Liu, Yi Li, Zhanghui Kuang, Jing-Hao Xue, Yimin Chen, Wenming Yang, Qingmin Liao, and Wayne Zhang. Towards impartial multi-task learning. In *International Conference on Learning Representations*, 2021.
- [Liu *et al.*, 2022] Haokun Liu, Derek Tam, Mohammed Muqeeth, Jay Mohta, Tenghao Huang, Mohit Bansal, and Colin A Raffel. Few-shot parameter-efficient fine-tuning is better and cheaper than in-context learning. *Advances in Neural Information Processing Systems*, 35:1950–1965, 2022.
- [Liu *et al.*, 2023] Bo Liu, Yihao Feng, Peter Stone, and Qiang Liu. Famo: Fast adaptive multitask optimization. *arXiv preprint arXiv:2306.03792*, 2023.
- [Maninis *et al.*, 2019] Kevis-Kokitsi Maninis, Ilija Radosavovic, and Iasonas Kokkinos. Attentive single-tasking of multiple tasks. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 1851–1860, 2019.
- [Navon *et al.*, 2022] Aviv Navon, Aviv Shamsian, Idan Achituve, Haggai Maron, Kenji Kawaguchi, Gal Chechik, and Ethan Fetaya. Multi-task learning as a bargaining game. In *International Conference on Machine Learning*, pages 16428–16446. PMLR, 2022.
- [Sener and Koltun, 2018] Ozan Sener and Vladlen Koltun. Multi-task learning as multi-objective optimization. *Advances in neural information processing systems*, 31, 2018.
- [Senushkin *et al.*, 2023] Dmitry Senushkin, Nikolay Patakin, Arseny Kuznetsov, and Anton Konushin. Independent component alignment for multi-task learning. In *Proceedings of*



*the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 20083–20093, 2023.

- [Silberman *et al.*, 2012] Nathan Silberman, Derek Hoiem, Pushmeet Kohli, and Rob Fergus. Indoor segmentation and support inference from rgbd images. In *Computer Vision–ECCV 2012: 12th European Conference on Computer Vision, Florence, Italy, October 7–13, 2012, Proceedings, Part V 12*, pages 746–760. Springer, 2012.
- [Yang and Hospedales, 2016] Yongxin Yang and Timothy Hospedales. Deep multi-task representation learning: A tensor factorisation approach. *arXiv preprint arXiv:1605.06391*, 2016.
- [Ye and Xu, 2022] Hanrong Ye and Dan Xu. Inverted pyramid multi-task transformer for dense scene understanding. In *European Conference on Computer Vision*, pages 514–530. Springer, 2022.
- [Yu *et al.*, 2020a] Tianhe Yu, Saurabh Kumar, Abhishek Gupta, Sergey Levine, Karol Hausman, and Chelsea Finn. Gradient surgery for multi-task learning. *Advances in Neural Information Processing Systems*, 33:5824–5836, 2020.
- [Yu *et al.*, 2020b] Tianhe Yu, Deirdre Quillen, Zhanpeng He, Ryan Julian, Karol Hausman, Chelsea Finn, and Sergey Levine. Meta-world: A benchmark and evaluation for multi-task and meta reinforcement learning. In *Conference on robot learning*, pages 1094–1100. PMLR, 2020.
- [Zhou *et al.*, ] Zhipeng Zhou, Liu Liu, Peilin Zhao, and Wei Gong. Pareto deep long-tailed recognition: A conflict-averse solution. In *The Twelfth International Conference on Learning Representations*.

# Appendix for ‘Injecting Imbalance Sensitivity for Multi-Task Learning’

## A Implementation Details

### A.1 MTL Baselines

**CAGrad** [Liu *et al.*, 2021a]: CAGrad strikes a balance between Pareto optimality and globe convergence by regulating the combined gradients in proximity to the average gradient.:

$$\max_{d \in \mathbb{R}^m} \min_{\omega \in \mathcal{W}} g_{\omega}^{\top} d \quad \text{s.t. } \|d - g_0\| \leq c \|g_0\| \quad (14)$$

In our experimental setup, we utilize the official code <sup>5</sup> that encompasses the implementations of MGDA, PCGrad, and CAGrad. Additionally, we extend this implementation to include IMGrad. For more comprehensive information, please consult the official implementation.

**Nash-MTL** [Navon *et al.*, 2022]: Nash-MTL provides the individual progress guarantee via the following objective:

$$\min_{\omega} \sum_i \beta_i(\omega) + \varphi_i(\omega) \quad (15)$$

$$\text{s.t. } \forall i, -\varphi_i(\omega) \leq 0, \quad \omega_i > 0. \quad (16)$$

where  $\varphi_i(\omega) = \log(\omega_i) + \log(g_i^{\top} G \omega)$ ,  $G = [g_1, g_2, \dots, g_K]$ . As demonstrated, the individual progress is ensured through the projection, subject to the constraint  $\beta_i = g_i^{\top} G \omega \geq \frac{1}{\omega_i}$ . However, in cases where conflicting individuals arise, such as when  $g_i$  conflicts with  $g_j$ , the official implementation <sup>6</sup> of Nash-MTL opts to skip the current step, which poses a potential risk in handling these conflicting individuals. It is important to acknowledge that the experiments conducted on CityScapes in the main text regarding Nash-MTL were performed using our re-implementation, as the official version does not offer the corresponding implementation.

**Our Extension to Nash-MTL:** As previously stated, Nash-MTL opts to skip the current step and retain the weights from the previous step only when negative terms exist in  $g_i^{\top} G$ , indicating a conflict issue in the current step. Moreover, the co-existence of both imbalance and conflict issues necessitates a specialized design for MTL. By assuming H-Lipschitz continuity on gradients, we expect a similar imbalance status between consecutive steps. To address the conflict issue, we incorporate imbalance sensitivity, which allows greater weighting on  $\sum_i g_i^{\top} G \omega$ .

<sup>5</sup><https://github.com/Cranial-XIX/CAGrad>

<sup>6</sup><https://github.com/AvivNavon/nash-mtl/tree/main>

### A.2 Datasets

**NYUv2:** NYUv2 is an indoor scene dataset comprising 1449 RGBD images, accompanied by dense per-pixel labeling encompassing 13 classes. This dataset is commonly employed for tasks such as semantic segmentation, depth estimation, and surface normal prediction.

**CityScapes:** CityScapes is a renowned benchmark dataset for multi-task learning, comprising 5000 high-resolution street-view images accompanied by dense per-pixel labels for tasks such as semantic segmentation and depth estimation. In accordance with the previous configuration [Liu *et al.*, 2021a; Navon *et al.*, 2022], we resized the images to 128\*256 dimensions prior to inputting them into the model, aiming to enhance computational efficiency.

**MT10:** MT10 is a widely recognized benchmark for multi-task reinforcement learning, encompassing 10 robot manipulation tasks. For visual references, please consult [Liu *et al.*, 2021a]. Our implementation builds upon the official implementation of CAGrad and leverages the well-established MTRL environment <sup>7</sup>.

### A.3 Synthetic Examples

We adopt the toy optimization example introduced in [Liu *et al.*, 2021a; Navon *et al.*, 2022], which involves two objectives that consider the variables  $\vartheta = (\vartheta_1, \vartheta_2)$ :

$$\mathcal{L}_1(\vartheta) = c_1(\vartheta) f_1(\vartheta) + c_2(\vartheta) g_1(\vartheta) \quad (17)$$

$$\mathcal{L}_2(\vartheta) = c_1(\vartheta) f_2(\vartheta) + c_2(\vartheta) g_2(\vartheta) \quad \text{where}$$

$$f_1(\vartheta) = \log(\max(|0.5(-\vartheta_1 - 7) - \tanh(\vartheta_2)|, 5.e - 6)) + 6,$$

$$f_2(\vartheta) = \log(\max(|0.5(-\vartheta_1 + 3) - \tanh(\vartheta_2) + 2|, 5.e - 6)) + 6,$$

$$g_1(\vartheta) = ((-\vartheta_1 + 7)^2 + 0.1 * (-\vartheta_2 - 8)^2) / 10 - 20,$$

$$g_2(\vartheta) = ((-\vartheta_1 - 7)^2 + 0.1 * (-\vartheta_2 - 8)^2) / 10 - 20,$$

$$c_1(\vartheta) = \max(\tanh(0.5 * \vartheta_2), 0),$$

$$c_2(\vartheta) = \max(\tanh(-0.5 * \vartheta_2), 0).$$

The initial points in Figure 2 of the main text are  $\{(-8.5, 7.5), (-8.5, 5), (0, 0), (9, 9), (10, -8)\}$ , while those in Figure 4 are  $\{(-8.5, 7.5), (0, 8), (5, 9)\}$ .

## B Convergence Analysis

Let’s first recall the convergence analysis of CAGrad in Theorem 4.

<sup>7</sup><https://github.com/facebookresearch/mtrl>

**Theorem 4.** (Convergence of CAGrad) With a fix step size  $\alpha$  and the assumption of  $H$ -Lipschitz ( $0 < H \leq 1/\alpha$ ) on gradients, i.e.,  $\|\nabla \mathcal{L}_i(\theta) - \nabla \mathcal{L}_i(\theta')\| \leq H \|\theta - \theta'\|$  for  $i = 1, 2, \dots, K$ . Denote  $\mathbf{d}^*(\theta_t)$  as the optimization direction of IMGrad at step  $t$ , then we have:

(1) If  $0 \leq c < 1$ , then CAGrad converges to stationary points of  $\mathcal{L}_0$  convergence rate in that

$$\sum_{t=0}^T \|\mathbf{g}_0(\theta_t)\|^2 \leq \frac{2(\mathcal{L}(\theta_0) - \mathcal{L}(\theta_{T+1}))}{\alpha(1 - c^2)} \quad (18)$$

*Proof.*

$$\begin{aligned} \mathcal{L}(\theta_{t+1}) - \mathcal{L}(\theta_t) &= \mathcal{L}(\theta_t - \alpha \mathbf{d}^*(\theta_t)) - \mathcal{L}(\theta_t) \\ &\leq -\alpha \mathbf{g}_0(\theta_t)^\top \mathbf{d}^*(\theta_t) + \frac{H\alpha^2}{2} \|\mathbf{d}^*(\theta_t)\|^2 \\ &\leq -\alpha \mathbf{g}_0(\theta_t)^\top \mathbf{d}^*(\theta_t) + \frac{\alpha}{2} \|\mathbf{d}^*(\theta_t)\|^2 \\ &\quad // \alpha \leq 1/H \\ &\leq -\frac{\alpha}{2} (\|\mathbf{g}_0(\theta_t)\|^2 + \|\mathbf{d}^*(\theta_t)\|^2 \\ &\quad - \|\mathbf{g}_0(\theta_t) - \mathbf{d}^*(\theta_t)\|^2) + \frac{\alpha}{2} \|\mathbf{d}^*(\theta_t)\|^2 \\ &= -\frac{\alpha}{2} (\|\mathbf{g}_0(\theta_t)\|^2 - \|\mathbf{g}_0(\theta_t) - \mathbf{d}^*(\theta_t)\|^2) \\ &\leq -\frac{\alpha}{2} (1 - c^2) \|\mathbf{g}_0(\theta_t)\|^2 \\ &\quad // \text{CAGrad's Bound} \end{aligned}$$

Using telescoping sums, we have  $\mathcal{L}(\theta_{t+1}) - \mathcal{L}(0) = -(\alpha\beta/2)(1 - c^2) \sum_{t=0}^T \|\mathbf{g}_m(\theta_t)\|^2$ . Thus, we further have

$$\begin{aligned} \min_{t \leq T} \|\mathbf{g}_0(\theta_t)\|^2 &\leq \frac{1}{T+1} \sum_{t=0}^T \|\mathbf{g}_0(\theta_t)\|^2 \\ &\leq \frac{2(\mathcal{L}(\theta_0) - \mathcal{L}(\theta_{T+1}))}{\alpha(1 - c^2)(T+1)} \quad (19) \end{aligned}$$

Generally, if  $\mathcal{L}$  is lower bounded, then  $\min_{t \leq T} \|\mathbf{g}_0(\theta_t)\|^2 = \mathcal{O}(1/T)$ . Besides, the upper bound of  $\min_{t \leq T} \|\mathbf{g}_0(\theta_t)\|^2$  is harnesses by  $\beta$ , which is positively related to the imbalance ratio of individuals.  $\square$

Take a close at the above convergence analysis, we can find that the proof only leverages the property of  $\|\mathbf{g}_0(\theta_t) - \mathbf{d}^*(\theta_t)\| \leq c \|\mathbf{g}_0(\theta_t)\|$ , ignoring the solved  $\mathbf{g}_\omega$ , which is essential for mitigating Pareto failures and imbalanced individual progress claimed in the main text. Hence we re-organize the proof as following:

*Proof.*

$$\begin{aligned} \mathcal{L}(\theta_{t+1}) - \mathcal{L}(\theta_t) &= \mathcal{L}(\theta_t - \alpha \mathbf{d}^*(\theta_t)) - \mathcal{L}(\theta_t) \\ &\leq -\alpha \mathbf{g}_0(\theta_t)^\top \mathbf{d}^*(\theta_t) + \frac{H\alpha^2}{2} \|\mathbf{d}^*(\theta_t)\|^2 \\ &\leq -\alpha \mathbf{g}_0(\theta_t)^\top \mathbf{d}^*(\theta_t) + \frac{\alpha}{2} \|\mathbf{d}^*(\theta_t)\|^2 \\ &\quad // \alpha \leq 1/H \\ &= -\alpha(1 + c \cos \varphi) \|\mathbf{g}_0(\theta_t)\|^2 + \frac{\alpha}{2} \|\mathbf{g}_0(\theta_t)\|^2 \\ &\quad + \frac{\alpha}{2} \|\mathbf{d}^*(\theta_t)\|^2 - \frac{\alpha}{2} \|\mathbf{g}_0(\theta_t)\|^2 \\ &\leq -\alpha(1 + c \cos \varphi) \|\mathbf{g}_0(\theta_t)\|^2 + \frac{\alpha}{2} \|\mathbf{g}_0(\theta_t)\|^2 \\ &\quad + \frac{\alpha}{2} \|\mathbf{d}^*(\theta_t) - \mathbf{g}_0(\theta_t)\|^2 \\ &= -\alpha(1 + c \cos \varphi) \|\mathbf{g}_0(\theta_t)\|^2 + \frac{\alpha}{2} \|\mathbf{g}_0(\theta_t)\|^2 \\ &\quad + \frac{\alpha c^2}{2} \|\mathbf{g}_0(\theta_t)\|^2 \\ &= -\frac{\alpha}{2} (1 - c^2 + 2c \cos \varphi) \|\mathbf{g}_0(\theta_t)\|^2 \\ &\quad // \text{Our Bound} \end{aligned}$$

where  $\varphi$  is the angle between  $\mathbf{g}_0(\theta_t)$  and  $\mathbf{g}_\omega(\theta_t)$ . And it should noted that although  $\|\mathbf{g}_0(\theta_t) - \mathbf{d}^*(\theta_t)\| \leq c \|\mathbf{g}_0(\theta_t)\|$  is constrained in CAGrad, the practical implementation always satisfied  $\|\mathbf{g}_0(\theta_t) - \mathbf{d}^*(\theta_t)\| = c \|\mathbf{g}_0(\theta_t)\|$ , thus we directly apply it in the above analysis. According to the final derived equation, the convergence rate of CAGrad is harnesses by  $\cos \varphi$ . Specifically, when all individuals are balanced, i.e.,  $\cos \varphi = 1$ , then our bound degrades to the CAGrad's bound.

In particular, we provide a tighter bound when  $0^\circ < \varphi \leq 90^\circ$ , but a looser bound when  $\varphi \geq 90^\circ$  (extreme imbalance). However, the percentage of extreme cases is very small (see Figure 5(b) in the main text) and therefore does not affect its convergence generally. Besides, according to the simplified objective in Eqn. 11 in the main text, IMGrad tends to choose a larger  $\varphi$ , leading to a slower convergence to guarantee a more balanced individual progress.  $\square$

## C Additional Experiments

### C.1 Imbalance Examination

We provide the statistical imbalance ratios of MTL approaches on CityScapes in Figure 7.

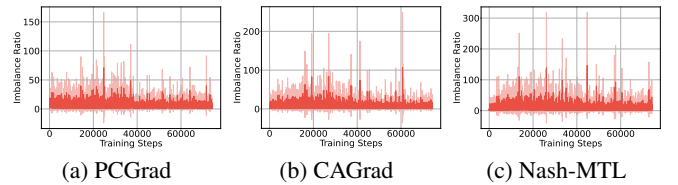


Figure 7: Statistical imbalance ratios of MTL approaches.

## C.2 Sensitivity of $\mu$

One concern arises regarding whether  $\mu$  ( $\cos\theta$ ) effectively adjusts the combined gradient. In other words, does  $\mu$  exhibit sufficient sensitivity to address varying imbalance circumstances? To address this question, we provide statistics of  $\mu$  for both NYUv2 and CityScapes datasets, which are presented in Figure 8. The results demonstrate that  $\mu$  exhibits a wide range of values that effectively cater to the manipulation requirements.

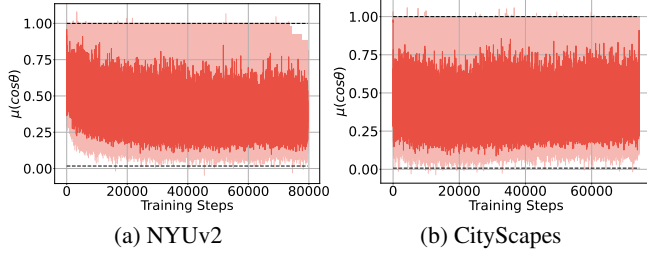


Figure 8: Sensitivity examinations of  $\mu$ .

## C.3 Verification of Negative Correlation between Imbalance Ratio and $\cos\theta$

To justify our choice of using  $\cos\theta$  as an alternative for the imbalance ratio in implementing IMGrad, it is necessary to establish their negative correlation. To demonstrate this, we track the imbalance ratio and  $\cos\theta$  while running CAGrad on NYUv2 and CityScapes datasets, presenting their statistical results in Figure 9. The results indicate a positive correlation between  $1/\text{Imb.}$  and  $\cos\theta$  during optimization, regardless of whether it involves two (CityScapes) or three (NYUv2) tasks. Additionally, we observe that the values of  $\cos\theta$  are confined within the range of  $[0, 1]$ , while the imbalance ratio can vary significantly. This discrepancy in stability makes  $\cos\theta$  a more suitable choice for modulation.

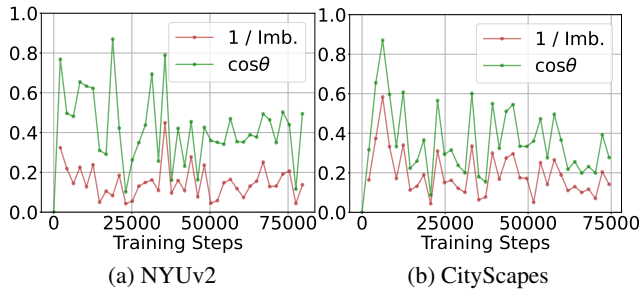


Figure 9: Negative correlation examination between imbalance ratio and  $\cos\theta$ .

## C.4 Alternatives of $\mu$

There are several alternatives to quantify the imbalance ratio among tasks, we provide the following choices:

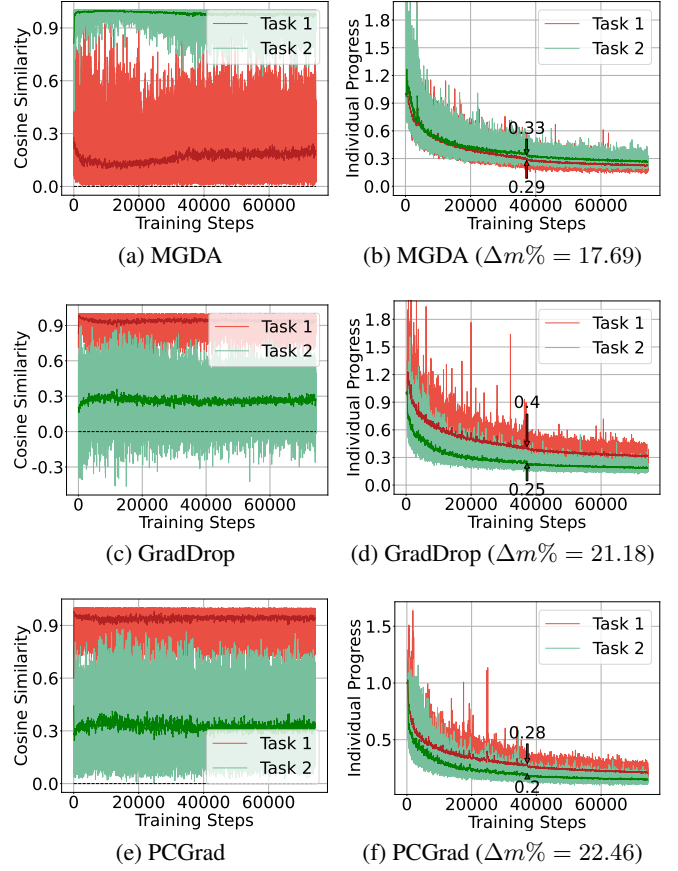


Figure 10: Pareto failures and individual progress examination. The top row presents the gradient similarity, while the bottom row illustrates the individual progress for MGDA and GradDrop, respectively.

- **Direct Calculation (DC):** DC is the standard operation to obtain the imbalance ratio by directly calculate the their norm ratio, i.e.,  $\|g_i\| / \|g_j\|$ . Specifically, to control the range of  $\mu$ , we adopt  $\mu = 1/(1 + \log(\|g_i\| / \|g_j\|))$ .
- **Least Mean (LM):** LM quantifies the imbalance ratio by calculating the cosine similarity between  $g_0$  and the individual with the smallest norm. The utilization of  $\theta$  in LM is depicted in Figure 11 (b).
- **Plumbline Mean (PM):** PM is just the operation that adopted in the main text, and is illustrated in Figure 11 (a).

The corresponding results are presented in Table 5, revealing that PM exhibits the most favorable overall performance. However, LM demonstrates competitiveness in the context of semantic segmentation tasks, while DC shows moderate performance. This outcome can be attributed to the wide range of imbalance ratios, which makes it challenging to achieve adaptive balancing of the decoupled objectives. The inferior performance of LM can be attributed to the variation of  $\cos\theta$ , which can range between -1 and 1, making it difficult to tune hyper-parameters effectively.



Table 5:  $\mu$  alternatives comparison on CityScapes. The GPU time reported here refers to the cumulative duration of the entire run.

| Method | Segmentation    |           | Depth          |         | $\Delta$ m% $\downarrow$ | GPU Time (h) $\downarrow$ |
|--------|-----------------|-----------|----------------|---------|--------------------------|---------------------------|
|        | (Higher Better) |           | (Lower Better) |         |                          |                           |
|        | mIoU            | Pix. Acc. | Abs Err        | Rel Err |                          |                           |
| CAGrad | 75.16           | 93.48     | 0.0141         | 37.60   | 11.58                    | 13.83                     |
| DC     | 74.20           | 93.36     | 0.0132         | 34.27   | 7.23                     | 16.90                     |
| LM     | 74.85           | 93.47     | 0.0132         | 36.56   | 8.83                     | 17.40                     |
| PM     | 74.84           | 93.39     | 0.0132         | 33.99   | 6.58                     | 20.25                     |

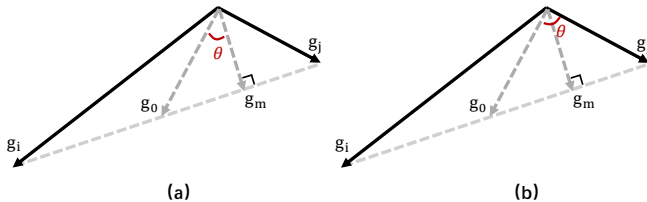


Figure 11: The illustration of different choice of  $\mu$ . (a) represents the PM, while (b) represents the LM.

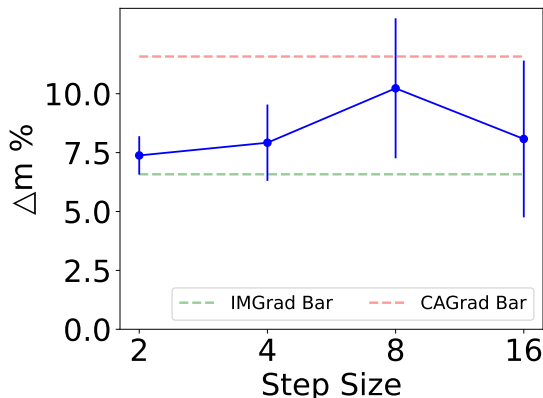


Figure 12: The impact of step size to the overall performance.

### C.5 Speedup Analysis

While the primary focus of this paper does not revolve around achieving speedup, it is worth mentioning that general speedup approaches, such as sampling a subset of tasks [Liu *et al.*, 2021a], utilizing feature-level gradients [Sener and Koltun, 2018], and updating  $\omega$  periodically instead of every iteration [Navon *et al.*, 2022], continue to be applicable to methods augmented with IMGrad.

On the other hand, based on the findings presented in Table 5, we have introduced two additional computation-efficient alternatives for  $\mu$ . These alternatives offer reduced GPU time requirements while still maintaining an acceptable level of performance sacrifice.

We have also implemented a commonly used practical speedup approach, i.e., updating the weights less frequently. In this regard, we have chosen the update step sizes from the set [2, 4, 8, 16], and the corresponding results are presented in Figure 12. Generally, as the step size increases, the overall performance ( $\Delta m\%$ ) tends to deteriorate, although it still

demonstrates improvements compared to the vanilla CAGrad. Interestingly, the performance appears to be better when the step size is set to 16 compared to 4 and 8, suggesting that frequent weight updates may not be necessary. However, further exploration of this aspect is left for future research.

## D Visualizations

### D.1 Synthetic Examples

In order to further illustrate the effectiveness of IMGrad in different imbalance scenarios, we present additional comparative synthetic examples in Table 6. The results reveal that LS, PCGrad, and CAGrad exhibit failure cases across nearly all scenarios, while IMTL [?] and Nash-MTL fail to reach the global optimum, although they do converge to the Pareto stationary. In contrast, IMGrad consistently achieves the optimal solution in all scenarios, demonstrating its robust and stable performance. This underscores the significance of imbalance-sensitivity in enhancing optimization outcomes.

### D.2 Pareto Failure and Individual Progress Examinations

Here we aim to provide a visual understanding of the advantages brought by IMGrad in terms of reducing Pareto failures and achieving a balanced individual progress. In Figure 13, concerning CAGrad, the IMGrad-augmented version can effectively avoid Pareto failures and optimizes both individuals in a more balanced manner by seeking a direction that exhibits rough similarities among all individuals. In the case of Nash-MTL, IMGrad successfully reduces the number of Pareto failures from 249 to 161 and further narrows the gap in individual progress.

Besides, we present supplementary analysis on the Pareto failure and individual progress using MGDA, PCGrad, and GradDrop, and showcase the outcomes in Figure 10. As depicted, both MGDA, PCGrad and GradDrop exhibit pronounced imbalance issues based on their gradient similarities, leading to unsatisfactory and imbalanced individual progress. These findings align with the tendencies and indications stated above, further supporting the claims made.

## E Past Imbalance-Sensitive MTL

As we have discussed Nash-MTL previously, here we mainly introduce another two imbalance-sensitive MTL methods, i.e., IMTL [?], and FAMO [Liu *et al.*, 2023].

**IMTL:** IMTL addresses the imbalance issue by integrating two key strategies. It employs gradient balance to adjust shared

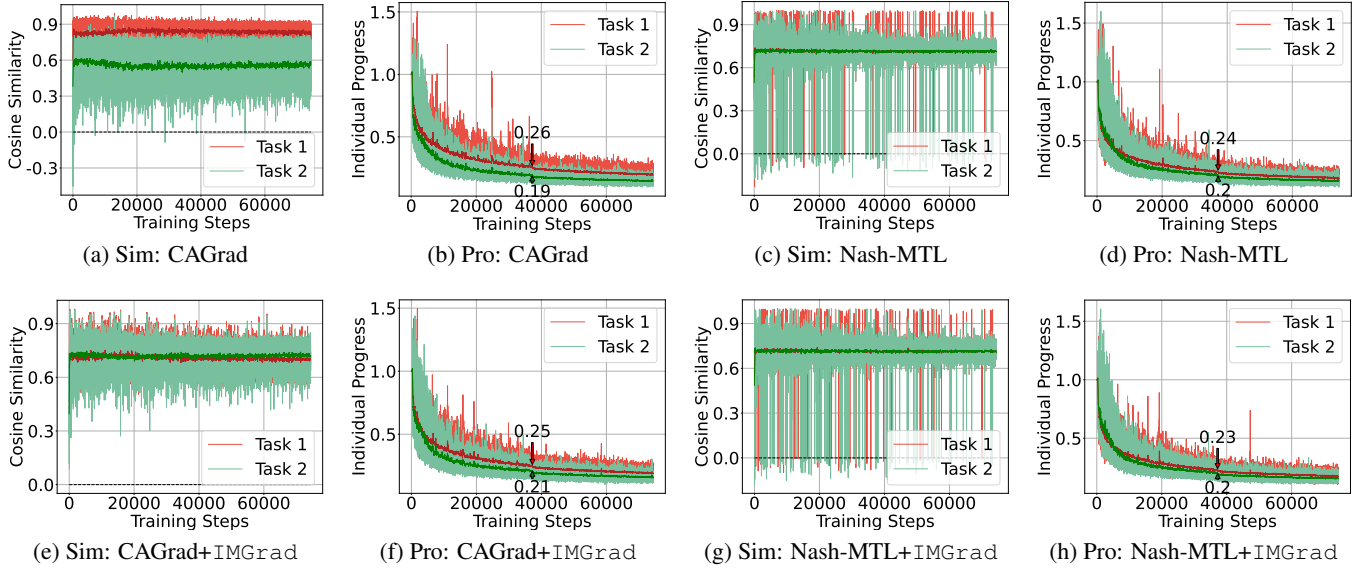


Figure 13: Pareto failures and individual progress examinations. ‘Sim’ and ‘Pro’ are short for similarity and progress, respectively.

parameters without bias towards any task, using a closed-form solution that ensures equal projections of the aggregated gradient onto individual task gradients. Additionally, it applies loss balance to dynamically weigh task losses, preventing any single task’s loss scale from dominating the training process. By combining these strategies into a hybrid approach, IMTL achieves scale invariance, maintaining robust performance regardless of the magnitude of task losses.

**FAMO:** FAMO is a gradient-free optimization-based MTL method, which introduces a dynamic weighting method that adapts to the performance of individual tasks, ensuring that each task progresses at a similar rate without the need for extensive computational resources. FAMO operates with a constant space and time complexity of  $\mathcal{O}(1)$  per iteration, which is a significant advantage over traditional methods that require  $\mathcal{O}(k)$  space and time complexities, where  $k$  represents the number of tasks. This efficiency makes FAMO particularly well-suited for large-scale multitask scenarios.

## F Limitations

Although our proposed method has demonstrated effectiveness, it is important to acknowledge its limitations. First, the method is only applicable to scenarios with a decoupled objective, such as CAGrad and Nash-MTL, since IMGrad aims to dynamically balance objectives during optimization. Second, the introduction of additional computations to calculate  $\mathbf{g}_m$  through MGDA represents another limitation. Although reducing the iterations of MGDA can mitigate this issue, this paper currently does not provide the corresponding performance and efficiency reports.

Table 6: Comparison of MTL optimization methods on synthetic two-task benchmark.  $(a_1, a_2)$  denotes  $a_1 * \mathcal{L}_1 + a_2 * \mathcal{L}_2$ .

