Three-Way Trade-Off in Multi-Objective Learning: Optimization, Generalization and Conflict-Avoidance

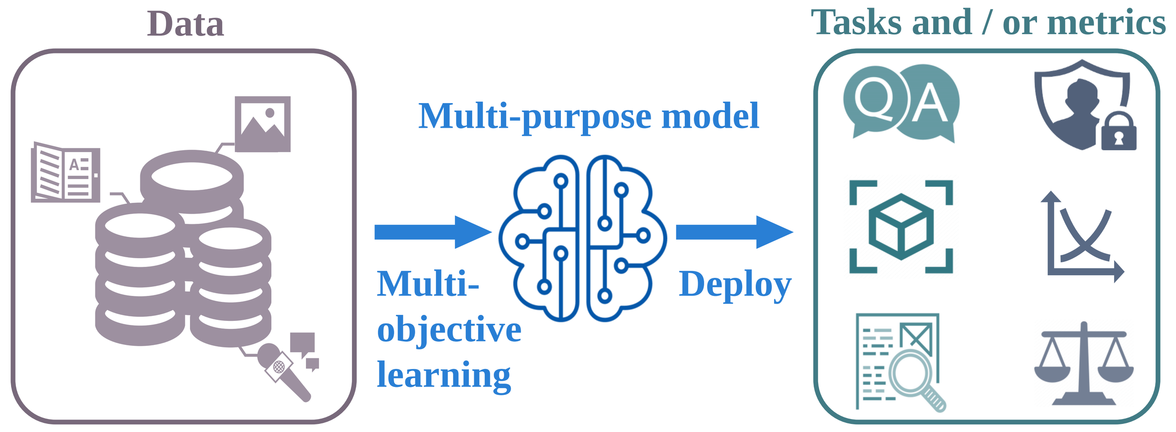
Lisha Chen*[1], Heshan Fernando*[1], Yiming Ying[2], Tianyi Chen[1]

[1]Rensselaer Polytechnic Institute

[2]University of Sydney

Paper Code

1. Context & Motivation: Multi-Objective Learning



Minimize multiple objectives simultaneously

$$\min_{x\in\mathbb{R}^d} F_S(x) := [f_{S,1}(x), f_{S,2}(x), \ldots, f_{S,M}(x)]$$

x : model parameter
S: training data

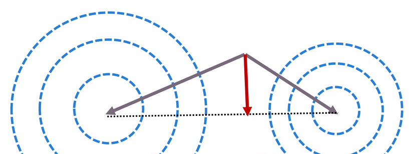
Multi-gradient descent Algorithm (MGDA) for MOL

Idea: maximize the worst descent amount among all objectives to avoid optimization conflict (getting stuck in the valley in the toy example)

$$\max_{d\in\mathbb{R}^d}\min_{\lambda\in\Delta^M} -\langle\nabla F_S(x)\lambda, d\rangle + \frac{1}{2}\rho\|\lambda\|^2 - \frac{1}{2}\|d\|^2$$

Reformulated as:

$$\max_{\lambda\in\Delta^M}\min_{d\in\mathbb{R}^d}\langle\nabla F_S(x)\lambda, d\rangle - \frac{1}{2}\rho\|\lambda\|^2 + \frac{1}{2}\|d\|^2$$



Update direction

Find a common descent direction to mitigate optimization conflicts

Conflict avoidant (CA) direction:

$$d(x) = -\nabla F_S(x)\lambda^*_\rho(x) \quad \text{s.t.} \quad \lambda^*_\rho(x) \in \arg\min_{\lambda\in\Delta^M} \|\nabla F_S(x)\lambda\|^2 + \rho\|\lambda\|^2$$

Stochastic variant: Multi-objective gradient with double sampling (MoDo)

Double sampling to mitigate gradient bias

Update CA weight λ (stochastic PGD of the subproblem):

$$\lambda_{t+1} = \Pi_{\Delta^M}\Big(\lambda_t - \gamma_t\big(\nabla F_{z_{t,1}}(x_t)^\top\nabla F_{z_{t,2}}(x_t) + \rho \mathrm{I}\big)\lambda_t\Big)$$

Update model parameter x along the approximate CA direction:

$$x_{t+1} = x_t - \alpha_t\nabla F_{Z_{t+1}}(x_t)\lambda_{t+1}, \quad \text{with} \quad Z_{t+1} = \{z_{t+1,1}, z_{t+1,2}\}$$

$\gamma_t = 0$ reduces to static weighting
$\gamma_t > 0$ approximates MGDA

MoDo interpolates between static weighting & MGDA, controlled by γ_t

2. Empirical Observations

- Dynamic weighting may not outperform the simplest static weighting.
- Generalization errors of dynamic weighting can be larger while optimization errors are similar.

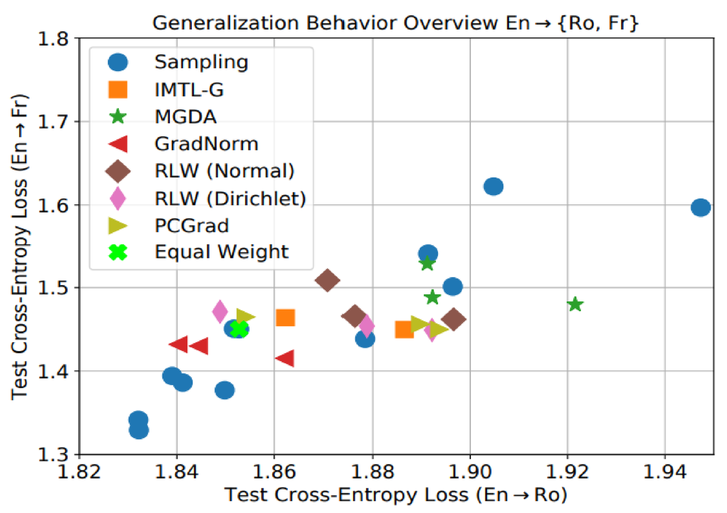


Figure 1: Test performance of MOL algorithms in Multilingual Machine Translation (Xin et al., 2022).

***Q1:** Major sources of errors in dynamic weighting methods?*
***Q2:** Cause of the testing performance degradation of dynamic weighting methods?*

Toy Example Illustration

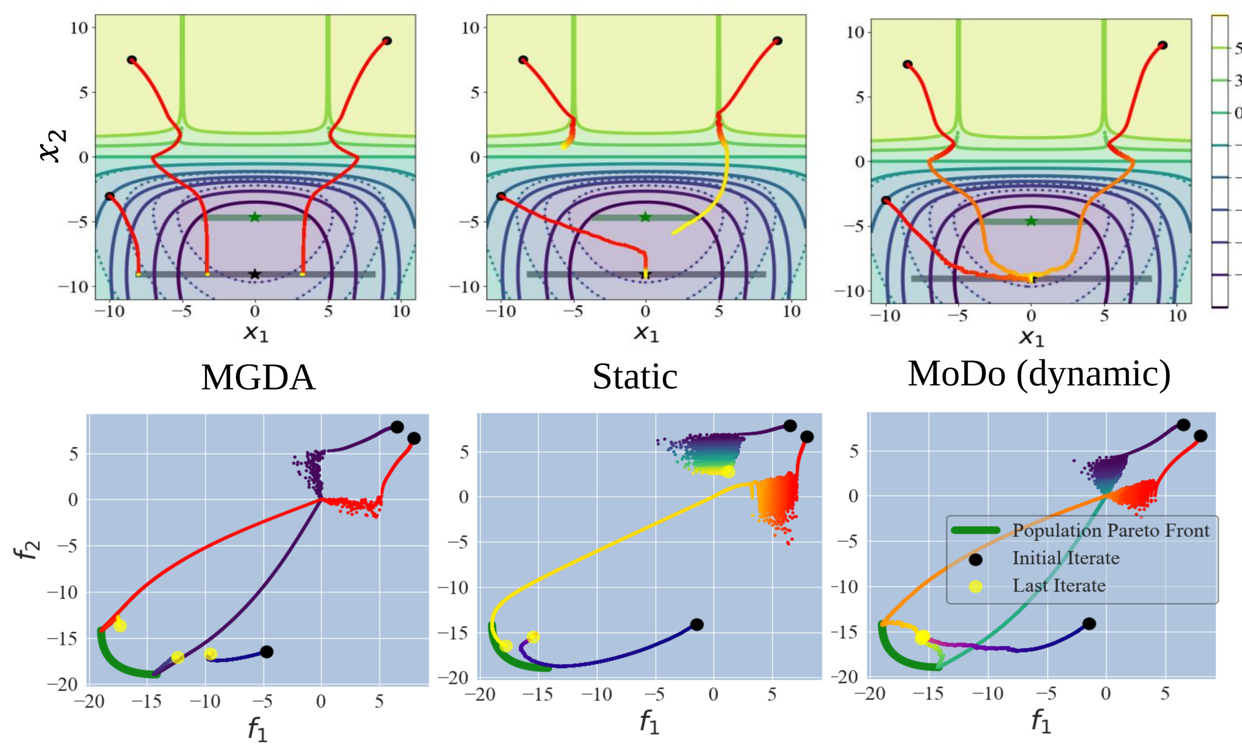
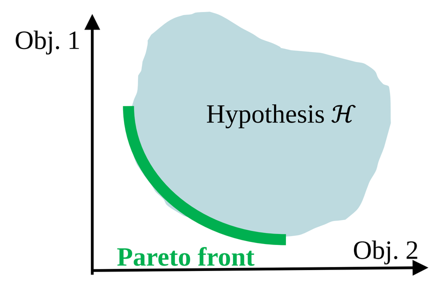


Figure 2: Optimization trajectories of three algorithms on the loss landscape and their convergence to the Pareto fronts.

- Though static weighting cannot avoid gradient conflict (sometimes stuck at local valley), it may still perform well or better than dynamic weighting during testing; while MoDo could prevent achieving the optimal test risk.

3. Theoretical Analysis



Pareto optimality (PO): cannot simultaneously improve all objectives.

Pareto stationarity (PS): $\min_{\lambda\in\Delta^M}\|\nabla F(x)\lambda\|^2 = 0$.

PO ⇒ PS

Risk decomposition

$$\underbrace{\min_{\lambda\in\Delta^M}\|\nabla F(x)\lambda\|}_{\text{PS population risk } R_{\mathrm{pop}}(x)} = \underbrace{\min_{\lambda\in\Delta^M}\|\nabla F(x)\lambda\| - \min_{\lambda\in\Delta^M}\|\nabla F_S(x)\lambda\|}_{\text{PS generalization error } R_{\mathrm{gen}}(x)} + \underbrace{\min_{\lambda\in\Delta^M}\|\nabla F_S(x)\lambda\|}_{\text{PS optimization error } R_{\mathrm{opt}}(x)}$$

$F_S(x)$: empirical objective, average over training data
$F(x)$: population objective, expectation over data distribution

Measure of conflict avoidance (CA):

CA direction distance $\mathcal{E}_{\mathrm{cad}}(x,\lambda) := \|d_\lambda(x) - d(x)\|^2$,

CA weight distance $\mathcal{E}_{\mathrm{caw}}(x,\lambda) := \|\lambda - \lambda^*_\rho(x)\|^2$.

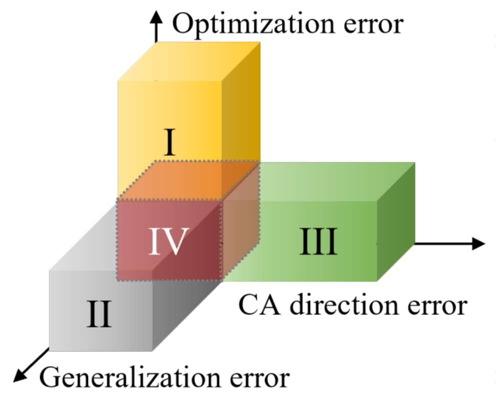
Generalization, optimization and conflict avoidance

Table 1: Comparison of Static & dynamic weighting in three errors.

Assmp	Method	Optimization	Generalization	Risk	CA weight distance
NC, Lip-C, S	Static	$(\alpha T)^{-\frac{1}{2}} + \alpha^{\frac{1}{2}}$	$T^{\frac{1}{2}}n^{-\frac{1}{2}}$	$n^{-\frac{1}{6}}$	$\Theta(1)$
	Dynamic	$(\alpha T)^{-\frac{1}{2}} + \alpha^{\frac{1}{2}} + \gamma^{\frac{1}{2}}$	$T^{\frac{1}{2}}n^{-\frac{1}{2}}$	$n^{-\frac{1}{6}}$	$\gamma\rho^{-1} + \alpha\gamma^{-1}\rho^{-2}$
SC, S	Static	$(1-\alpha)^{\frac{T}{2}} + \alpha^{\frac{1}{2}}$	$n^{-\frac{1}{2}}$	$n^{-\frac{1}{2}}$	$\Theta(1)$
	Dynamic	$\min\{(\alpha T)^{-\frac{1}{2}} + \alpha^{\frac{1}{2}} + \gamma^{\frac{1}{2}} + \rho^{\frac{1}{2}}, (1-\alpha)^{\frac{T}{2}} + \alpha^{\frac{1}{2}} + \gamma T\}$	$\begin{cases} n^{-\frac{1}{2}}, \gamma = \mathcal{O}(T^{-1}) \\ T^{\frac{1}{2}}n^{-\frac{1}{2}}, \text{ o.w.} \end{cases}$	$\begin{cases} n^{-\frac{1}{2}} \\ n^{-\frac{1}{6}} \end{cases}$	$\gamma\rho^{-1} + \alpha\gamma^{-1}\rho^{-2}$

NC/SC: nonconvex/strongly convex
Lip-C: Lipschitz continuous
S: smooth
n: training data size
T: number of iterations
γ: step size to update CA weight
α: step size to update model
ρ: regularization

The Fundamental Three-Way Trade-Off in MOL



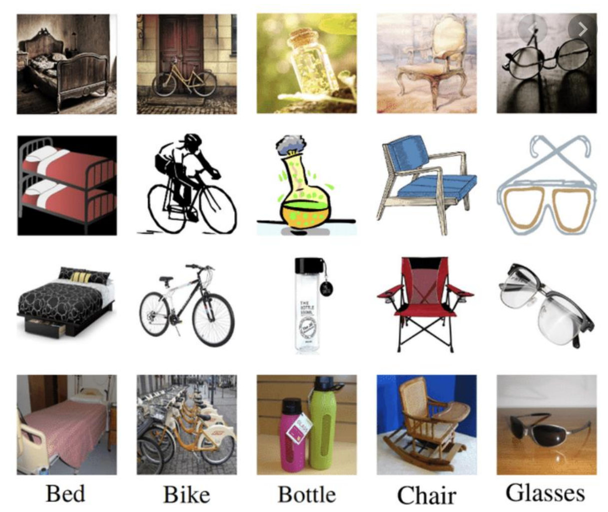
Region	Opt.	Gen.	Conflict	Step sizes α & γ	Iterations T & data size n
I	↑	↓	↓	$\gamma = \omega(\alpha)$	$T = o(n)$
II	↓	↑	↓	$\gamma = \omega(\alpha)$	$T = \omega(n)$
III	↓	↓	↑	$\gamma = o(T^{-1})$	$T = \omega(n)$
IV	↘	↘	↘	$\gamma = \Theta(\alpha^{\frac{1}{2}}) = \Theta(T^{-\frac{1}{4}})$	$T = \Theta(n^{\frac{4}{5}})$

Figure 3: An illustration of three-way trade-off among optimization, generalization, and conflict avoidance in the strongly convex case.
↓: diminishing in an optimal rate w.r.t. n; ↑: growing in a fast rate w.r.t. n;
↘: diminishing w.r.t. n, but not in an optimal rate.

- The three errors can simultaneously diminish, but only at suboptimal rates compared to the optimal rates they can achieve.
- CA direction distance reduction could prevent achieving the optimal test risk.

4. Practical Applications to Multi-Task Learning

Image classification



Semantic scene understanding

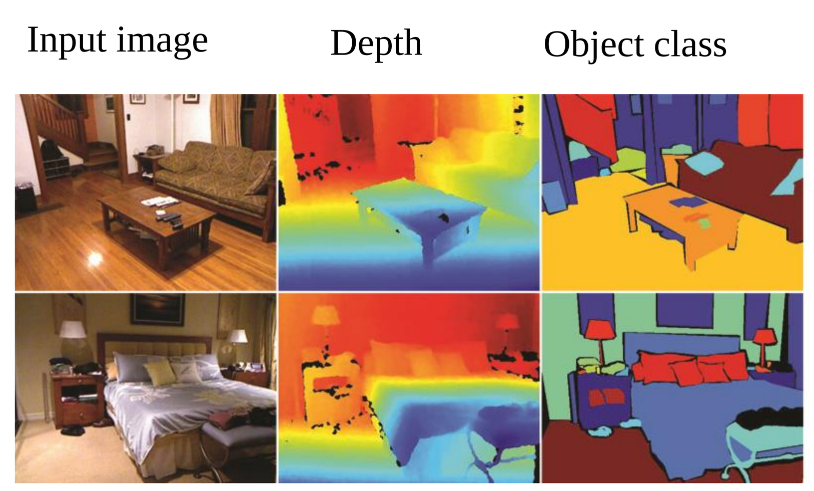


Table 2: Image classification accuracy on Office-home dataset.

Method	Art	Clipart	Product	Real-world	$\Delta\mathcal{A}_{\mathrm{st}}\%\downarrow$	$\Delta\mathcal{A}_{\mathrm{id}}\%\downarrow$
Static (EW)	62.99	76.48	88.45	77.72	0.00	5.02
MGDA-UB (Lin et al., 2022a)	64.32	75.29	89.72	79.35	-1.02	4.04
PCGrad (Yu et al., 2020)	63.94	76.05	88.87	78.27	-0.53	4.51
CAGrad (Liu et al., 2021a)	63.75	75.94	89.08	78.27	-0.48	4.56
MoCo (Fernando et al., 2023)	64.14	**79.85**	89.62	79.57	-2.48	2.68
MoDo (Ours)	**66.22**	78.22	**89.83**	**80.32**	**-3.08**	**2.11**

Check out more results & benchmarks in the paper.

5. Theoretical Applications to MOL Algorithms

Our theoretical framework can be used to analyze other algorithms.

Table 3: Theoretical applications to other MOL algorithms and the three errors.

Algorithm	Bounded function	Opt.	CA dist.	Gen.
MoCo (Fernando et al., 2023, Lemma 2, Thm 2)	✗	$T^{-\frac{1}{10}}$	$T^{-\frac{1}{5}}$	-
MoCo (Fernando et al., 2023, Thm 4)	✓	$T^{-\frac{1}{2}}$	-	-
MoCo (Ours, Thms 4.1-4.3)	✗	$T^{-\frac{1}{8}}$	$T^{-\frac{1}{4}}$	$T^{\frac{1}{2}}n^{-\frac{1}{2}}$
MoDo (Ours, Thms 3.1,3.3,3.5)	✗	$T^{-\frac{1}{2}}$	-	$T^{\frac{1}{2}}n^{-\frac{1}{2}}$
MoDo (Ours, Thms 3.1,3.3,3.5)	✗	$T^{-\frac{1}{4}}$	$T^{-\frac{1}{4}}$	$T^{\frac{1}{2}}n^{-\frac{1}{2}}$