Table 1: Comparison of GRL applied to fundament network (FADA) and HADA

UDA	A→C	A→P	A→R	Avg	MSDA	Clipart	Infograph	Painting	Avg	SSDA(1_{shot})	R→C	R→P	P→C	Avg
FADA	55.3	71.3	77.5	68.5	FADA	61.5	23.5	53.9	45.7	FADA	69.5	66.7	70.0	67.6
HADA	**56.8**	**75.2**	**79.8**	**70.9**	HADA	**63.9**	**25.9**	**56.1**	**47.9**	HADA	**71.7**	**67.1**	**72.8**	**69.5**

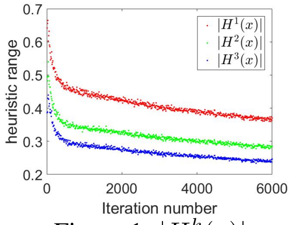

Figure 1: $|H^k(x)|$

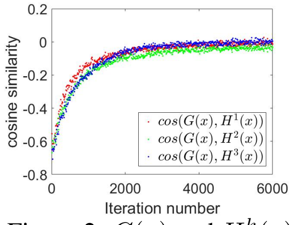

Figure 2: $G(x)$ and $H^k(x)$

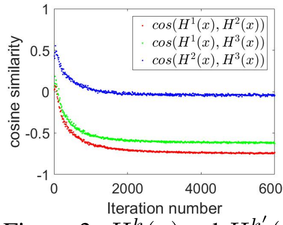

Figure 3: $H^k(x)$and $H^{k'}(x)$

Table 2: M in MSDA

HADA	Clipart	Infograph
$M = 1$	63.0	25.5
$M = 2$	63.7	25.7
$M = 3$	63.9	25.9
$M = 4$	64.0	26.0
$M = 5$	64.0	26.3
$M = 6$	64.1	26.3
$M = 7$	64.1	26.2

Reviewer1. 1. Network structure. We apply gradient reversal layer (GRL) to the single fundament network (FADA). Some results of FADA could be found in Table 1, which performs worse than HADA by a large margin. Besides, we also try to concatenate $F(x)$ and $H(x)$ as the input of a one-layer neural network, and applying GRL to $F(x)$, but the results are poor. For example, on A→C on Office-Home, its accuracy is 30.1 compared to 56.8 by HADA.

2. Sub-network. Each H^k is initialized with random weights generated from Gaussian with different variance, resulting in different output ranges in the training process, as shown in Figure 1. It could also ensure distinguished local domain-specific properties captured by different H^k, as shown in Figure 2 and 3 . $H^k(x)$ can effectively capture rich domain-specific properties, evidenced by the low similarity between $H^k(x)$ and $G(x)$, *i.e.*, $\cos(G(x), H^k(x)) \approx 0$, as shown in Figure 2 . Moreover, $H^k(x)$ contains different local property, since $H^k(x)$ and $H^{k'}(x)$ $(k \neq k')$ is orthogonal or negatively correlated, *i.e.*, $\cos(H^k(x), H^{k'}(x)) \leq 0$, as shown in Figure 3. Therefore, the aggregation of $H^k(x)$ could express more diversified domain specific properties encoded by each sub-network.

3. Experiments. The experiment settings and hyper-parameters follow [30][33]. Dataset separation on Office-Home follows [33], and the separation on Domainnet follows [30]. In tasks of UDA, SSDA and MSDA, the total number of iteration is fixed. Though the model might reach higher results during training, we still record the final performance when reaching the highest number of iteration for fair comparison. So we emphasize that early stopping is not used.

4. Motivation. The tasks of heuristic search and HADA are quite different, but heuristic search inspires us to devise both the heuristic network and termination constraint. To clarify the relationship, we will elaborate the connection between the two in the revised paper and supplementary.

Reviewer3. 1. Statistical significance. We report each result as the average accuracy on three trials. By calculating the std, we find that the performance gain by HADA is statistically significant. In Table 3, HADA achieves 47.9±0.40, which outperforms comparisons by large margins.

2. Number of M. Results of different M for MSDA are shown in Table 2. The results reach highest when $M = 6$, and it seems the performance does not degrade with a larger M. So task such as MSDA contains more domain properties, and tends to need a larger M. But $M = 3$ could still guarantee well-performed results on arbitrary number of domains.

3. Limitation. Separating domain-specific and domain-invariant parts without extra information could hardly achieve perfect separation in real situations. It is possible to introduce some extra domain-specific knowledge to enhance the separation. Also the structure of heuristic network could be enhanced by Neural Architecture Search (NAS).

Reviewer4. 1. Bound. The heuristic function could estimate and explicitly reduce the remaining domain-specific parts. The explicit reduction could result in more effective reduction of domain properties, with lower error bound. The bound is analyzed in many recent DA methods such as [33][56]. Lower error bound provides better theoretical guarantee for better model, as validated by experiments.

2. Ablation study. Our main contributions are heuristic sub-networks, similarity initialization, and termination loss. We tried HADA without each function in Table 2 in the paper. The performance without termination is poor, showing that the termination loss is more important compared with similar initialization and multiple sub-networks. The classification and adversarial loss constitute the basic framework of domain adaptation, which should not be removed.

3. Separation. Most existing separation methods are designed by statistic analysis, which seems to be less effective in real situations. DSN [6] separates the representations by input image reconstruction, but the improvement over DANN [17] is little (from 90.3 to 91.2 on Synth Digits to SVHN). We achieve significant improvement over DANN (from 57.6 to 70.9 on Office-Home).