Supplementary File of "Optimal Transport-Based Distributional Pairing in Transfer Multiobjective Optimization"

S-I. Additional Experiments

A. Scalability on Higher Objective Space Dimensions

TABLE S-I
IGD Results of NSGA-III, NSGA-III-EAS and NSGA-III-OTES averaged over 20 Independent Runs. The Kruskal-Wallis Test at a 0.05 Significance Level Was Performed to Compare NSGA-III-OTES, NSGA-III, and NSGA-III-EAS

Target Tasks	Iterations	NSGA-III		NSGA-III-EAS		NSGA-III-OTES
		Average IGD ± Std		Average IGD ± Std		Average IGD ± Std
mDTLZ1	20	0.1004±0.0099	+	0.0795±0.0058	+	**0.0672±0.0058**
	30	0.0720±0.0072		0.0618±0.0068		**0.0524±0.0062**
	40	0.0569±0.0071		0.0508±0.0065		**0.0428±0.0064**
	50	0.0462±0.0057		0.0429±0.0058		**0.0365±0.0049**
mDTLZ2	20	0.1697±0.0088	+	0.1858±0.0081	+	**0.1545±0.0066**
	30	0.1254±0.0081		0.1414±0.0069		**0.1150±0.0054**
	40	0.1007±0.0066		0.1153±0.0064		**0.0937±0.0055**
	50	0.0850±0.0049		0.0975±0.0049		**0.0806±0.0046**
mDTLZ3	20	0.2005±0.0120	+	0.2158±0.0121	+	**0.1698±0.0106**
	30	0.1427±0.0097		0.1585±0.0117		**0.1291±0.0110**
	40	0.1114±0.0060		0.1240±0.0120		**0.1020±0.0058**
	50	0.0918±0.0046		0.1039±0.0112		**0.0876±0.0048**
mDTLZ4	20	0.1677±0.0079	+	0.1776±0.0127	+	**0.1298±0.0091**
	30	0.1215±0.0068		0.1311±0.0087		**0.1022±0.0073**
	40	0.0939±0.0057		0.1045±0.0081		**0.0841±0.0050**
	50	0.0786±0.0040		0.0886±0.0076		**0.0732±0.0044**
$+/-/=$		4/0/0		4/0/0		

Here, we demonstrate another advantage of the proposed OTES: its effectiveness in transfer optimization for higher-dimensional objective spaces. To illustrate this, we employ mDTLZ1-mDTLZ4 problems with five objectives as examples and use the widely recognized NSGA-III [1] paradigm as the backbone optimizer for handling such many-objective problems. The utilized source data is generated by solving medium-correlation source tasks with five objectives. Table S-I presents the average IGD values achieved after 20, 30, 40, and 50 iterations by NSGA-III, NSGA-III-EAS, and NSGA-III-OTES, where NSGA-III-EAS and NSGA-III-OTES extend NSGA-III with evolutionary autoencoding search and OTES as the transfer techniques, respectively. The results indicate that NSGA-III-OTES provides the best convergence across all four benchmark problems. The table also includes the results of the Kruskal-Wallis test at a 0.05 significance level to confirm the statistical significance of the outcomes, where $+$", $-$", and "$\approx$" indicate that NSGA-III-OTES performs better than, worse than, or similarly to a competitor, respectively. According to the results, NSGA-III-OTES significantly outperforms NSGA-III and NSGA-III-EAS across all four target tasks. These findings validate the effectiveness of the proposed OTES in extending to higher-dimensional objective spaces.

B. Combination with the Mixture Model-Based Transfer

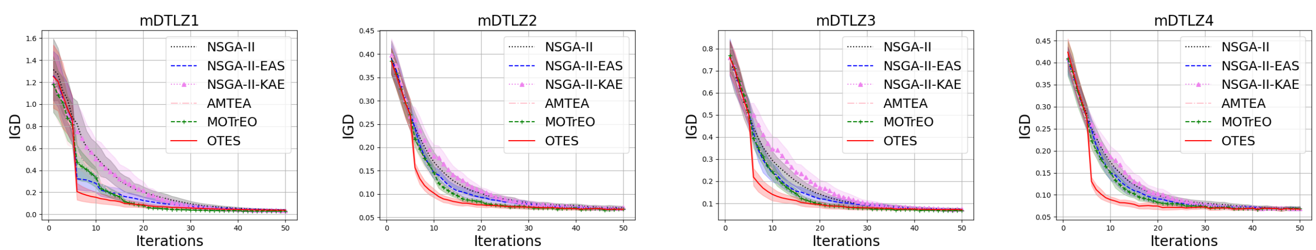
It is important to note that OTES primarily relies on optimal transport-based solution representation alignment to facilitate transfer optimization. Existing research has shown that solution representation alignment can be effectively combined with model-based transfer techniques to enhance algorithm performance. A notable example is MOTrEO, which integrates solution representation alignment based on objective function value sorting with a Gaussian mixture model-based transfer approach, achieving excellent optimization capabilities. An interesting question is whether OTES, as a solution representation alignment method, can also be combined with model-based transfer approaches to further improve optimization results. To investigate this, this subsection introduces a derived version of OTES, called OTES-MOTrEO, where the original solution representation alignment module in MOTrEO is replaced by OTES. As a comparison, we also compare its performance with MOTrEO and the original OTES. The averaged IGD values reached after 20, 30, 40, and 50 iterations are reported in Table S-II.

Table S-II shows that OTES-MOTrEO achieves faster convergence than both MOTrEO and OTES on all the benchmark problems. The table also presents the results of the Kruskal-Wallis test at a 0.05 significance level to confirm the statistical significance of the outcomes, where $+$", $-$", and "$\approx$" indicate that OTES-MOTrEO performs better than, worse than, or similarly to a competitor, respectively. Notably, OTES-MOTrEO significantly outperforms MOTrEO and OTES across all eight target MOPs. These results demonstrate that embedding OTES into model-based transfer techniques can further enhance the convergence performance compared to traditional solution representation alignment based optimizers and standalone OTES.

TABLE S-II
IGD RESULTS AND HV RESULTS OF MOTREO, OTES, AND OTES-MOTREO AVERAGED OVER 20 INDEPENDENT RUNS. THE KRUSKAL-WALLIS TEST AT A 0.05 SIGNIFICANCE LEVEL WAS PERFORMED TO COMPARE OTES-MOTREO WITH MOTREO AND OTES

Target Tasks	Iterations	MOTrEO		OTES		OTES-MOTrEO	MOTrEO		OTES		OTES-MOTrEO
		Average IGD ± Std		Average IGD ± Std		Average IGD ± Std	Average HV ± Std		Average HV ± Std		Average HV ± Std
mDTLZ1	20	0.0748±0.0137	+	0.0725±0.0192	+	**0.0537±0.0104**	1.2724±0.0090	+	1.2797±0.0082	+	**1.2864±0.0059**
	30	0.0397±0.0030		0.0504±0.0120		**0.0356±0.0022**	1.2939±0.0011		1.2898±0.0044		**1.2950±0.0020**
	40	0.0329±0.0014		0.0408±0.0075		**0.0299±0.0017**	1.2969±0.0008		1.2938±0.0024		**1.2978±0.0013**
	50	0.0299±0.0014		0.0360±0.0053		**0.0283±0.0016**	1.2981±0.0007		1.2960±0.0015		**1.2988±0.0008**
mDTLZ2	20	0.0826±0.0038	+	0.0733±0.0048	+	**0.0727±0.0033**	0.6431±0.0155	+	0.6595±0.0146	+	**0.6776±0.0101**
	30	0.0724±0.0028		0.0700±0.0038		**0.0674±0.0028**	0.6771±0.0111		0.6765±0.0113		**0.6937±0.0066**
	40	0.0680±0.0026		0.0692±0.0025		**0.0667±0.0022**	0.6866±0.0070		0.6852±0.0087		**0.6991±0.0062**
	50	0.0668±0.0019		0.0674±0.0031		**0.0660±0.0024**	0.6950±0.0051		0.6940±0.0099		**0.7031±0.0082**
mDTLZ3	20	0.0982±0.0073	+	0.0861±0.0094	+	**0.0797±0.0049**	0.6081±0.0166	+	0.6257±0.0250	+	**0.6575±0.0173**
	30	0.0774±0.0030		0.0752±0.0063		**0.0713±0.0041**	0.6578±0.0117		0.6561±0.0217		**0.6761±0.0136**
	40	0.0712±0.0030		0.0716±0.0033		**0.0681±0.0035**	0.6770±0.0098		0.6761±0.0145		**0.6883±0.0096**
	50	0.0691±0.0024		0.0699±0.0042		**0.0675±0.0031**	0.6865±0.0070		0.6817±0.0096		**0.6969±0.0070**
mDTLZ4	20	0.0826±0.0044	+	0.0752±0.0036	+	**0.0725±0.0046**	0.6460±0.0147	+	0.6633±0.0145	+	**0.6753±0.0076**
	30	0.0732±0.0023		0.0695±0.0033		**0.0683±0.0030**	0.6739±0.0090		0.6772±0.0117		**0.6930±0.0096**
	40	0.0688±0.0024		0.0692±0.0031		**0.0668±0.0034**	0.6864±0.0090		0.6889±0.0100		**0.6970±0.0069**
	50	0.0681±0.0026		0.0661±0.0024		**0.0666±0.0030**	0.6927±0.0070		0.6934±0.0118		**0.7033±0.0065**
$mDTLZ1^{-1}$	20	0.0473±0.0087	+	0.0662±0.0268	+	**0.0429±0.0094**	0.0478±0.0039	+	0.0458±0.0052	+	**0.0521±0.0016**
	30	0.0301±0.0023		0.0442±0.0157		**0.0288±0.0017**	0.0571±0.0009		0.0533±0.0035		**0.0580±0.0011**
	40	0.0269±0.0010		0.0336±0.0092		**0.0261±0.0011**	0.0591±0.0006		0.0571±0.0022		**0.0597±0.0010**
	50	0.0258±0.0008		0.0296±0.0054		**0.0255±0.0012**	0.0600±0.0005		0.0590±0.0015		**0.0609±0.0005**
$mDTLZ2^{-1}$	20	0.0714±0.0040	+	0.0673±0.0039	+	**0.0658±0.0037**	0.6564±0.0048	+	0.6842±0.0046	−	**0.6863±0.0041**
	30	0.0667±0.0030		0.0646±0.0022		**0.0644±0.0025**	0.6737±0.0043		**0.6890±0.0047**		0.6885±0.0048
	40	0.0648±0.0018		0.0648±0.0023		**0.0634±0.0029**	0.6806±0.0034		**0.6922±0.0045**		0.6913±0.0034
	50	**0.0639±0.0019**		0.0649±0.0017		**0.0639±0.0018**	0.6841±0.0033		**0.6936±0.0044**		0.6924±0.0031
$mDTLZ3^{-1}$	20	0.0745±0.0039	+	0.0768±0.0080	+	**0.0700±0.0037**	0.6470±0.0075	+	0.6569±0.0151	+	**0.6704±0.0115**
	30	0.0680±0.0023		0.0704±0.0056		**0.0658±0.0019**	0.6674±0.0058		0.6735±0.0102		**0.6797±0.0074**
	40	0.0662±0.0020		0.0669±0.0039		**0.0659±0.0030**	0.6758±0.0045		0.6806±0.0079		**0.6836±0.0043**
	50	0.0654±0.0022		0.0656±0.0033		**0.0648±0.0023**	0.6789±0.0043		0.6857±0.0038		**0.6870±0.0042**
$mDTLZ4^{-1}$	20	0.0702±0.0031	+	0.0676±0.0061	+	**0.0644±0.0025**	0.6566±0.0057	+	0.6808±0.0076	+	**0.6857±0.0052**
	30	0.0663±0.0025		0.0642±0.0025		**0.0637±0.0030**	0.6741±0.0040		0.6879±0.0057		**0.6892±0.0042**
	40	0.0643±0.0019		0.0657±0.0024		**0.0636±0.0023**	0.6803±0.0041		0.6899±0.0044		**0.6925±0.0049**
	50	0.0640±0.0014		0.0641±0.0027		**0.0635±0.0016**	0.6846±0.0026		0.6931±0.0037		**0.6936±0.0029**
+/−/=		8/0/0		8/0/0			8/0/0		7/0/1		

C. Scalability Across Different Levels of Source-Target Correlation



(a) (b) (c) (d)

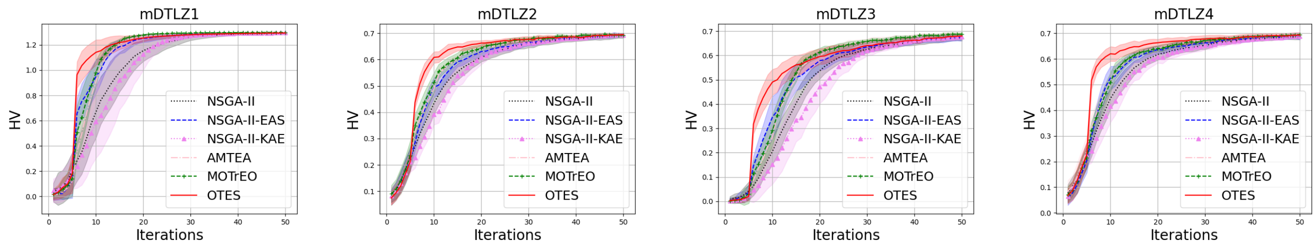
Fig. S-1. Comparison of IGD convergence trends average over 20 independent runs of NSGA-II, NSGA-II-EAS, NSGA-II-KAE, ATMEA, MOTrEO and OTES. Here, the low-correlation source data is utilized to assist all of the transfer optimization algorithms. Details on the generation of such low-correlation source data can be found in [2]. Shaded areas represent one standard deviation on either side of the mean. (a) mDTLZ1. (b) mDTLZ2. (c) mDTLZ3. (d) mDTLZ4.

In the previous experiments, we primarily evaluated the performance of our method using medium-correlation source data. In this section, we further assess the effectiveness of OTES by utilizing the low-correlation source dataset introduced in [3], demonstrating OTES's ability to adapt to varying levels of source-target correlations. Specifically, we set $\delta_1 = 0.3$ and $\delta_2 = 0.4$ to construct a low-correlation source task, and subsequently generate $\mathscr{P}_{s,k}$ by solving this task. More detail information can be found in [3]. We use mDTLZ1-mDTLZ4 as examples and solve them using NSGA-II, NSGA-II-EAS, NSGA-II-KAE, ATMEA, MOTrEO, and OTES, respectively. The IGD and HV convergence trends of six two algorithms over 20 independent runs are shown in Fig. S-1 and Fig. S-2, respectively. The results shown in Fig. S-1 and Fig. S-2 indicate that, even with low-correlation source data, OTES achieves faster convergence than its competitors. This demonstrates that OTES can effectively handle different levels of source-target correlations and mitigate the occurrence of negative transfer.

D. Influence of Hyperparameters

In OTES, two hyperparameters are introduced: the transfer frequency k_{tr} and the regularization coefficient λ. In this subsection, we further examine the impact of these two hyperparameters by utilizing the mDTLZ benchmark problems.

1) Influence of k_{tr}: To investigate the influence of k_{tr}, we set it to four different values, i.e., 2, 5, 10, and 20. OTES is executed under each setting, and the performances of the algorithm are compared accordingly. The average IGD values achieved after 10, 20, 30, 40, and 50 iterations are reported in Table S-III. From the results, it can be observed that the setting $k_{tr} = 5$ yields the best performance. Therefore, this value is adopted in our experiments.



(a) (b) (c) (d)

Fig. S-2. Comparison of HV convergence trends average over 20 independent runs of NSGA-II, NSGA-II-EAS, NSGA-II-KAE, ATMEA, MOTrEO and OTES. Here, the low-correlation source data is utilized to assist all of the transfer optimization algorithms. Details on the generation of such low-correlation source data can be found in [2]. Shaded areas represent one standard deviation on either side of the mean. (a) mDTLZ1. (b) mDTLZ2. (c) mDTLZ3. (d) mDTLZ4.

TABLE S-III
IGD Results of OTES with $k_{tr}=2$, $k_{tr}=5$, $k_{tr}=10$, and $k_{tr}=20$ averaged over 20 Independent Runs.

Target Tasks	Iterations	$k_{tr}=2$	$k_{tr}=5$	$k_{tr}=10$	$k_{tr}=20$
		Average IGD ± Std	Average IGD ± Std	Average IGD ± Std	Average IGD ± Std
mDTLZ1	10	**0.1032**±0.0284	0.1161±0.0264	0.4634±0.0802	0.4863±0.1038
	20	0.0714±0.0234	**0.0696**±0.0167	0.0893±0.0347	0.2210±0.0516
	30	0.0564±0.0180	**0.0499**±0.0105	0.0556±0.0112	0.0707±0.0197
	40	0.0471±0.0146	**0.0389**±0.0063	0.0415±0.0061	0.0516±0.0079
	50	0.0413±0.0106	**0.0346**±0.0037	0.0365±0.0037	0.0372±0.0048
mDTLZ2	10	**0.0899**±0.0067	0.0927±0.0091	0.1680±0.0146	0.1652±0.0164
	20	0.0773±0.0048	**0.0754**±0.0062	0.0770±0.0040	0.0999±0.0088
	30	0.0732±0.0044	**0.0714**±0.0047	0.0713±0.0037	0.0724±0.0034
	40	0.0697±0.0031	**0.0695**±0.0038	0.0688±0.0037	0.0705±0.0030
	50	0.0703±0.0043	**0.0676**±0.0041	0.0686±0.0040	0.0684±0.0033
mDTLZ3	10	0.1186±0.0183	**0.1168**±0.0159	0.3199±0.0322	0.2928±0.0498
	20	0.0918±0.0109	**0.0859**±0.0075	0.0964±0.0132	0.1381±0.0165
	30	0.0828±0.0072	**0.0764**±0.0055	0.0805±0.0075	0.0777±0.0039
	40	0.0759±0.0046	**0.0728**±0.0044	0.0739±0.0038	0.0752±0.0040
	50	0.0739±0.0048	**0.0696**±0.0034	0.0700±0.0034	0.0701±0.0024
mDTLZ4	10	0.0970±0.0126	**0.0866**±0.0083	0.1788±0.0169	0.1745±0.0175
	20	0.0773±0.0064	**0.0746**±0.0052	0.0773±0.0054	0.0980±0.0083
	30	0.0730±0.0059	**0.0688**±0.0030	0.0721±0.0038	0.0710±0.0028
	40	0.0715±0.0051	**0.0670**±0.0029	0.0676±0.0030	0.0701±0.0034
	50	0.0693±0.0031	**0.0680**±0.0031	0.0680±0.0026	0.0670±0.0031

2) Influence of λ: We further examine the influence of λ. Specifically, λ is set to three different values: 0.1, 0.5, and 1.0. For each setting, OTES is executed 20 times, and the performance of the algorithm is recorded. The average IGD values obtained after 10, 20, 30, 40, and 50 iterations are presented in Table S-IV. According to the results, the setting $\lambda = 0.1$ yields the best performance.

E. Time Consumption of Sinkhorn-based Optimal Transport

As an explicit transfer technique, OTES requires establishing a coupling between the solutions in the source and target populations prior to learning the mapping between their respective search spaces. This coupling is computed using the Sinkhorn algorithm. However, employing the Sinkhorn algorithm introduces additional computational overhead, raising the question of whether this overhead significantly impacts the algorithm's overall efficiency. To assess this, we evaluate the time consumption of OTES in this subsection. Specifically, we compare the runtime of OTES with that of NSGA-II-EAS on 10-dimensional problems, as shown in Table S-V. We select NSGA-II-EAS as the baseline because the only major difference between the two methods is the inclusion of the Sinkhorn process in OTES, allowing for a direct assessment of its time cost. The results indicate that the Sinkhorn algorithm introduces averagely 1.8610 seconds of additional computation time, which is relatively minor. Therefore, the overhead introduced by the Sinkhorn algorithm is acceptable and does not substantially affect the overall runtime of OTES.

In addition, since the Sinkhorn algorithm tends to incur higher computational complexity when dealing with high-dimensional decision spaces, we further examine the time consumption of OTES under such conditions. Specifically, we compare the performance of NSGA-II-EAS and OTES on 50-dimensional problems, with the number of iterations fixed at 200. The corresponding runtime results are presented in Table S-V. As observed, although OTES requires more computational time than NSGA-II-EAS, the additional overhead accounts for only approximately 51.61% of the runtime of NSGA-II-EAS. This

TABLE S-IV
IGD RESULTS OF OTES WITH $\lambda = 0.1$, $\lambda = 0.5$, AND $\lambda = 1.0$ AVERAGED OVER 20 INDEPENDENT RUNS.

Target Tasks	Iterations	$\lambda = 0.1$	$\lambda = 0.5$	$\lambda = 1.0$
		Average IGD ± Std	Average IGD ± Std	Average IGD ± Std
mDTLZ1	10	**0.1114**±0.0325	0.1119±0.0306	0.1416 ±0.0515
	20	**0.0667**±0.0202	0.0732±0.0202	0.0829 ±0.0302
	30	**0.0484**±0.0145	0.0511±0.0108	0.0553 ±0.0141
	40	**0.0388**±0.0087	0.0402±0.0066	0.0441 ±0.0070
	50	**0.0344**±0.0064	0.0349±0.0033	0.0373 ±0.0042
mDTLZ2	10	**0.0889**±0.0072	0.1014±0.0102	0.1200 ±0.0129
	20	**0.0745**±0.0037	0.0795±0.0038	0.0814 ±0.0082
	30	**0.0715**±0.0039	0.0733±0.0039	0.0763 ±0.0054
	40	**0.0679**±0.0027	0.0700±0.0028	0.0704 ±0.0027
	50	**0.0663**±0.0027	0.0686±0.0031	0.0691 ±0.0024
mDTLZ3	10	**0.1114**±0.0165	0.1324±0.0182	0.1629 ±0.0241
	20	**0.0857**±0.0104	0.0945±0.0103	0.1047 ±0.0111
	30	**0.0762**±0.0072	0.0798±0.0042	0.0864 ±0.0072
	40	0.0736±0.0064	**0.0722**±0.0032	0.0780 ±0.0048
	50	0.0715±0.0038	**0.0709**±0.0031	0.0738 ±0.0033
mDTLZ4	10	**0.0867**±0.0081	0.1071±0.0139	0.1257 ±0.0149
	20	**0.0754**±0.0053	0.0767±0.0060	0.0814 ±0.0047
	30	0.0712±0.0043	**0.0709**±0.0031	0.0735 ±0.0044
	40	0.0696±0.0041	**0.0682**±0.0023	0.0711 ±0.0036
	50	0.0682±0.0027	**0.0673**±0.0025	0.0679 ±0.0027

level of overhead remains within an acceptable range, indicating that OTES maintains practical efficiency even in higher-dimensional scenarios.

TABLE S-V
TIME CONSUMPTION OF NSGA-II-EAS AND OTES ON SOLVING MDTLZ AND MDTLZ^{-1} PROBLEMS.

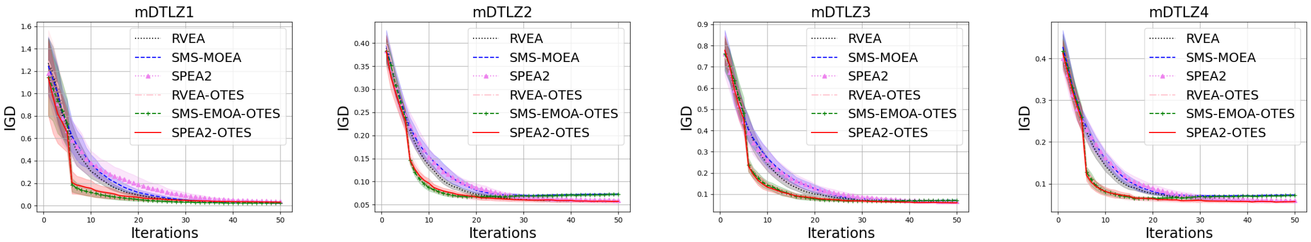
Target Tasks	$d = 10$		$d = 50$	
	Time Consumption (seconds)			
	NSGA-II-EAS	OTES	NSGA-II-EAS	OTES
mDTLZ1	4.4078	8.6711	47.384	72.8038
mDTLZ2	4.7845	5.6572	44.8108	68.3865
mDTLZ3	4.3890	5.7258	44.4185	64.1942
mDTLZ4	4.3967	6.1684	44.9926	74.9013
mDTLZ5	4.2569	6.1855	44.5908	62.9681
mDTLZ6	4.8625	6.1909	44.8490	68.4200
mDTLZ7	4.4579	6.1359	44.3652	66.5990
mDTLZ8	4.4483	6.1565	45.2015	68.4346
Average	4.5004	6.3614	45.0766	68.3384

F. Compatibility with Various Backbone Optimizers

In Section I, we highlighted that the proposed OTES framework is compatible with mixture model-based transfer techniques. In this subsection, we further demonstrate that OTES is also compatible with various backbone optimizers. To validate this, we integrate OTES with three representative optimization algorithms: RVEA, SMS-MOEA, and SPEA2. The resulting variants, referred to as RVEA-OTES, SMS-MOEA-OTES, and SPEA2-OTES, are evaluated on the mDTLZ benchmark problems. The convergence trajectories of all six algorithms are presented in Fig. S-3. The results show that, across different backbone optimizers, the incorporation of OTES consistently improves convergence performance. These findings confirm the versatility and compatibility of the proposed OTES framework with a variety of optimization paradigms.

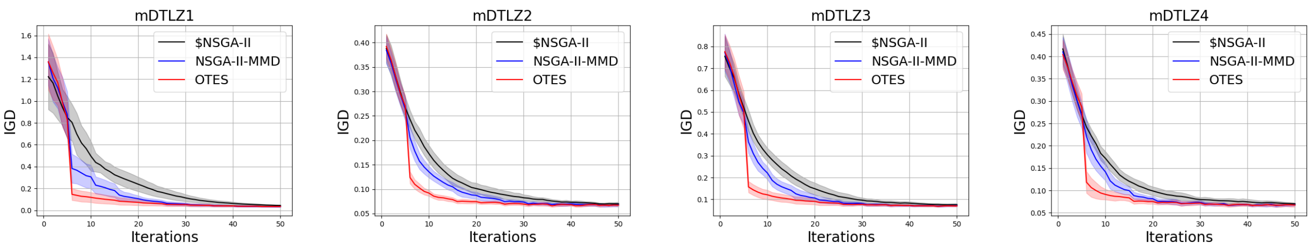
G. Compare with MMD based Alignment

In previous work, we primarily compared OTES with solution representation alignment methods based on sorting-based pairing, and demonstrated the superior performance of OTES. As discussed in Section III, OTES adopts a distribution-based alignment strategy, leveraging optimal transport to estimate the Wasserstein distance between the source and target data distributions, and learning a mapping to align them accordingly. In Section II, we also introduced MMD-based approachessuch as transfer component analysiswhich similarly aim to achieve distribution alignment. This raises an important question: how does OTES compare to MMD-based alignment methods? To answer this, we include a comparison with an MMD-based method. Specifically, we implement MMD, where a kernel-based approach is used to compute the distance between empirical distributions. A linear mapping is trained using MMD as the loss function, and the optimization is performed via a quasi-Newton method. To ensure a fair comparison, we adopt NSGA-II as the backbone optimizer for both methods. The MMD-based



(a) (b) (c) (d)

Fig. S-3. Comparison of IGD convergence trends average over 20 independent runs of RVEA, SMS-MOEA, SPEA2, RVEA-OTES, SMS-MOEA-OTES, and SPEA2-OTES. Shaded areas represent one standard deviation on either side of the mean. (a) mDTLZ1. (b) mDTLZ2. (c) mDTLZ3. (d) mDTLZ4.



(a) (b) (c) (d)

Fig. S-4. Comparison of IGD convergence trends average over 20 independent runs of NSGA-II, NSGA-II-MMD, and OTES. Shaded areas represent one standard deviation on either side of the mean. (a) mDTLZ1. (b) mDTLZ2. (c) mDTLZ3. (d) mDTLZ4.

alignment method is embedded into NSGA-II, and the resulting algorithm is denoted as NSGA-II-MMD. For reference, we also include the original NSGA-II algorithm without any alignment technique as a baseline. The convergence trends of NSGA-II, NSGA-II-MMD, and OTES are shown in Fig. S-4. The results clearly demonstrate that OTES outperforms both the baseline and the MMD-based alignment method, further validating its effectiveness and superior capability in achieving efficient alignment for transfer optimization.

References

[1] K. Deb and H. Jain, "An evolutionary many-objective optimization algorithm using reference-point-based nondominated sorting approach, part i: solving problems with box constraints," *IEEE transactions on evolutionary computation*, vol. 18, no. 4, pp. 577–601, 2013.

[2] Y. Li and W. Gong, "Multiobjective multitask optimization with multiple knowledge types and transfer adaptation," *IEEE Transactions on Evolutionary Computation*, 2024.

[3] J. Liu, A. Gupta, and Y.-S. Ong, "Bayesian inverse transfer in evolutionary multiobjective optimization," *ACM Trans. Evol. Learn. Optim.*, jun 2024. Just Accepted.