Solving Few-Shot Multiobjective Multitask Optimization via Iterative Sequential Transfer

Tingyang Wei, Haofeng Wu, Ananda Phan Iman, Zhao Wei, Jiao Liu, Yew-Soon Ong

College of Computing and Data Science, Nanyang Technological University
Centre for Frontier AI Research (CFAR), A*STAR
Department of AI Convergence, Gwangju Inst. of Sci. & Tech. (GIST)

2026.06.22

IEEE WCCI 2026 Maastricht

Outline

- Background
  - Multitask Optimization
  - Multiobjective Multitask Optimization
  - Few-Shot Challenges
  - Solving Multitask Optimization via (Iterative) Sequential Transfer Optimization
- Instantiation 1: Iterative sequential transfer optimization can improve multitask optimization
- Instantiation 2: Iterative sequential transfer can improve sequential transfer optimizer in multitask optimization
- Instantiation 3: Improved multitask optimization in multitask hyperparameter optimization
- Conclusions and Potential Future Work

Multitask Optimization

- Multitask Optimization
  - $argmin_{x_m \in \mathcal{X}_m} f_m(x_m), m \in [M]$

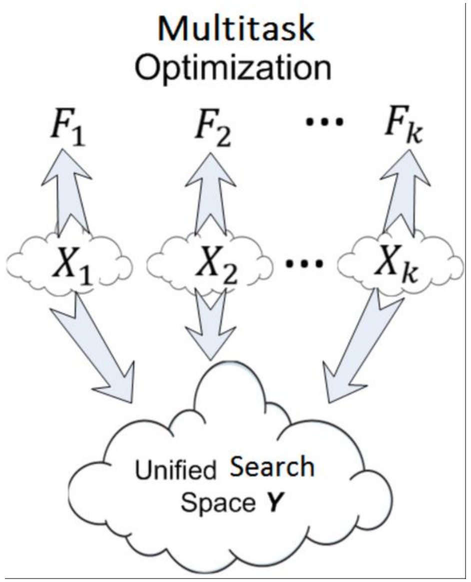
    Solve M optimization tasks simultaneously

  - Why? **Assuming** M problems share certain similarities, we use their **solution distributions and surrogate models** as transferable knowledge to accelerate convergence.

Multitask Optimization

- Multitask Optimization
  - $argmin_{x_m \in \mathcal{X}_m} f_m(x_m), m \in [M]$

    Solve M optimization tasks simultaneously
  - Implicit Transfer
    - Unified evolutionary process for multiple tasks into a single or multiple populations.
  - Explicit Transfer



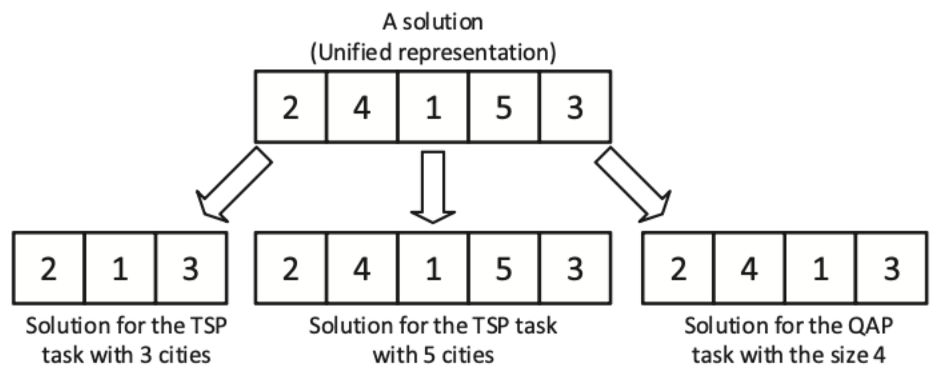


Fig. 1. The interpretation of unified representation for each task.

Ong, Yew-Soon, and Abhishek Gupta. "Evolutionary multitasking: A computer science view of cognitive multitasking." Cognitive Computation 8.2 (2016): 125-142.
Yuan, Yuan, et al. "Evolutionary multitasking in permutation-based combinatorial optimization problems: Realization with TSP, QAP, LOP, and JSP." *2016 IEEE Region 10 Conference (TENCON)*. IEEE, 2016.

Multitask Optimization

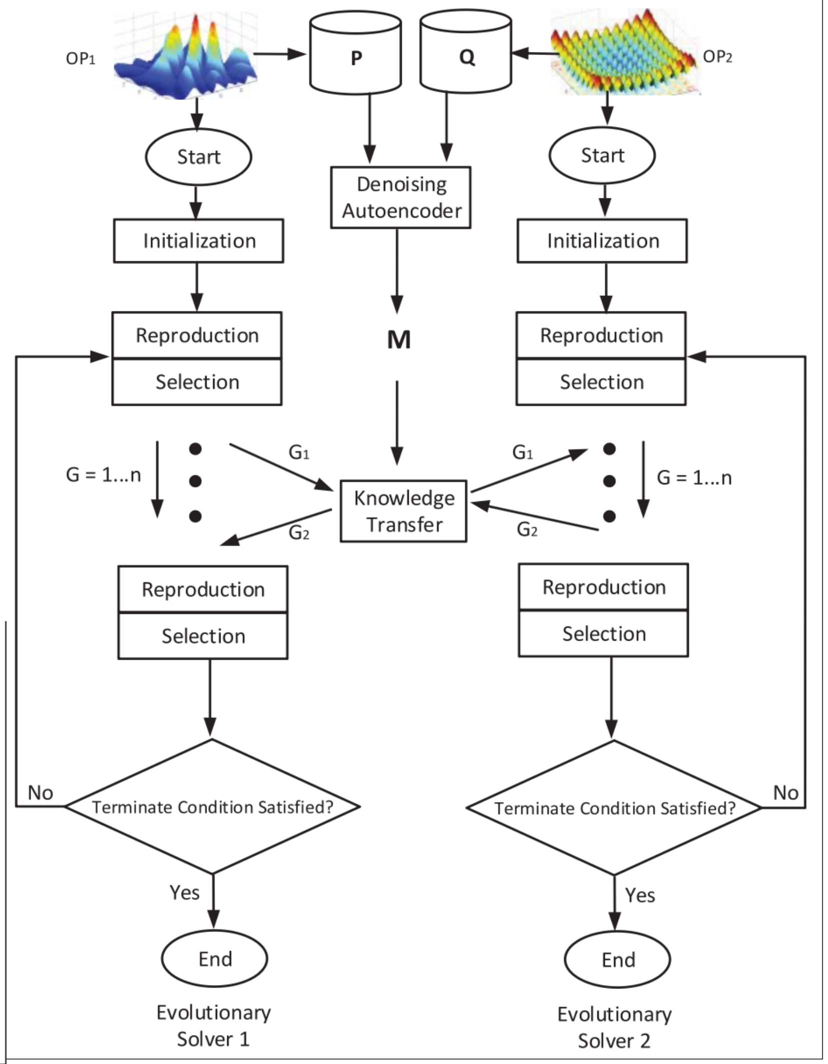
- Multitask Optimization
  - $argmin_{x_m \in \mathcal{X}_m} f_m(x_m), m \in [M]$

Solve M optimization tasks simultaneously

  - Implicit Transfer
  - Explicit Transfer
    - Map high-quality solutions from one task to another task.

CONCLUSION REMARK ON TASK TRANSFER STRATEGY

Strategy	Operation[a]	Complexity[b]	Reference
Distribution	$\hat{x} = x + \alpha[0.5 - mean(x_s)]$	$O(SNM^2)$	[46]
Distribution	$\hat{x} = x * mean(x_t)/mean(x_s)$	$O(SNM^2)$	[48]
Distribution	$\hat{x} = x - mean(x_s) + mean(x_t)$	$O(SNM^2)$	[49]
Distribution	$\hat{x} = x + \alpha[mean(x_{pro}) - x] + rand(dist(x, mean(x_s)) + dist(mean(x_s), mean(x_{pro}))$	$O(SNM^2)$	[50]
Matching	$\hat{x} = Mx$, M matches ranked individuals in x_s and x_t with least square method	$O(M^2N^2(N+S))$	[51], [52]
Matching	$\hat{x} = Mx$, M matches ranked individuals in x_s and x_t with neural networks	$O(M^2N^2(N+S))$	[53]
Matching	$\hat{x} = Mx$, M matches location information in $CVRP_s$ and $CVRP_t$ with $L1$-norm	$O(SNM^2T)$[c]	[54]
Hybrid	$\hat{x} = Ax + b, A = L_s L_t^{-1}, b = mean(x_t) - mean(x_s)A$	$O(SNM^2)$	[55]
Hybrid	$\hat{x} = MA_s x$, M matches the principle components A_s in x_s and A_t in x_t	$O(S^3M^2)$	[56], [57]
Hybrid	$\hat{x} = M_t^T M_s x$, M_i matches task i and latent common space	-	[58]



Feng, Liang, et al. "Evolutionary multitasking via explicit autoencoding." IEEE transactions on cybernetics 49.9 (2018): 3457-3470.
Wei, Tingyang, et al. "A review on evolutionary multitask optimization: Trends and challenges." IEEE Transactions on Evolutionary Computation 26.5 (2021): 941-960.

Multiobjective Multitask Optimization

- Multiobjective Multitask Optimization

  - $argmin_{x_m \in \mathcal{X}_m} F_m(x_m), m \in [M]$

    Solve M multiobjective optimization tasks simultaneously

  - Why? **Assuming** K multi-objective problems share certain similarities, we use their **Pareto-optimal solution distributions and surrogate models** as transferable knowledge to accelerate convergence.

Multiobjective Multitask Optimization

- Multiobjective Multitask Optimization
  - $argmin_{x_m \in \mathcal{X}_m} F_m(x_m), m \in [M]$

Solve M multiobjective optimization tasks simultaneously

  - Implicit Transfer
  - Explicit Transfer
    - Tricks are generally compatible with single-objective multitask optimization.

CONCLUSION REMARK ON TASK TRANSFER STRATEGY

Strategy	Operation[a]	Complexity[b]	Reference
Distribution	$\hat{x} = x + \alpha[0.5 - mean(x_s)]$	O(SNM^2)	[46]
Distribution	$\hat{x} = x * mean(x_t)/mean(x_s)$	O(SNM^2)	[48]
Distribution	$\hat{x} = x - mean(x_s) + mean(x_t)$	O(SNM^2)	[49]
Distribution	$\hat{x} = x + \alpha[mean(x_{pro}) - x] + rand(dist(x, mean(x_s)) + dist(mean(x_s), mean(x_{pro}))$	O(SNM^2)	[50]
Matching	$\hat{x} = Mx$, M matches ranked individuals in x_s and x_t with least square method	O($M^2N^2(N+S)$)	[51], [52]
Matching	$\hat{x} = Mx$, M matches ranked individuals in x_s and x_t with neural networks	O($M^2N^2(N+S)$)	[53]
Matching	$\hat{x} = Mx$, M matches location information in $CVRP_s$ and $CVRP_t$ with $L1$-norm	O(SNM^2T)[c]	[54]
Hybrid	$\hat{x} = Ax + b, A = L_s L_t^{-1}, b = mean(x_t) - mean(x_s)A$	O(SNM^2)	[55]
Hybrid	$\hat{x} = MA_s x$, M matches the principle components A_s in x_s and A_t in x_t	O(S^3M^2)	[56], [57]
Hybrid	$\hat{x} = M_t^T M_s x$, M_i matches task i and latent common space	-	[58]

Wei, Tingyang, et al. "A review on evolutionary multitask optimization: Trends and challenges." IEEE Transactions on Evolutionary Computation 26.5 (2021): 941-960.

Few-Shot Challenges in (Multiobjective) Multitask Optimization

- Few-Shot Challenges
  - To conduct (multiobjective) optimization **within limited evaluation budgets**.
  - $argmin_{x \in \mathcal{X}} \, f(x)$, $\boldsymbol{C_{max} = C}$
  - $argmin_{x \in \mathcal{X}} \, F(x) = argmin_{x \in \mathcal{X}} \left(f_1(x), \ldots, f_M(x)\right)$, $\boldsymbol{C_{max} = C}$

"Few-Shot" means budget $\boldsymbol{C}$ can be really small.

Why "Few Shot"?

1. Real problems entail costly function calls.
2. Problems of other forms (multi-objective) entails even more function calls.

Few-Shot Challenges in (Multiobjective) Multitask Optimization

- Few-Shot Challenges
  - To conduct (multiobjective) optimization **within limited evaluation budgets**.
  - $argmin_{x \in \mathcal{X}} f(x)$, $\boldsymbol{C_{max} = C}$
  - $argmin_{x \in \mathcal{X}} F(x) = argmin_{x \in \mathcal{X}}\left(f_1(x), \ldots, f_M(x)\right)$, $\boldsymbol{C_{max} = C}$

"Few-Shot" means budget $\boldsymbol{C}$ can be really small.

  - Limited evaluation budgets restrict the data/model-driven transfer strategies.

Matching	$\hat{x} = Mx$, M matches ranked individuals in x_s and x_t with least square method
Matching	$\hat{x} = Mx$, M matches ranked individuals in x_s and x_t with neural networks
Matching	$\hat{x} = Mx$, M matches location information in $CVRP_s$ and $CVRP_t$ with $L1$-norm
Hybrid	$\hat{x} = Ax + b$, $A = L_s L_t^{-1}$, $b = mean(x_t) - mean(x_s)A$
Hybrid	$\hat{x} = MA_s x$, M matches the principle components A_s in x_s and A_t in x_t
Hybrid	$\hat{x} = M_t^T M_s x$, M_i matches task i and latent common space

Limited samples cannot well capture the relationship between search spaces.

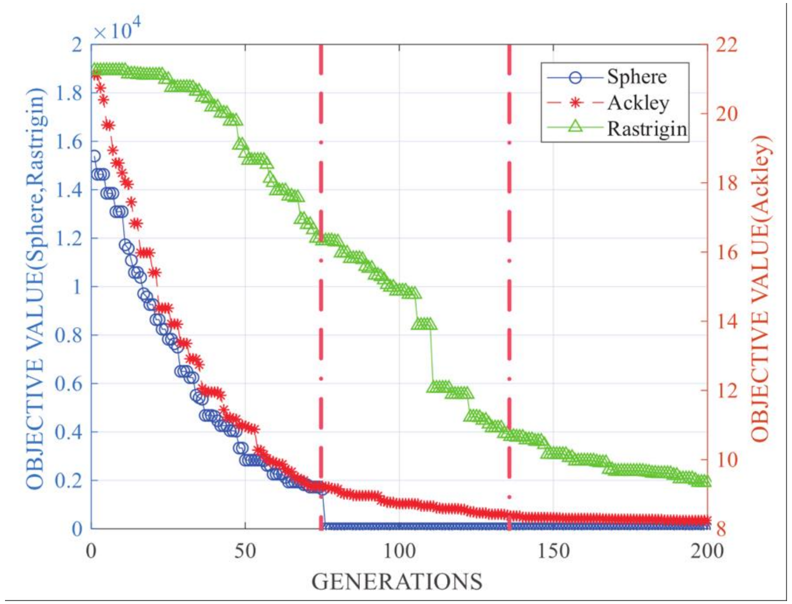
Wei, Tingyang, et al. "A review on evolutionary multitask optimization: Trends and challenges." IEEE Transactions on Evolutionary Computation 26.5 (2021): 941-960.

Few-Shot Challenges in (Multiobjective) Multitask Optimization

- Few-Shot Challenges
  - To conduct (multiobjective) optimization **within limited evaluation budgets**.
  - $argmin_{x \in \mathcal{X}}\ f(x)$, $\boldsymbol{C_{max} = C}$
  - $argmin_{x \in \mathcal{X}}\ F(x) = argmin_{x \in \mathcal{X}}\left(f_1(x), \dots, f_M(x)\right)$, $\boldsymbol{C_{max} = C}$

"Few-Shot" means budget $\boldsymbol{C}$ can be really small.

  - Stagnated optimization tasks cannot serve as informative source tasks to those target tasks.



The **blue** curve is stagnated and should not serve as source tasks to others.

Gong, Maoguo, et al. "Evolutionary multitasking with dynamic resource allocating strategy." IEEE Transactions on Evolutionary Computation 23.5 (2019): 858-869.
Wei, Tingyang, and Jinghui Zhong. "Towards generalized resource allocation on evolutionary multitasking for multi-objective optimization." IEEE Computational Intelligence Magazine 16.4 (2021): 20-37.

Our Solution: Solve Multitask Optimization via Sequential Transfer (STrO)

- Sequential Transfer (STrO)
  - To conduct optimization on the target tasks via information gathered in source tasks.
  - $min_{x \in \mathcal{X}} f_M(x), given \{\mathcal{D}_{S_m}\}_{m=1}^{M-1}$

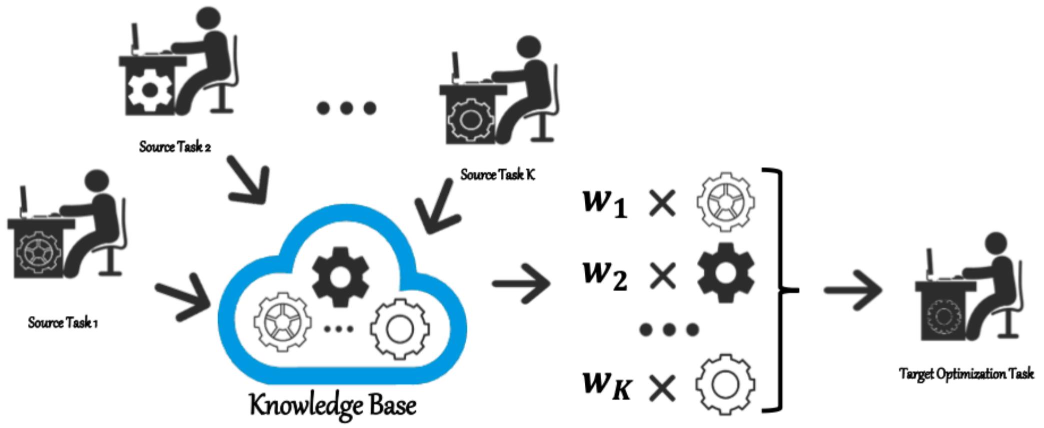


Fig. 1: In sequential transfer optimization, knowledge from pre-optimized tasks is extracted and stored in a knowledge base. This accumulated knowledge is later utilized to solve the target task.

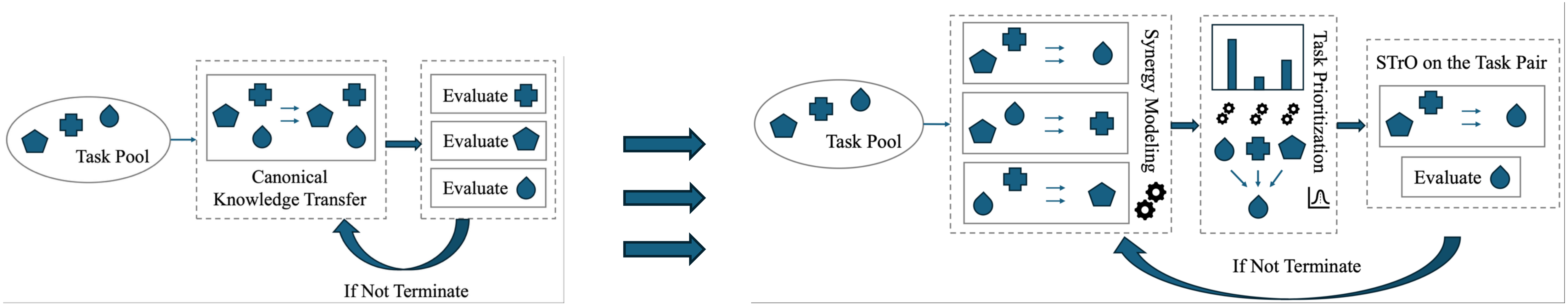
Multitask optimization can be modeled as a sequence of sequential transfer optimization!

Hou, Yaqing, et al. "Bridging the gap between theory and practice: Benchmarking transfer evolutionary optimization." arXiv preprint arXiv:2404.13377 (2024).

Solve Multitask Optimization via Iterative STrO

$$min_{x \in \mathcal{X}} f_M(x), given \{\mathcal{D}_{S_m}\}_{m=1}^{M-1}$$

- Multitask Optimization via Iterative STrO



Solve multiple optimization tasks evenly.

- Select one prioritized task
- Optimize this target task via other sources **(STrO)**

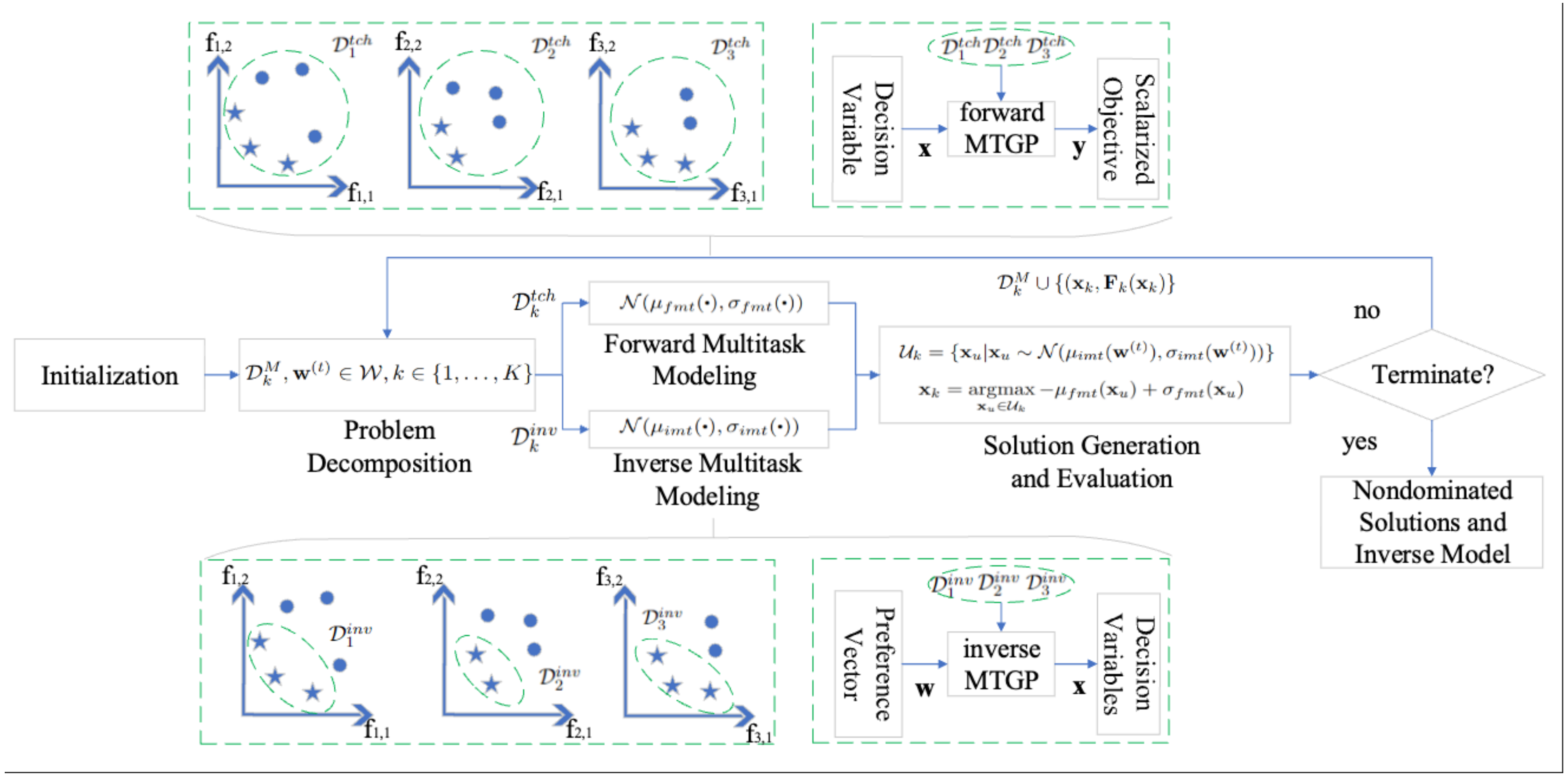
How to select the prioritized task?

Outline

- Background
  - Multitask Optimization
  - Multiobjective Multitask Optimization
  - Few-Shot Challenges
  - Solving Multitask Optimization via (Iterative) Sequential Transfer Optimization
- Instantiation 1: Iterative sequential transfer optimization can improve multitask optimization
- Instantiation 2: Iterative sequential transfer can improve sequential transfer optimizer in multitask optimization
- Instantiation 3: Improved multitask optimization in multitask hyperparameter optimization
- Conclusions and Potential Future Work

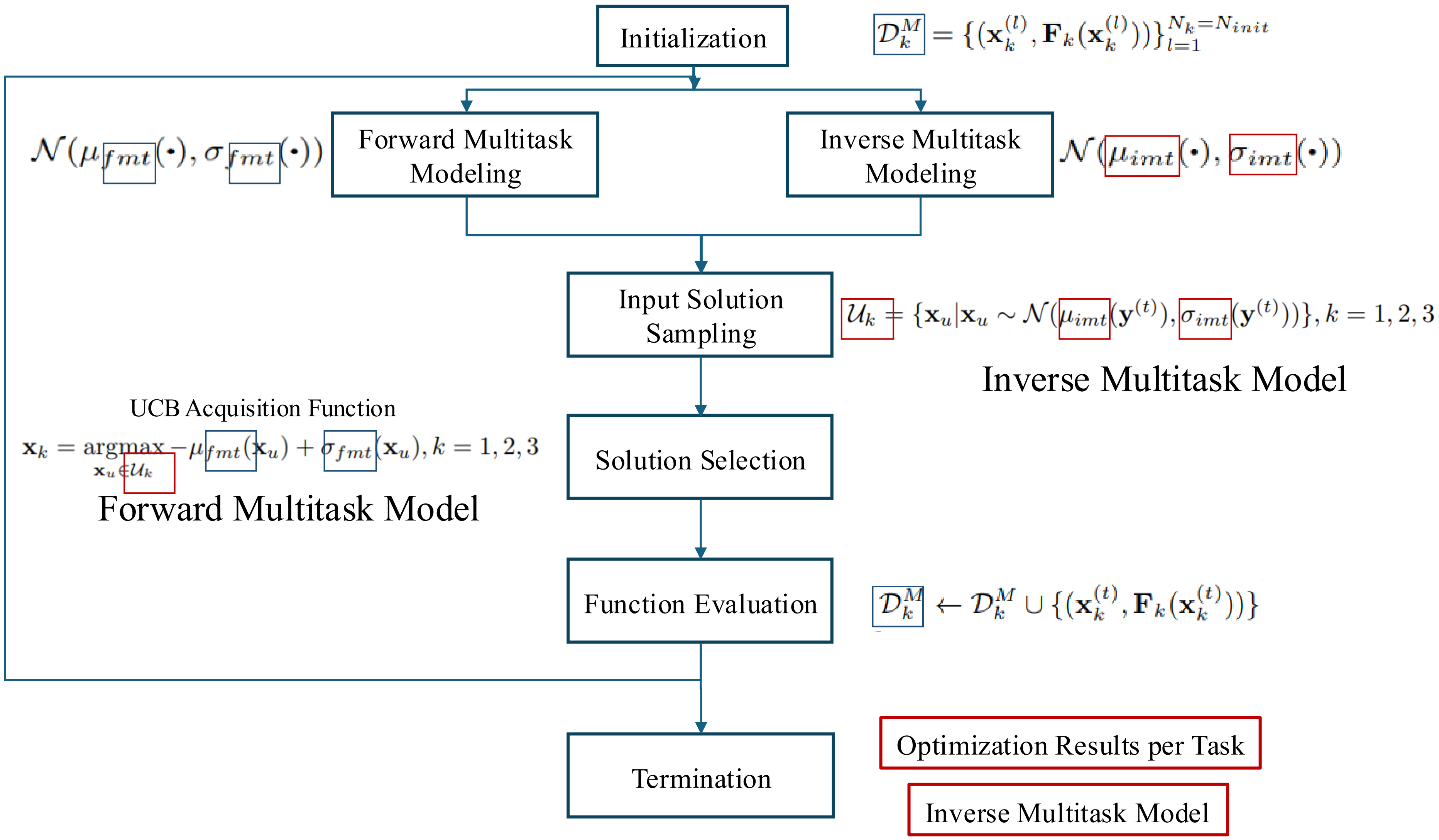
Iterative sequential transfer optimization can improve multitask optimizer

- Base multitask optimizer (F-invTrEMO)[1]:
- Optimize multiple multiobjective problems simultaneously through knowledge transfer via **forward multitask surrogate model** and **inverse multitask surrogate model**:

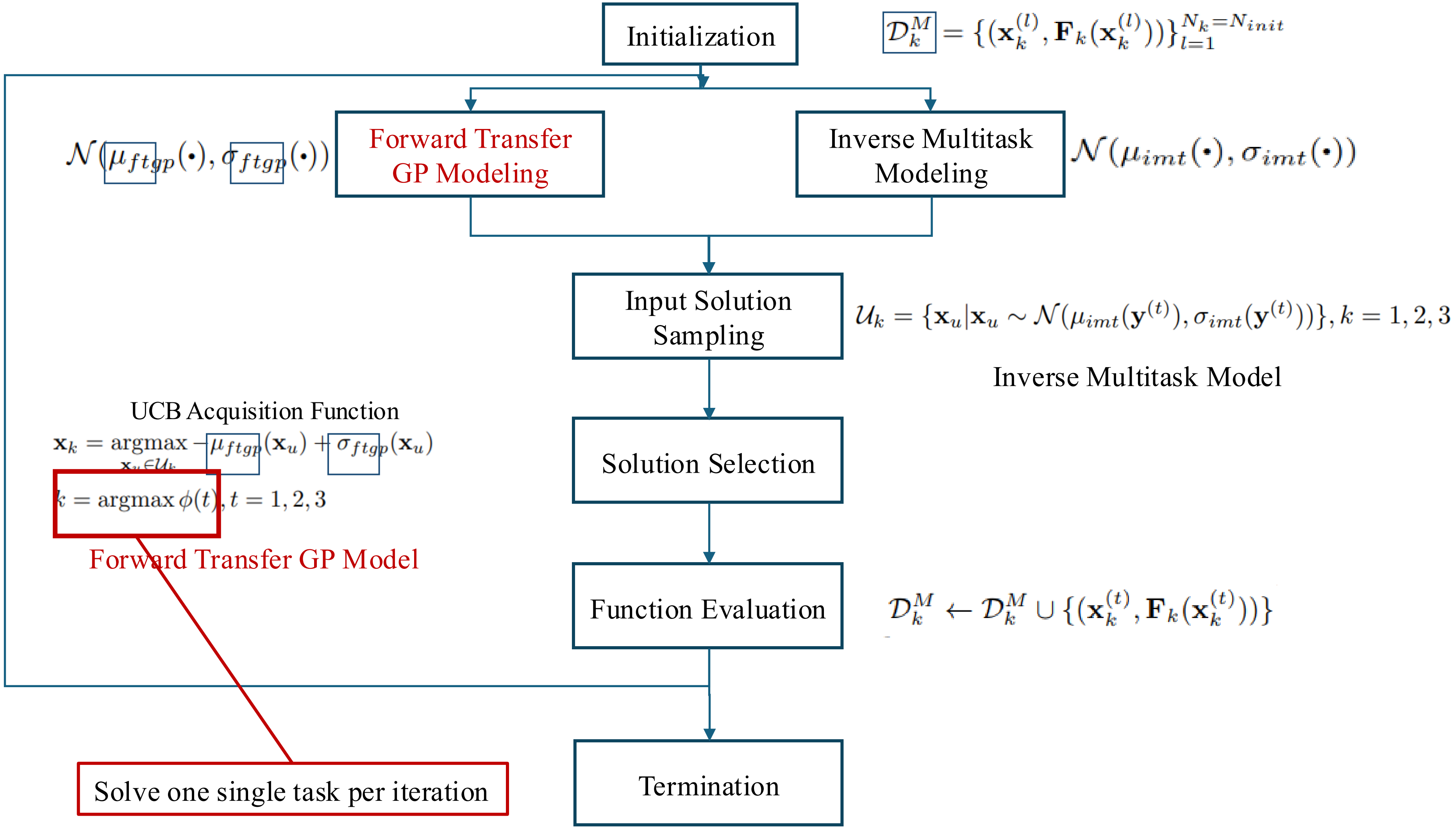


[1] Wei, Tingyang, et al. "Bayesian forward-inverse transfer for multiobjective optimization." International Conference on Parallel Problem Solving from Nature. Cham: Springer Nature Switzerland, 2024.

Base Multitask Optimizer: F-invTrEMO



Multitask Optimization via Iterative Sequential Transfer on F-invTrEMO



Multitask Optimization via Iterative Sequential Transfer on F-invTrEMO

- Forward Transfer GP Model: $\mathcal{N}(\mu_{ftgp}(\cdot), \sigma_{ftgp}(\cdot))$
  - For source tasks 1, 2, …, K and a target task. We build K transfer GP models:

$$\tilde{k}(\mathbf{x}, \mathbf{x}') = \begin{cases} \lambda k(\mathbf{x}, \mathbf{x}'), & \mathbf{x} \in \mathbf{X}_{\mathcal{S}} \ \& \ \mathbf{x}' \in \mathbf{X}_{\mathcal{T}} \\ & \text{or } \mathbf{x} \in \mathbf{X}_{\mathcal{T}} \ \& \ \mathbf{x}' \in \mathbf{X}_{\mathcal{S}} \\ k(\mathbf{x}, \mathbf{x}'), & \text{otherwise.} \end{cases}$$

  - $\lambda_{i,j} \in [-1,1]$ is a learnt kernel hyperparameter that can capture positive/negative correlations between source *i* and target *j*

$$\phi(t) = \min |\lambda_{i,t}|, \ \ i = 1, \ldots, K$$

Detecting negative transfer

  - We select the target task with the largest function $\phi(t)$

Less likely to occur negative transfer

  - Why?
    - GP-UCB optimization regret can be bounded by $\mathbf{I}([y^{(1)}, y^{(2)}, \ldots, y^{(N)}]; \boldsymbol{f}_T) = \frac{1}{2}\log|I + \sigma_n^{-2} K_{TT}|$
    - GP-UCB with transfer, the regret can be bounded by

$$\begin{aligned} &\mathbf{I}([y_{\mathcal{T}}^{(1)}, \ldots, y_{\mathcal{T}}^{(N_{\mathcal{T}})}]; \boldsymbol{f}_{\mathcal{T}} | \mathcal{D}_{\mathcal{S}}, \lambda, M) \\ &= \frac{1}{2} \log |I + \sigma_{\mathcal{T}}^{-2} (K_{\mathcal{T}\mathcal{T}} - \lambda^2 K'_{\mathcal{T}\mathcal{S}} (K_{\mathcal{S}\mathcal{S}} + \sigma_{\mathcal{S}}^2 I)^{-1} K'_{\mathcal{S}\mathcal{T}})| \end{aligned}$$

    - Larger $|\lambda_{i,t}|$ stand for a tighter regret bound, meaning a more accurate GP model can be constructed[1].

[1] Min, Alan Tan Wei, Abhishek Gupta, and Yew-Soon Ong. "Generalizing transfer Bayesian optimization to source-target heterogeneity." *IEEE Transactions on Automation Science and Engineering* 18.4 (2020): 1754-1765

Results

- Compared Methods:
  - ParEGO
  - F-invTrEMO
  - F-invTrEMO-IST (ours)

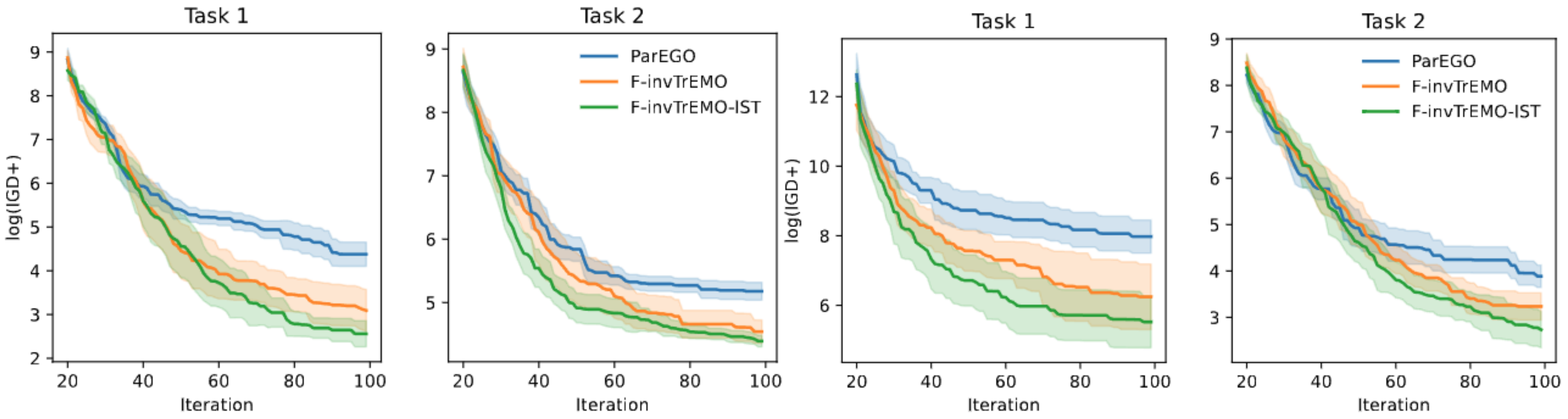


TABLE I
IGD+ Comparison Results on multitask Optimization Benchmark For ParEGO, F-invTrEMO, and F-invTrEMO-IST

Problems	Tasks	ParEGO	F-invTrEMO	F-invTrEMO-IST
CIHS	Task-1	1.516E+02 (1.545E+01) +	8.061E+01 (1.534E+01) +	**7.156E+01 (1.142E+01)**
	Task-2	4.877E+00 (3.150E-01) +	3.557E+00 (2.329E-01) ≈	**3.389E+00 (1.925E-01)**
CIMS	Task-1	2.523E+02 (2.057E+01) +	2.086E+02 (2.658E+01) +	**1.954E+02 (4.069E+01)**
	Task-2	4.174E+00 (2.983E-01) +	3.819E+00 (2.859E-01) +	**3.606E+00 (1.799E-01)**
CILS	Task-1	4.564E+01 (2.021E+00) +	4.435E+01 (1.592E+00) +	**4.113E+01 (1.601E+00)**
	Task-2	**3.050E-01 (1.040E-02)** -	7.938E-01 (5.305E-02) +	6.856E-01 (4.870E-02)
PIHS	Task-1	8.728E+01 (9.112E+00) +	3.234E+01 (8.086E+00) +	**1.399E+01 (1.978E+00)**
	Task-2	1.841E+02 (1.678E+01) +	9.913E+01 (1.033E+01) +	**8.109E+01 (3.761E+00)**
PIMS	Task-1	2.678E+00 (1.539E-01) +	**4.324E-01 (3.790E-02)** -	4.784E-01 (5.205E-02)
	Task-2	3.736E+02 (1.269E+01) +	3.503E+02 (5.258E+00) ≈	**3.407E+02 (4.165E+00)**
PILS	Task-1	**1.008E+00 (1.390E-02)** ≈	1.029E+00 (1.165E-02) ≈	1.041E+00 (9.500E-03)
	Task-2	1.423E+01 (6.948E-01) +	1.258E+01 (8.887E-01) +	**1.053E+01 (6.600E-01)**
NIHS	Task-1	3.174E+03 (6.287E+02) +	1.651E+03 (5.881E+02) +	**5.934E+02 (2.174E+02)**
	Task-2	5.820E+01 (4.852E+00) +	2.834E+01 (3.824E+00) +	**1.903E+01 (4.056E+00)**

Outline

- Background
  - Multitask Optimization
  - Multiobjective Multitask Optimization
  - Few-Shot Challenges
  - Solving Multitask Optimization via (Iterative) Sequential Transfer Optimization
- Instantiation 1: Iterative sequential transfer optimization can improve multitask optimization
- Instantiation 2: Iterative sequential transfer can improve sequential transfer optimizer in multitask optimization
- Instantiation 3: Improved multitask optimization in multitask hyperparameter optimization
- Conclusions and Potential Future Work

Iterative sequential transfer can improve sequential transfer optimizer in multitask optimization

- We directly use a sequential transfer optimizer to solve multitask optimization
  - AMTEA
  - Utilizing Gaussian Mixture Model plus transfer stacking to dynamically control the knowledge transfer from source to target.

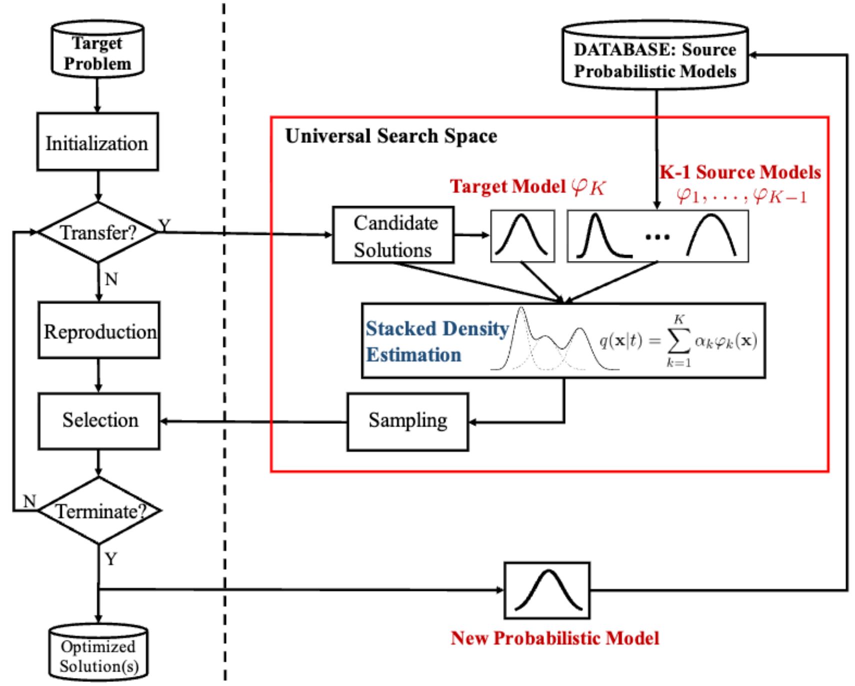


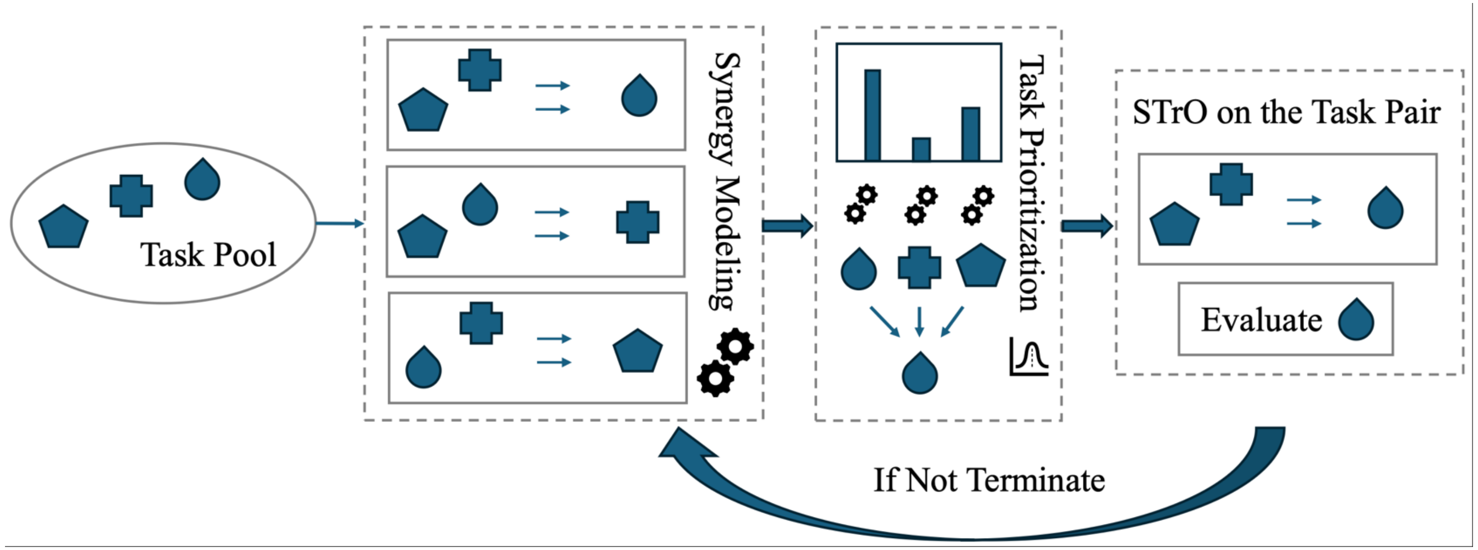
Fig. 1: A conceptual illustration of the proposed AMTEA.

Da, Bingshui, Abhishek Gupta, and Yew-Soon Ong. "Curbing negative influences online for seamless transfer evolutionary optimization." *IEEE transactions on cybernetics* 49.12 (2018): 4365-4378.

Iterative sequential transfer can improve sequential transfer optimizer in multitask optimization

- We directly use a sequential transfer optimizer to solve multitask optimization
  - AMTEA
  - Utilizing Gaussian Mixture Model plus transfer stacking to dynamically control the knowledge transfer from source to target.

- Under our framework, AMTEA can be directly used to solve a multitask optimization problem
  - Can iterative target selection make AMTEA better for solving multitask optimization?



Results

- Compared Methods:
  - ParEGO
  - AMTEA
  - AMTEA-IST (ours)

TABLE S-I
IGD+ COMPARISON RESULTS ON MULTITASK OPTIMIZATION BENCHMARK FOR PAREGO, AMTEA, AND AMTEA-IST

Problems	Tasks	ParEGO	AMTEA	AMTEA-IST
CIHS	Task-1	8.412E+01 (8.859E+00) +	6.392E+01 (5.758E+00) ≈	**6.314E+01 (4.842E+00)**
	Task-2	3.673E+00 (1.373E-01) ≈	**3.202E+00 (1.530E-01)** -	3.663E+00 (2.081E-01)
CIMS	Task-1	1.251E+02 (6.850E+00) +	1.026E+02 (1.079E+01) +	**8.299E+01 (8.457E+00)**
	Task-2	2.668E+00 (9.690E-02) -	**2.554E+00 (1.140E-01)** -	2.870E+00 (1.923E-01)
CILS	Task-1	**3.400E+01 (1.934E+00)** -	4.209E+01 (1.494E+00) +	3.810E+01 (1.712E+00)
	Task-2	2.243E-01 (5.650E-03) +	2.136E-01 (7.850E-03) +	**1.892E-01 (8.900E-03)**
PIHS	Task-1	6.839E+01 (1.157E+01) +	5.505E+01 (6.409E+00) +	**4.675E+01 (8.927E+00)**
	Task-2	1.454E+02 (7.436E+00) +	1.356E+02 (9.290E+00) ≈	**1.295E+02 (8.145E+00)**
PIMS	Task-1	2.158E+00 (2.162E-01) +	8.786E-01 (1.249E-01) ≈	**8.410E-01 (4.580E-02)**
	Task-2	3.377E+02 (4.546E+00) ≈	3.355E+02 (4.538E+00) ≈	**3.335E+02 (4.739E+00)**
PILS	Task-1	1.016E+00 (6.050E-03) +	9.833E-01 (9.000E-03) ≈	**9.482E-01 (1.635E-02)**
	Task-2	9.252E+00 (4.003E-01) +	**7.599E+00 (3.862E-01)** ≈	7.877E+00 (5.489E-01)
NIHS	Task-1	9.446E+02 (1.847E+02) +	4.015E+02 (7.916E+01) +	**3.457E+02 (6.560E+01)**
	Task-2	3.584E+01 (2.767E+00) +	3.466E+01 (3.816E+00) ≈	**3.344E+01 (1.626E+00)**

Outline

- Background
  - Multitask Optimization
  - Multiobjective Multitask Optimization
  - Few-Shot Challenges
  - Solving Multitask Optimization via (Iterative) Sequential Transfer Optimization
- Instantiation 1: Iterative sequential transfer optimization can improve multitask optimization
- Instantiation 2: Iterative sequential transfer can improve sequential transfer optimizer in multitask optimization
- **Instantiation 3: Improved multitask optimization in multitask hyperparameter optimization**
- Conclusions and Potential Future Work

Experiments and Results

- Problems:
  - Hyperparameter Optimization Problems[1]
    - (HPO-1)
    - XG-Boost on 3 machine learning classification tasks
      - Credit Approval[2]
      - Medical Diagnosis[2]
      - Speech Recognition[2]
    - (HPO-2)
    - XG-Boost in 'gbtree' mode on medical diagnosis problem
    - XG-Boost in 'dart' mode on medical diagnosis problem

    - Objectives?
      - Classification error
      - Model RAM (Important to consider during deployment)
      - IAS – Interaction Strength of Features (A quantitative measure of model interpretability)

TABLE II
IGD+ COMPARISON RESULTS ON MULTITASK HYPERPARAMETER TUNING PROBLEMS FOR F-INVTREMO AND F-INVTREMO-IST

Problems	Tasks	F-invTrEMO	F-invTrEMO-IST
HPO-1	Task-1	0.0502 (0.0028) +	**0.0455 (0.0021)**
	Task-2	0.0439 (0.0020) ≈	**0.0423 (0.0014)**
HPO-2	Task-1	0.0325 (0.0017) +	**0.0303 (0.0031)**
	Task-2	0.0282 (0.0019) ≈	**0.0279 (0.0014)**
	Task-3	0.0421 (0.0030) +	**0.0392 (0.0023)**

[1] Pfisterer, Florian, et al. "Yahpo gym-an efficient multi-objective multi-fidelity benchmark for hyperparameter optimization." International Conference on Automated Machine Learning. PMLR, 2022.
[2] Vanschoren, Joaquin, et al. "OpenML: networked science in machine learning." ACM SIGKDD Explorations Newsletter 15.2 (2014): 49-60.

Outline

- Background
  - Multitask Optimization
  - Multiobjective Multitask Optimization
  - Few-Shot Challenges
  - Solving Multitask Optimization via (Iterative) Sequential Transfer Optimization
- Instantiation 1: Iterative sequential transfer optimization can improve multitask optimization
- Instantiation 2: Iterative sequential transfer can improve sequential transfer optimizer in multitask optimization
- Instantiation 3: Improved multitask optimization in multitask hyperparameter optimization
- Conclusions and Potential Future Work

Conclusion

- In few-shot settings, multitask optimization can be challenging.
- We solve it by modeling multitask optimization as a sequence of sequential transfer optimizer.
- A curated iterative sequential transfer strategy can enhance the performance on MTO based on both multitask optimizer and sequential transfer optimizer
- We call for the attention to the unification between two fields:
  - Multitask optimization
  - Sequential transfer

Further Directions

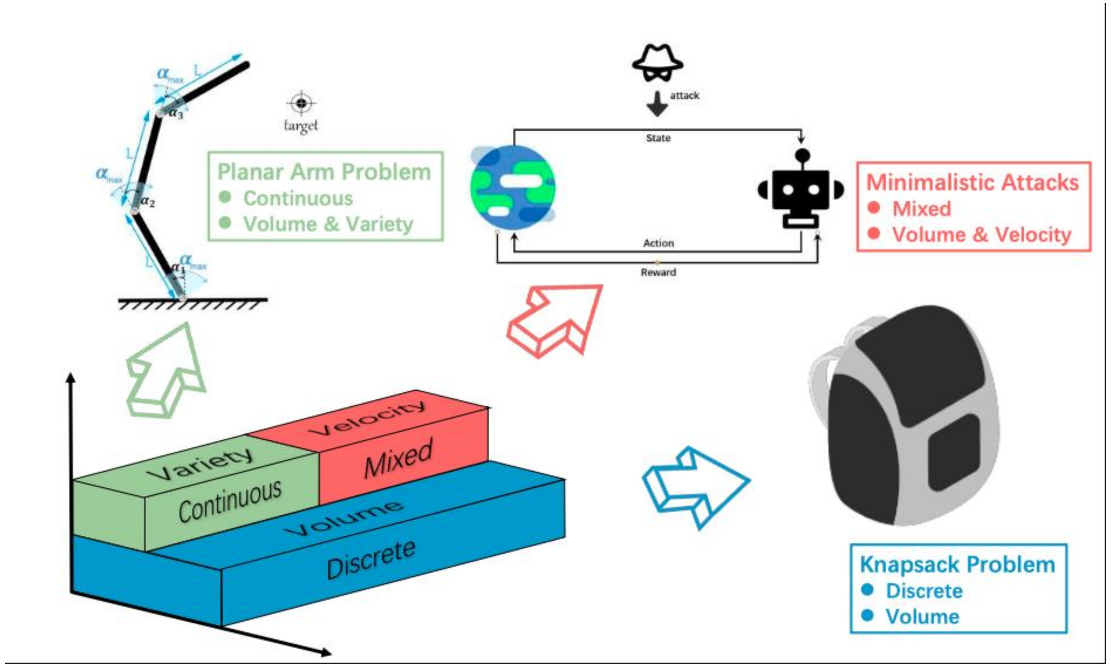
- We call for the attention to the unification between two fields:
  - Multitask optimization
  - Sequential transfer

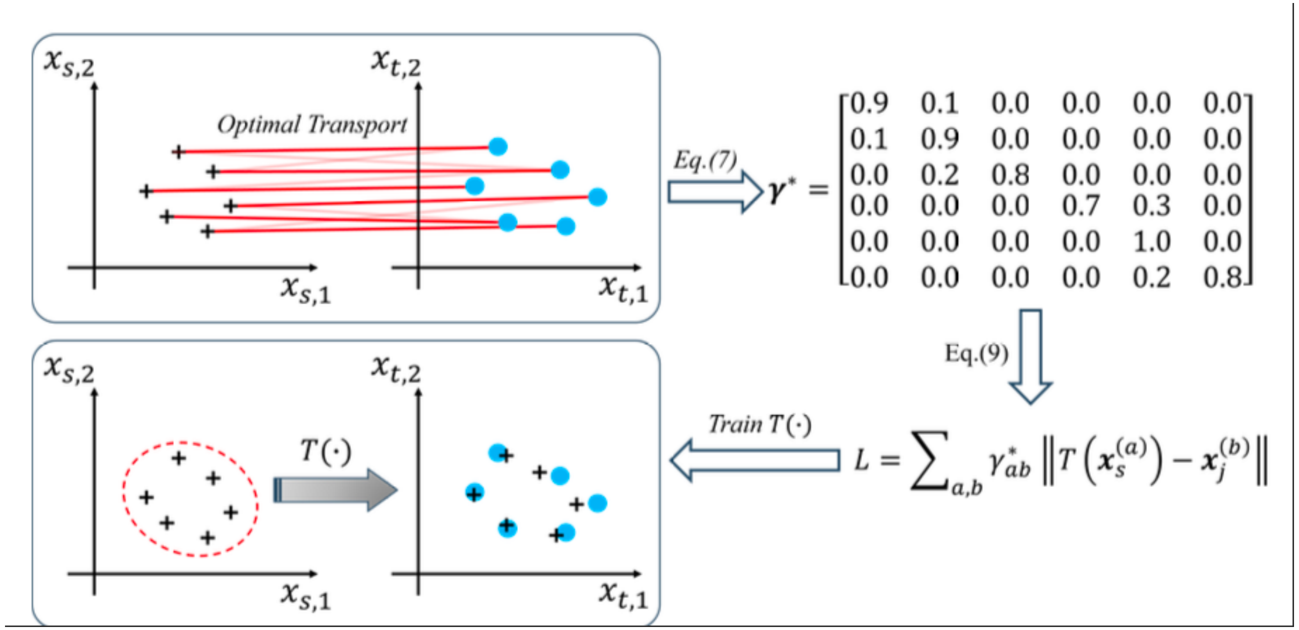
Bridging the Gap Between Theory and Practice: Benchmarking Transfer Evolutionary Optimization

Yaqing Hou, Wenqiang Ma, Abhishek Gupta, Kavitesh Kumar Bali, Hongwei Ge, Qiang Zhang, Carlos A. Coello Coello *Fellow, IEEE* and Yew-Soon Ong, *Fellow, IEEE*

Optimal Transport-Based Distributional Pairing in Transfer Multiobjective Optimization

Jiao Liu, Weiming Liu, Joel Tay Wei En, Caishun Chen, Puay Siew Tan, and Yew-Soon Ong





Thank you!