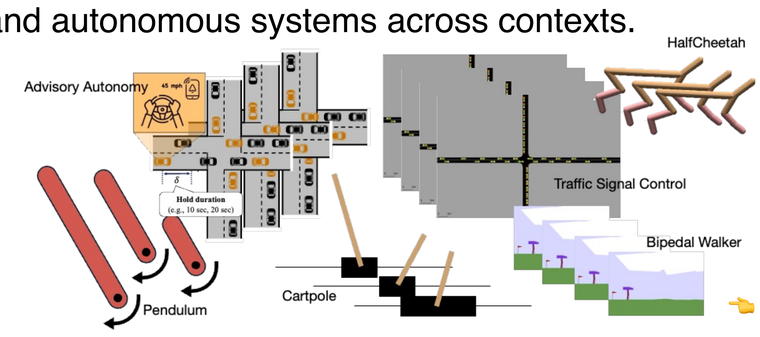
# Efficient Source Tasks Selection for Zero-shot Transfer in Contextual Reinforcement Learning

Jung-Hoon Cho, Vindula Jayawardana, Sirui Li, Cathy Wu
Massachusetts Institute of Technology

## Introduction

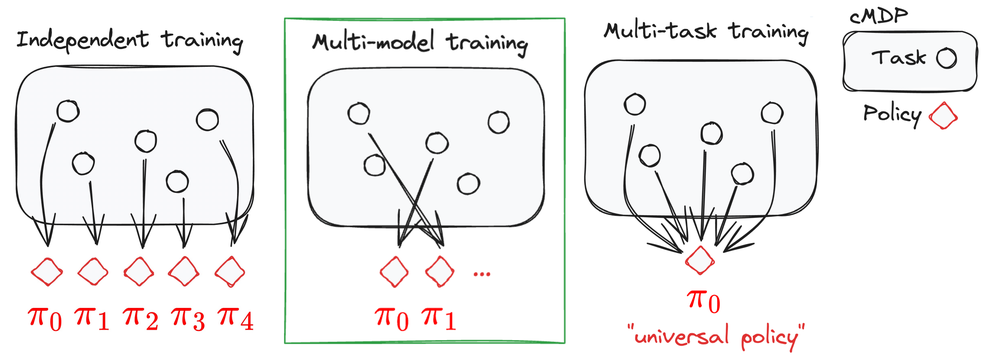
- Despite deep reinforcement learning (DRL) remarkable success across domains, DRL models often exhibit **brittleness when exposed to small variations in task settings**.
- **Contextual Markov Decision Process (CMDP)** extends traditional MDPs with additional contextual information, leading to multiple, closely related MDPs.
- **Motivation**: Aim to improve generalization for practical applications like robotics and autonomous systems across contexts.




**Examples of CMDP**

## Model-Based Transfer Learning

- **Resource-intensive traditional DRL approaches**
  (e.g. Independent training or multi-task training)




### MBTL Algorithm

**Strategically select source tasks for multi-model training**

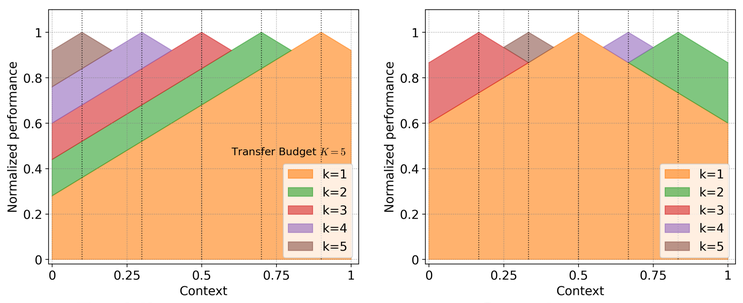
**How?** Simple & explicit modeling of generalization gap (model-based)

**Algorithm 1** Model-based Transfer Learning (MBTL)

**Input:** CMDPs $c\mathcal{M}(x)$, Task $x \in X$, Transfer budget $K$
**Output:** $\boldsymbol{\pi}$ and $V$
*Initialize* : $J, V = 0\ \forall x \in X, \boldsymbol{\pi} = \{\}, k = 1$
**while** $k \leq K$ **do**
  $x_k \leftarrow$ **NextTask**$(J_{k-1}, V_{k-1})$
  $\pi_k \leftarrow \text{Train}(\mathcal{M}(x_k))$
  $\boldsymbol{\pi} \leftarrow \boldsymbol{\pi} \cup \{\pi_k\}$
  Update $J(x), V(x)$
  $k \leftarrow k + 1$
**end while**
**return** $\boldsymbol{\pi}$ and $V$

### Simple methods [1]

**Assumptions:** Training performance is constant.



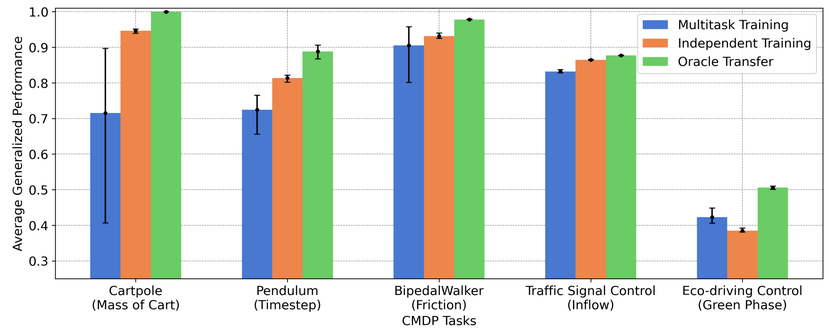

**Equidistant strategy** **Greedy strategy**

### Other Baselines

Independent training, Multitask training, Oracle

## Zero-shot Transfer

- **Transfer Learning**: Utilize already trained policy to solve another task
  *"Training is expensive, but inference (evaluation) is cheap."*
- Zero-shot transfer is **cheap & works remarkably well**

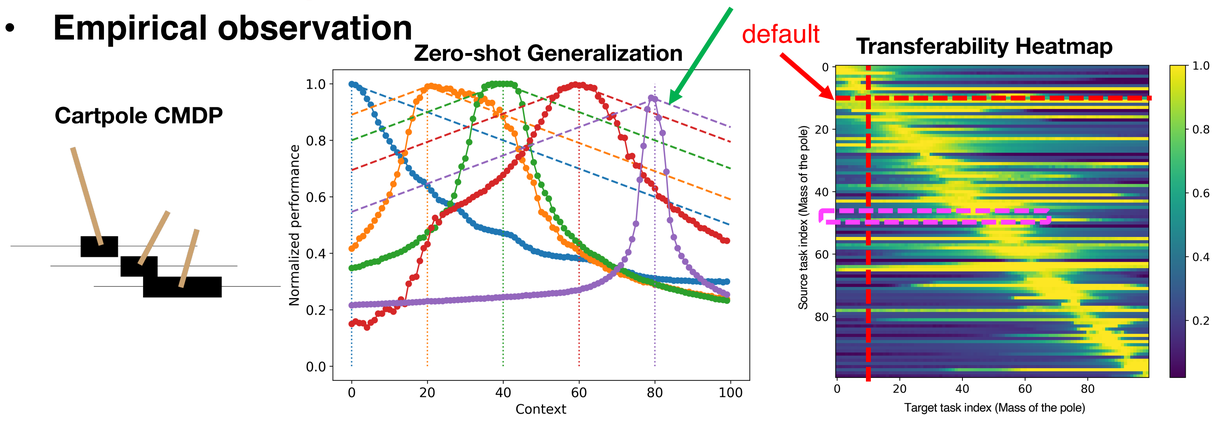



- **Challenges**: It's difficult to decide which tasks to train first! 🤯
- **Algorithmic question:** How to select which task to train?
  → **Source task selection problem**
- Optimize task selection to maximize the performance across contexts

### Generalization Gap

- **Empirical observation**

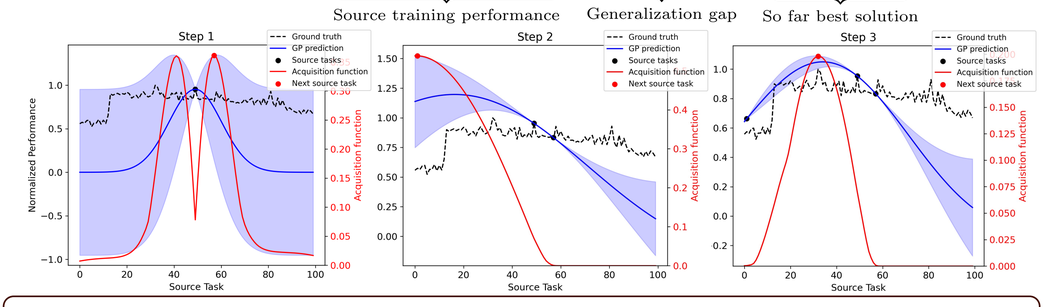
Can we model this with **linear function**?




### MBTL with Bayesian optimization

- **Challenge**: Training performance is not constant!
- **Gaussian Process (GP)** to estimate training performance
- Acquisition function (estimated performance considering gen. gap)

$$a(x; x_{(1:k-1)}) = \mathbb{E}_{x' \in X}[[\underbrace{\mu_{k-1}(x) + \beta_k^{1/2}\sigma_{k-1}(x)}_{\text{Source training performance}} - \underbrace{\Delta J(x, x')}_{\text{Generalization gap}} - \underbrace{U(x'; x_{(1:k-1)})}_{\text{So far best solution}}]_+]$$




**Theorem 3**: MBTL-GP achieves **sublinear regret** $R_k \leq O(\sqrt{k \log k})$ with high probability

# Numerical Experiments

## Standard control CMDP benchmarks

Pendulum, Cartpole, HalfCheetah, BipedalWalker

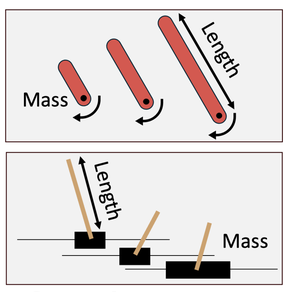



Table 2: Comparative performance of different methods on standard control CMDP tasks

| Benchmark | | Baselines | | | MBTL (Ours) | | | Oracle |
|---|---|---|---|---|---|---|---|---|
| Task | Context Variation | Random | Independent | Multitask | MBTL-GS | MBTL-ES | MBTL-GP | Oracle Transfer |
| Number of Trained Models | | $k$ | $N$ | 1 | $k$ | $K$ | $k$ | $N$ |
| **Pendulum** | Length | 0.7270 | 0.7383 | 0.6830 | 0.7327 | 0.7092 | **0.7697** | 0.7969 |
| **Pendulum** | Mass | 0.6329 | 0.6237 | 0.5793 | 0.6408 | 0.6092 | **0.6827** | 0.7132 |
| **Pendulum** | Timestep | 0.7989 | 0.8135 | 0.7247 | **0.8177** | 0.7541 | 0.8141 | 0.8801 |
| **Cartpole** | Mass of Cart | 0.7221 | **0.9466** | 0.7153 | 0.6501 | 0.7516 | 0.8212 | 0.9838 |
| **Cartpole** | Length of Pole | 0.8121 | 0.9110 | 0.5441 | 0.8217 | 0.8428 | **0.9124** | 0.9875 |
| **Cartpole** | Mass of Pole | 0.8858 | **0.9560** | 0.6073 | 0.8744 | 0.7909 | 0.9351 | 1.0000 |
| **BipedalWalker** | Gravity | 0.9330 | 0.9281 | 0.7898 | 0.9359 | **0.9494** | 0.9393 | 0.9674 |
| **BipedalWalker** | Friction | 0.9650 | 0.9317 | 0.9051 | 0.9645 | 0.9664 | **0.9713** | 0.9778 |
| **BipedalWalker** | Scale | 0.8605 | 0.8694 | 0.7452 | 0.8792 | 0.8496 | **0.8886** | 0.9107 |
| **HalfCheetah** | Gravity | 0.8542 | 0.6679 | 0.6292 | 0.8663 | 0.8634 | **0.9073** | 0.9544 |
| **HalfCheetah** | Friction | 0.8567 | 0.6693 | 0.7242 | 0.8591 | 0.8703 | **0.9274** | 0.9663 |
| **HalfCheetah** | Stiffness | 0.8533 | 0.6561 | 0.7007 | 0.8785 | 0.7817 | **0.9146** | 0.9674 |
| **Average** | | 0.8251 | 0.8093 | 0.6957 | 0.8267 | 0.8116 | **0.8736** | 0.9255 |

†Higher the better. The bold indicates the best for each benchmark, excluding the Oracle transfer. Detailed results with variance for each method are provided in Appendix A.2.2.

## Transportation CMDP benchmarks

Traffic Signal Control, Eco-driving control, Advisory autonomy

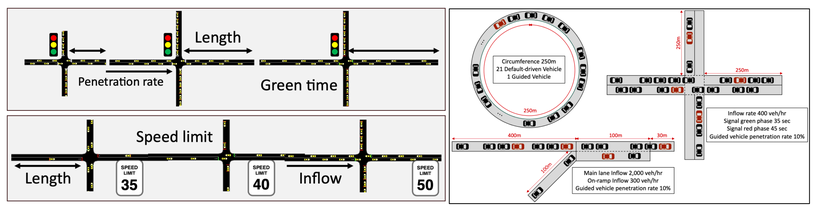



Table 1: Comparative performance of different methods on traffic CMDP tasks

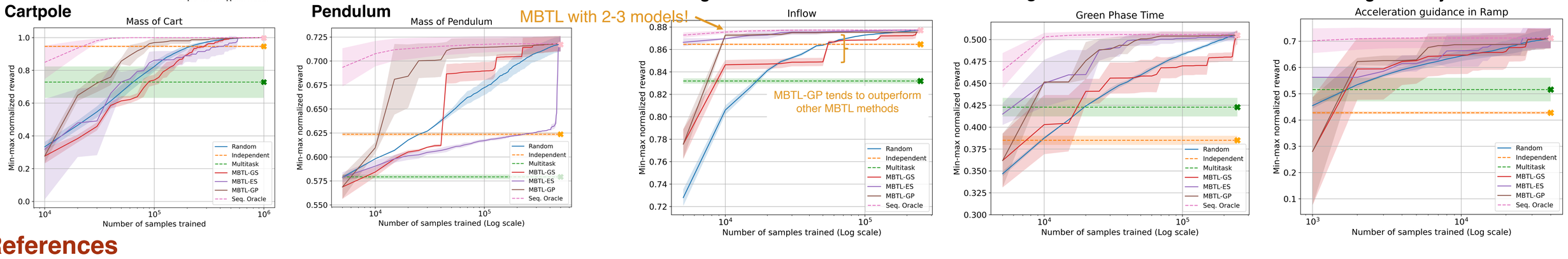
| Benchmark | | Baselines | | | MBTL (Ours) | | | Oracle |
|---|---|---|---|---|---|---|---|---|
| Task | Context Variation | Random | Independent | Multitask | MBTL-GS | MBTL-ES | MBTL-GP | Oracle Transfer |
| Number of Trained Models | | $k$ | $N$ | 1 | $k$ | $K$ | $k$ | $N$ |
| **Traffic Signal** | Road Length | 0.9249 | **0.9409** | 0.8242 | 0.9278 | 0.9213 | 0.9371 | 0.9409 |
| **Traffic Signal** | Inflow | 0.8457 | 0.8646 | 0.8319 | 0.8496 | **0.8700** | 0.8673 | 0.8768 |
| **Traffic Signal** | Speed Limit | 0.8821 | 0.8857 | 0.6083 | **0.8862** | 0.8858 | 0.8854 | 0.8876 |
| **Eco-Driving** | Penetration Rate | 0.5959 | 0.5260 | 0.1945 | 0.5827 | 0.5934 | **0.6323** | 0.6660 |
| **Eco-Driving** | Inflow | 0.4774 | 0.4061 | 0.2229 | 0.4673 | 0.4705 | **0.5108** | 0.5528 |
| **Eco-Driving** | Green Phase | 0.4406 | 0.3850 | 0.4228 | 0.4431 | 0.4557 | **0.4700** | 0.5027 |
| **AA-Ring-Acc** | Hold Duration | 0.8924 | 0.8362 | 0.9209 | 0.8776 | 0.9057 | **0.9242** | 0.9552 |
| **AA-Ring-Vel** | Hold Duration | 0.9785 | 0.9589 | 0.9720 | 0.9807 | 0.9772 | **0.9816** | 0.9822 |
| **AA-Ramp-Acc** | Hold Duration | 0.6050 | 0.4276 | 0.5158 | 0.6143 | 0.5956 | **0.6318** | 0.7111 |
| **AA-Ramp-Vel** | Hold Duration | 0.6690 | 0.5473 | 0.5034 | 0.5907 | 0.6787 | **0.7182** | 0.7686 |
| **Average** | | 0.7312 | 0.6778 | 0.6017 | 0.7220 | 0.7354 | **0.7559** | 0.7844 |

†Higher the better. The bold indicates the best for each benchmark, excluding the Oracle transfer. Detailed results with variance for each method are provided in Appendix A.2.2.
‡AA: Advisory autonomy tasks, Ring: Single lane ring, Ramp: Highway ramp, Acc: Acceleration guidance, Vel: Speed guidance.

### Traffic Signal Control

### Eco-driving control

### Real-time driving advisory




## References

[1] J.-H. Cho, S. Li, J. Kim, and C. Wu, "Temporal Transfer Learning for Traffic Optimization with Coarse-grained Advisory Autonomy." In revision.
[2] J.-H. Cho, V. Jayawardana, S. Li, and C. Wu, "Model-Based Transfer Learning for Contextual Reinforcement Learning." EWRL17, NeurIPS 2024. To Appear.

Massachusetts Institute of Technology

Model-Based Transfer Learning (MBTL) effectively selects training data for solving contextual MDPs by modeling the generalization gap with respect to the context.

**(20-35x more sample efficient than independent & multi-task baselines)**