Using Task Features for Zero-Shot Knowledge Transfer in Lifelong Learning

**Summary**
Knowledge transfer between tasks requires an accurate estimate of the inter-task relationships, which is inefficient in lifelong learning settings. We develop a lifelong reinforcement learning method that incorporates high-level task descriptors to model the inter-task relationships.

→ Improves the performance of the learned task policy
→ Accurately predicts the policy for a new task via zero-shot learning, given only the task description

**Motivation**
Lifelong learning accelerates training of each consecutive new task by building upon previously acquired knowledge via transfer

→ Relevant knowledge/tasks must be identified before transfer can occur
→ Requires interacting with the new task (i.e., sampling trajectories, learning, etc.) to characterize it

**Background: Policy Gradient (PG) Methods**
- Agent interacts with environment, taking consecutive actions
- PG methods support continuous state and action spaces
  - Have shown recent success in applications to robotic control

**Background: Task Descriptors**
- Policy for task
- Improves the performance of the learned task policy
- Requires interacting with the new task (i.e., sampling trajectories, learning, etc.) to characterize it

**Alternative Idea:** Can we use a high-level description of the task to identify relevant knowledge for transfer in lifelong learning?

Example task descriptor: physical specification of a quadrotor

**Sharing Knowledge Between Multiple Tasks**
- Policy for task: \( \pi_t : A \times X \to [0, 1] \)
- Factor the policy as \( \theta_t = L s_t \)

**Multi-Task Learning (TaDeML)**
- Fit via alternating optimization

**Lifelong Learning (TaDeLL)**
1. Merge \( L \) and \( D \) into single dictionary \( K \)
2. Estimate policy \( \alpha_t \) via single-task learning
3. Sparse code estimated policy and descriptor in \( K \)
4. Update \( L \) and \( D \)

**Incorporating Task Descriptors into Lifelong Learning**
Key Idea: Relate policy parameters and task descriptors via coupled dictionary learning

**Algorithm 1**
1. \( L = \text{RandomMatrix}_{A \times \mu} \)
2. while some task (2D or m(t)) is available do
   1. Compute \( \alpha^0 \) and \( \Theta_0 \) from \( \alpha^0 \)
   2. \( \Theta(t) = \text{sampleTrajectory}(Z(t)) \)
   3. \( \Theta(t) = \text{sampleTrajectory}(Z(t), \alpha(t)) \)
   4. end if

**Experimental Results on Dynamical Systems**
- Train on 40 different consecutive control tasks, transfer to new task
- Task descriptors improve policies from multi-task and lifelong learning

**Acknowledgements and Notes**
This research was supported by ONR grant N00014-11-1-0139 and AFRL grant FA8750-14-1-0069.
* Authors contributed equally