|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | | Dynamics | | | | | | | |
| Fixed | | | | Changing | | | |
| Rewards | Fixed | *No need for unsupervised or meta-learning* | | | | See [Figure 2](#figure2) | Model-free | Model-based | Hybrid |
| Meta-RL algorithm | MB-MPO [6] | * GrBAL/ReBAL [2] * MOLe [11] | *More challenging /advanced case* |
| Task Acquisition mechanism  (see [table 2](#table2)) | * ADR [9], [16] * LSDR [18] * Option 1 | |
| Changing | See [Figure 1](#figure1) | Model-free | Model-based | Hybrid | *More challenging/advanced case* (Option 5?) | | | |
| Meta-RL algorithm | MAML [8], etc | Option 2 | *More challenging /advanced case* |
| Task Acquisition mechanism  (see [table 2](#table2)) | * DIAYN [3] * DADS [4] (Option 3?) * Meta-ADR [15] * VDS [17] * Option 4 (see [table 3](#table3)) |

Table 1: Current Landscape and Available Options

# List of Options:

* [Option 1](#option1)
* [Option 2](#option2)
* [Option 3](#option3)
* [Option 4](#option4)
* [Option 5](#option5)
* [Option 6](#option6)
* [Option 7](#option7)

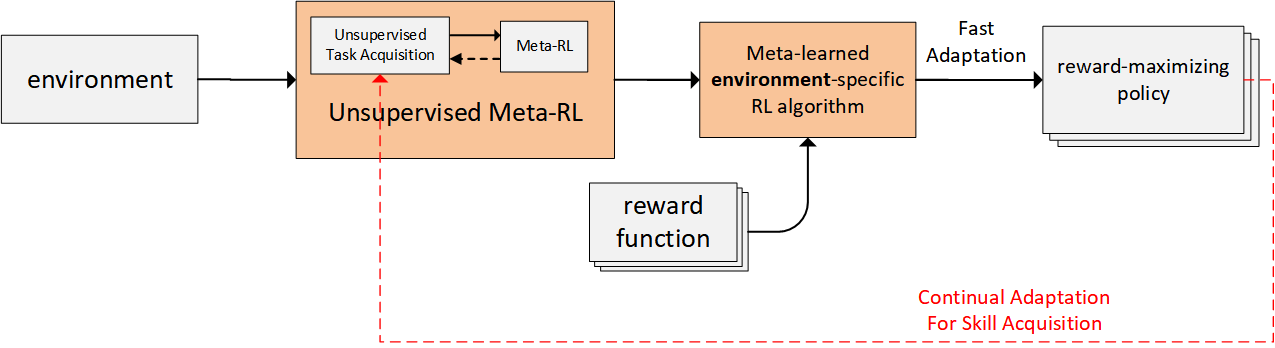


Figure 1: Unsupervised Meta-RL: Changing Rewards and Fixed Environment Case. (Adapted from [1])

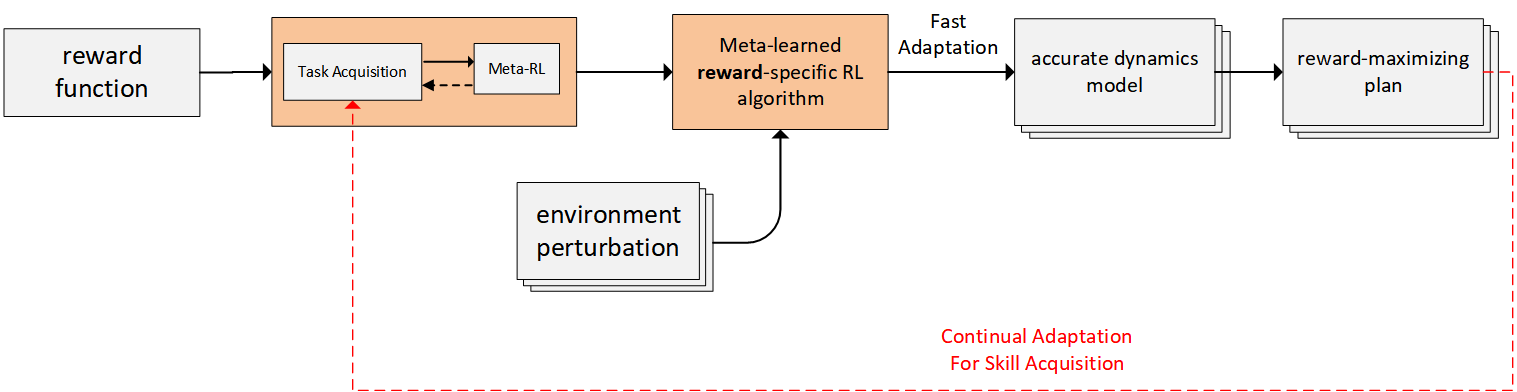


Figure : Automatic task acquisition for Meta-RL: Fixed Rewards and Changing Environment Case

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Type of Task | Algorithm | Task distribution attribute | | | Limitations |
| What parameters to vary | Their ranges | How to sample from their ranges |
| Either | Meta-RL algorithms |  |  |  |  |
| Automatic Curriculum Generation Methods |  |  | **✓** |  |
| Environment-based | ActiveDR |  |  | **✓** | Target environment is kind of known (via the reference env/params) |
| AutoDR |  | **✓** |  |  |
| LSDR |  | **✓** | **✓** | There might be some coupling happening with the learned policy |
| Reward-based | Meta-ADR |  |  | **✓** | Treats the difference in policy behaviour between pre and post adaptation phases as a sign of difficulty of the current task to the current policy (including positive transfer/adaptation) |
| VDS |  |  | **✓** | Applied to goal-conditioned RL? |
| DIAYN | **✓** | **✓** | **✓** | Current working implementation is based on uniform sampling |
| DADS | **✓** | **✓** | **✓** |

Table : Algorithms and what they automate in terms of task distribution attributes (Option 6?)

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Limitations / Advantages | Continual Meta-RL algorithm/framework | | | | |
| Continual skill adaptation | Online MAML [19]  (FTML) | TAML [20] | Continual MAML [21] | MOLe |
| Memory |  | Expensive  (stores all encountered tasks) |  |  | Expensive  (the size of the ensemble grows over time and is unbounded) |
| Compute |  | Expensive  (meta-trains on all stored tasks continually) | Expensive  (uses BGD optimizer which is computationally expensive) |  |  |
| Performance (and efficiency) on new tasks | Initially [at least] better |  |  |  |  |
| Paradigm | Interleaving training and testing periods continually | Data stream | | | |
| Overfitting | Meta-training on skills could be more robust then on specific tasks |  |  |  |  |

Table 3: Comparison of Continual skill adaptation against candidate continual meta-RL methods

|  |  |  |  |
| --- | --- | --- | --- |
| Setting | Criteria | Algorithm group 1 | Algorithm group 2 |
| Changing rewards | asymptotic performance, sample/data efficiency, speed (at train & test times) and generalizability | Learning adaptable priors over a policy (e.g. via MAML) | model-based RL algorithms with online planning (e.g. PETS [5], MBPO [12], VMBPO [13], MnM [14], etc) |
| Learning adaptable priors over a policy whose tasks are reward functions (e.g. via MAML) | learning adaptable priors over a policy whose tasks are dynamic models (e.g. via MB-MPO) |
| Automatic task acquisition of predictable, diverse skills with online planning via composing acquired skills (DADS) | model-based RL algorithms with online planning (e.g. MBPO, VMBPO, MnM, etc) |
| automatic diverse skill acquisition (DIAYN) with learned adaptable priors over a policy (e.g. via MAML) |
| Changing dynamics | asymptotic performance, sample/data efficiency, speed (at train & test times) and generalizability | Learning adaptable priors over a model (e.g. via GrBAL/ReBAL) | a method which utilizes an ensemble of models (e.g. PETS or MB-MPO) |
| Learning adaptable priors over a policy whose tasks are dynamic models of an ensemble (e.g. via MB-MPO) | a simple ensemble of dynamic models (e.g. PETS) |
| Quality of Exploration  In the algorithms which use an ensemble of models (e.g. PETS & MB-MPO) | implicit exploration via using a distribution of reward functions in learning a model | exploration via disagreement [7] (and/or other types of curiosity or intrinsic-reward based exploration, etc) |

Table 4: Options for Comparative studies (Option 7?)

# References:

[1] Gupta, A., Eysenbach, B., Finn, C., & Levine, S. (2018). Unsupervised Meta-Learning for Reinforcement Learning. ArXiv, abs/1806.04640.

[2] Nagabandi, A., Clavera, I., Liu, S., Fearing, R., Abbeel, P., Levine, S., & Finn, C. (2019). Learning to Adapt in Dynamic, Real-World Environments through Meta-Reinforcement Learning. arXiv: Learning.

[3] Eysenbach, B., Gupta, A., Ibarz, J., & Levine, S. (2019). Diversity is All You Need: Learning Skills without a Reward Function. ArXiv, abs/1802.06070.

[4] Sharma, A., Gu, S., Levine, S., Kumar, V., & Hausman, K. (2020). Dynamics-Aware Unsupervised Discovery of Skills. ArXiv, abs/1907.01657.

[5] Chua, K., Calandra, R., McAllister, R., & Levine, S. (2018). Deep Reinforcement Learning in a Handful of Trials using Probabilistic Dynamics Models. NeurIPS.

[6] Clavera, I., Rothfuss, J., Schulman, J., Fujita, Y., Asfour, T., & Abbeel, P. (2018). Model-Based Reinforcement Learning via Meta-Policy Optimization. *CoRL*

[7] Pathak, D., Gandhi, D., & Gupta, A. (2019). Self-Supervised Exploration via Disagreement. ArXiv, abs/1906.04161.

[8] Finn, C., Abbeel, P., & Levine, S. (2017). Model-Agnostic Meta-Learning for Fast Adaptation of Deep Networks. ICML.

[9] Mehta, B., Diaz, M., Golemo, F., Pal, C.J., & Paull, L. (2019). Active Domain Randomization. CoRL.

[10] Gutierrez, R.L., & Leonetti, M. (2020). Information-theoretic Task Selection for Meta-Reinforcement Learning. *ArXiv, abs/2011.01054.*

[11] Nagabandi, A., Finn, C., & Levine, S. (2019). Deep Online Learning via Meta-Learning: Continual Adaptation for Model-Based RL. *ArXiv, abs/1812.07671.*

[12] Janner, M., Fu, J., Zhang, M., & Levine, S. (2019). When to Trust Your Model: Model-Based Policy Optimization. *NeurIPS.*

[13] Chow, Y., Cui, B., Ryu, M.K., & Ghavamzadeh, M. (2021). Variational Model-based Policy Optimization. *IJCAI.*

[14] Eysenbach, B., Khazatsky, A., Levine, S., & Salakhutdinov, R. (2021). Mismatched No More: Joint Model-Policy Optimization for Model-Based RL. *ArXiv, abs/2110.02758.*

[15] Mehta, B., Deleu, T., Raparthy, S.C., Pal, C., & Paull, L. (2020). Curriculum in Gradient-Based Meta-Reinforcement Learning. ArXiv, abs/2002.07956.

[16] OpenAI, Akkaya, I., Andrychowicz, M., Chociej, M., Litwin, M., McGrew, B., Petron, A., Paino, A., Plappert, M., Powell, G., Ribas, R., Schneider, J., Tezak, N.A., Tworek, J., Welinder, P., Weng, L., Yuan, Q., Zaremba, W., & Zhang, L. (2019). Solving Rubik's Cube with a Robot Hand. ArXiv, abs/1910.07113.

[17] Zhang, Y., Abbeel, P., & Pinto, L. (2020). Automatic Curriculum Learning through Value Disagreement. ArXiv, abs/2006.09641.

[18] Mozifian, M., Higuera, J.C., Meger, D., & Dudek, G. (2019). Learning Domain Randomization Distributions for Transfer of Locomotion Policies. ArXiv, abs/1906.00410.

[19] Finn, C., Rajeswaran, A., Kakade, S.M., & Levine, S. (2019). Online Meta-Learning. ICML.

[20] He, X., Sygnowski, J., Galashov, A., Rusu, A.A., Teh, Y.W., & Pascanu, R. (2019). Task Agnostic Continual Learning via Meta Learning. ArXiv, abs/1906.05201.

[21] Caccia, M., Rodríguez, P., Ostapenko, O., Normandin, F., Lin, M., Page-Caccia, L., Laradji, I.H., Rish, I., Lacoste, A., Vázquez, D., & Charlin, L. (2020). Online Fast Adaptation and Knowledge Accumulation (OSAKA): a New Approach to Continual Learning. NeurIPS.

# Repositories:

|  |  |  |
| --- | --- | --- |
| Algorithm | Official | Others |
| DIAYN | * <https://github.com/haarnoja/sac/blob/master/sac/algos/diayn.py> * <https://github.com/ben-eysenbach/sac/blob/master/sac/algos/diayn.py> | * <https://github.com/navneet-nmk/Hierarchical-Meta-Reinforcement-Learning/blob/master/maml_rl/policies/empowerment_skills.py> * <https://github.com/p-christ/Deep-Reinforcement-Learning-Algorithms-with-PyTorch/blob/master/agents/hierarchical_agents/DIAYN.py> * <https://github.com/Steven-Ho/diayn-sac> |
| GrBAL | * <https://github.com/iclavera/learning_to_adapt> | * <https://github.com/Knoxantropicen/model-based-meta-rl> |

# Problem dimensions:

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Task | | | | Data | | | Reward | | Environment | | | | | | Phases | |
| IDs/boundaries | | Source | Distribution | Distribution | Source | | Params | Transition dynamics model | | | observability | State & action spaces |
| Known / defined | discrete | given | i.i.d. | stationary | Trial-&-Error | Demos | Known /dense | Fixed | Fixed | Known | | Deterministic | Full obs. | discrete | 2 phases (fixed) | |
| Semi-known | Continuous | Partially given | predictable | Partial non-stationary | Semi-known / sparse | Changing | Changing | Unknown, unlearned | Unknown, learned | Stochastic | Part obs. | Continuous  (or Mix) | Lifelong, 2 phases (interleaving) | Lifelong, online |
| unknown | curriculum | Full non-stationary | Unknown / non-existent or very sparse |
| Not given | adversarial | Mix | |
| OOD |

|  |  |
| --- | --- |
|  | Train |
|  | Test |
|  | Both |

# Desiderata / Objectives:

|  |  |
| --- | --- |
| Performance (jump-start and asymptotic) | Effective/structured exploration |
| Consistency | Computational, memory and time resources |
| Stability &/or Convergence  (related: Reliability / reproducibility) | Safety |
| Data/sample efficiency | Uncertainty awareness |
| Expressivity | Feasibility / Realizability |
| Robustness | Privacy |
| Learning/Adaptation speed and ability | Interpretability / Explainability |
| Transferability / Portability & Generalization | Complexity |