Evolution meets reinforcement learning

Matyáš Lorenc

Faculty of Mathematics and Physics, Charles University

8th April 2025

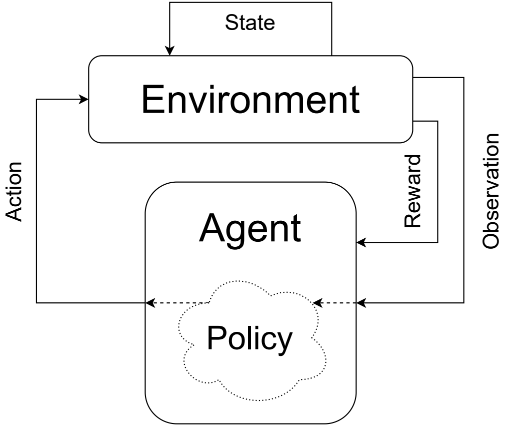
1. Prerequisites

2. Evolutionary algorithms in reinforcement learning

3. Various hybrid approaches for reinforcement learning

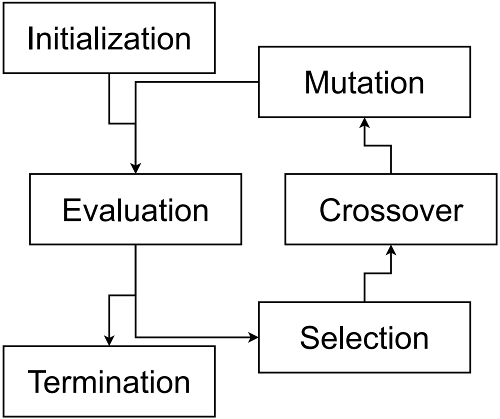
4. Evolutionary reinforcement learning and its derivatives

1. Prerequisites
2. Evolutionary algorithms in reinforcement learning
3. Various hybrid approaches for reinforcement learning
4. Evolutionary reinforcement learning and its derivatives

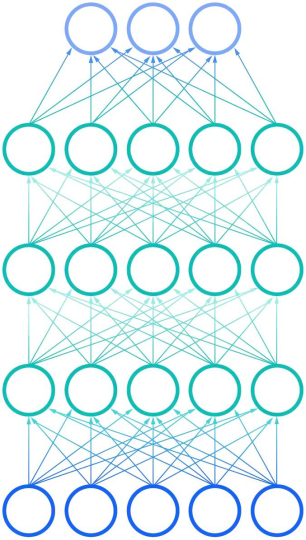


- Deep Q-learning
- Direct policy learning
  - Monte Carlo (TRPO, PPO)
  - Actor-Critic (TD3, SAC)
- On-policy / off-policy

- Genetic algorithms
- Evolution strategies
  - Classical
  - Distributional
- Genetic programming



1. Prerequisites
2. Evolutionary algorithms in reinforcement learning
3. Various hybrid approaches for reinforcement learning
4. Evolutionary reinforcement learning and its derivatives



- NNs = parametrized functions
  - GAs or ESs → weights
  - GAs or GP → structures
- GP → interpretable and / or compact policies
- Novelty search, Quality-diversity

Gradients

Evolution

Gradients

+ Faster convergence

Evolution

Gradients	**Evolution**
+ Faster convergence	
+ Better sample efficiency	

Gradients

+ Faster convergence
+ Better sample efficiency
+ The TD methods can handle even infinite episodes

Evolution

Gradients

+ Faster convergence
+ Better sample efficiency
+ The TD methods can handle even infinite episodes

Evolution

+ Strong out-of-the-box exploration (+ powerful exploration techniques)

Gradients

+ Faster convergence
+ Better sample efficiency
+ The TD methods can handle even infinite episodes

Evolution

+ Strong out-of-the-box exploration (+ powerful exploration techniques)
+ Easy (massive) parallelization

Gradients

+ Faster convergence
+ Better sample efficiency
+ The TD methods can handle even infinite episodes

Evolution

+ Strong out-of-the-box exploration (+ powerful exploration techniques)
+ Easy (massive) parallelization
+ No problem with sparse or long action-reward time horizont rewards

Gradients

+ Faster convergence
+ Better sample efficiency
+ The TD methods can handle even infinite episodes

Evolution

+ Strong out-of-the-box exploration (+ powerful exploration techniques)
+ Easy (massive) parallelization
+ No problem with sparse or long action-reward time horizont rewards
+ More robust convergence properties (fewer and less sensitive hyperparameters)

Gradients

+ Faster convergence
+ Better sample efficiency
+ The TD methods can handle even infinite episodes

Evolution

+ Strong out-of-the-box exploration (+ powerful exploration techniques)
+ Easy (massive) parallelization
+ No problem with sparse or long action-reward time horizont rewards
+ More robust convergence properties (fewer and less sensitive hyperparameters)
+ Multiobjective optimization $\rightarrow$ uniformly approximated Pareto front

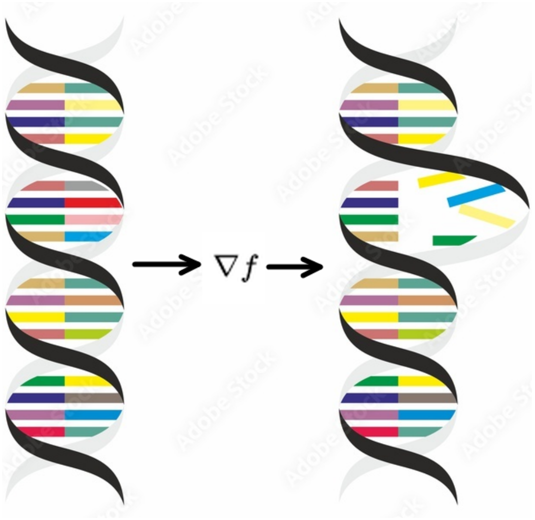
Gradients

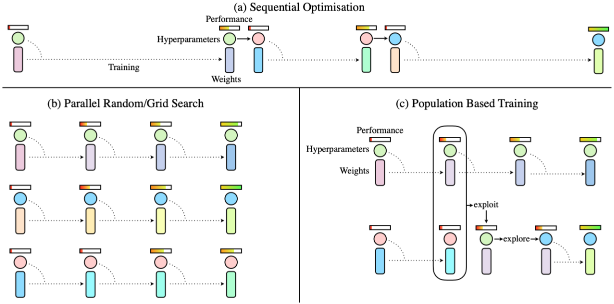
- + Faster convergence
- + Better sample efficiency
- + The TD methods can handle even infinite episodes

Evolution

- + Strong out-of-the-box exploration (+ powerful exploration techniques)
- + Easy (massive) parallelization
- + No problem with sparse or long action-reward time horizont rewards
- + More robust convergence properties (fewer and less sensitive hyperparameters)
- + Multiobjective optimization $\rightarrow$ uniformly approximated Pareto front
- + Multiagent RL $\rightarrow$ no need to have problem formulated as MDP

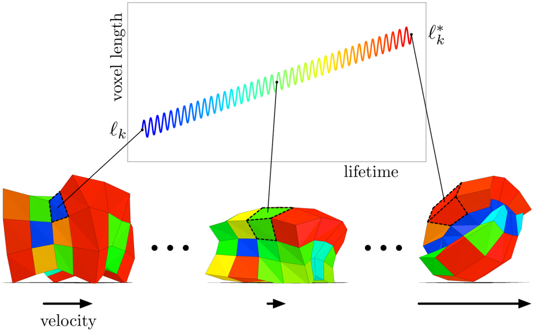
- Gradient mutation
- Sequential combination (for example, in multiobjective RL)
- Utilize principles from some gradient algorithms in evolutionary ones (e.g., trust regions, natural gradients, etc.)





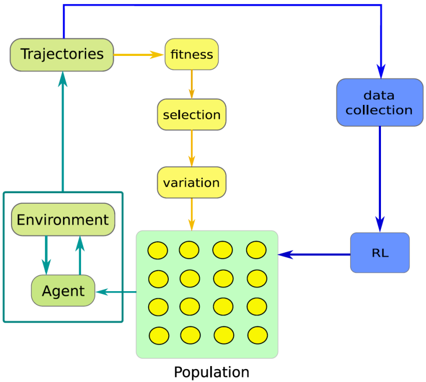
- Hyperparameter optimization
- Population-based training
- Evolving partial policies (GP + NNs to default to for non-specified actions)

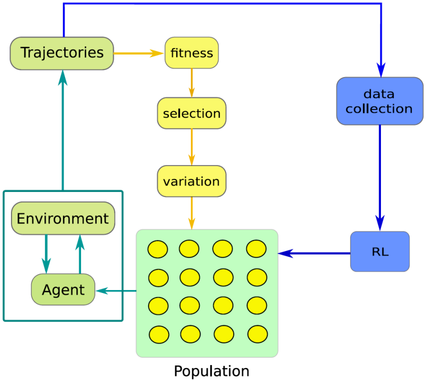
- Agent morphology
- Parameters of loss function for policy gradients (Meta-RL)
- Reward functions (Meta-RL)
- Critics

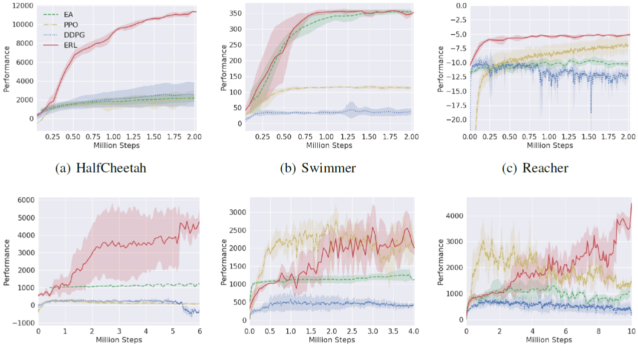


In RL, an action might be needed for:

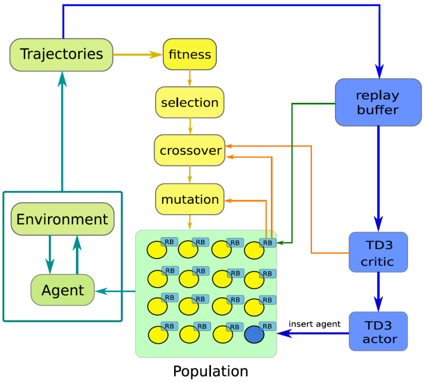
- Each step in the environment
- Critic update (estimating TD error)
- Policy update (imitation learning)

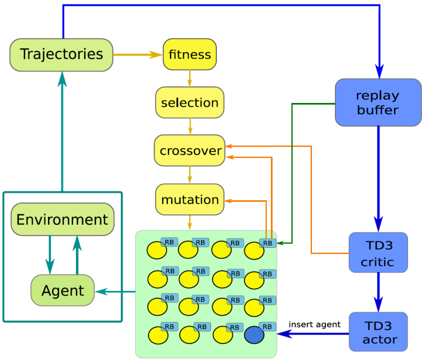

Trajectories
fitness
selection
variation
data collection
Environment
Agent
RL
Population

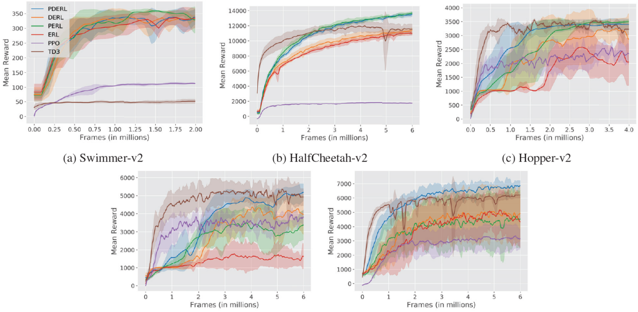




(a) HalfCheetah (b) Swimmer (c) Reacher
(d) Ant (e) Hopper (f) Walker2D


Trajectories
fitness
selection
crossover
mutation
replay buffer
Environment
Agent
TD3 critic
TD3 actor
insert agent
RB
Population

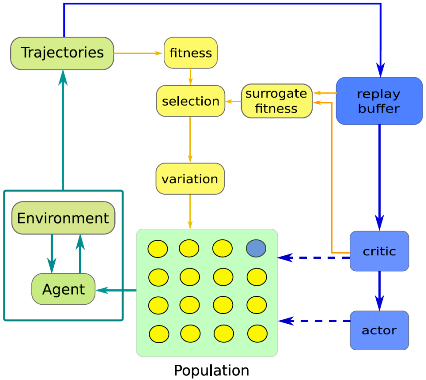


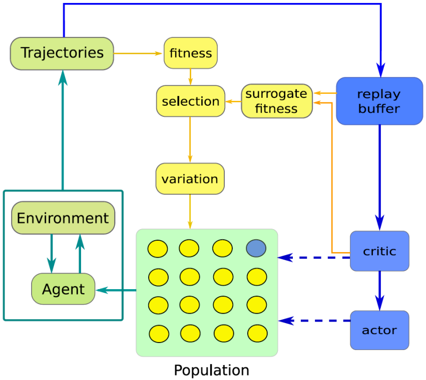


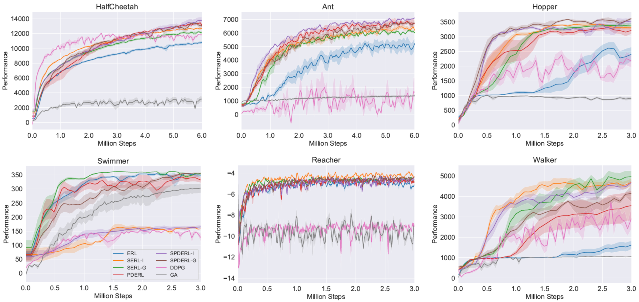
(a) Swimmer-v2 (b) HalfCheetah-v2 (c) Hopper-v2

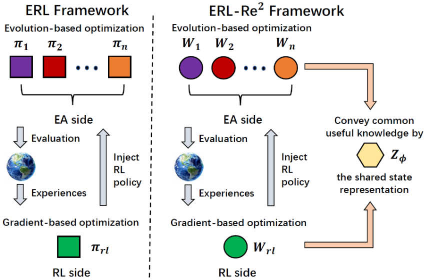
(d) Walker2d-v2 (e) Ant-v2

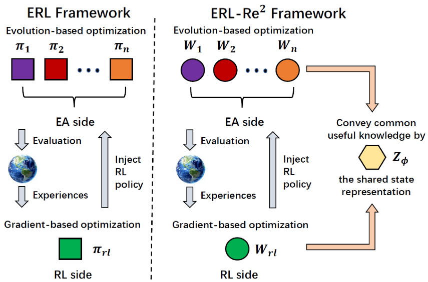
Let's crank up the speed! – SERL, SPDERL

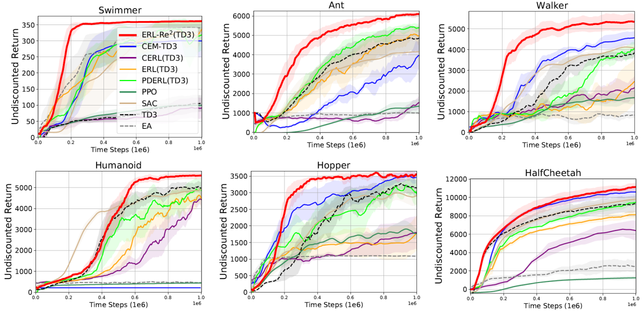



Trajectories
fitness
selection
surrogate fitness
replay buffer
variation
Environment
Agent
critic
actor
Population

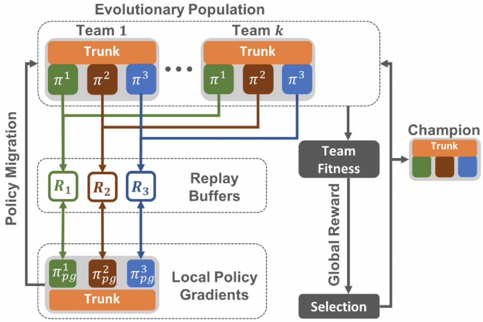

HalfCheetah
Ant
Hopper
Swimmer
Reacher
Walker
Performance
Million Steps
ERL
SERL-I
SERL-G
PDERL
SPDERL-I
SPDERL-G
DDPG
GA


ERL Framework
Evolution-based optimization
π_1 π_2 π_n
EA side
Evaluation
Experiences
Inject RL policy
Gradient-based optimization
π_{rl}
RL side
ERL-Re^2 Framework
Evolution-based optimization
W_1 W_2 W_n
EA side
Evaluation
Experiences
Inject RL policy
Gradient-based optimization
W_{rl}
RL side
Convey common useful knowledge by
Z_ϕ
the shared state representation


ERL Framework
Evolution-based optimization
EA side
Evaluation
Experiences
Inject RL policy
Gradient-based optimization
RL side
ERL-Re^2 Framework
Evolution-based optimization
EA side
Evaluation
Experiences
Inject RL policy
Gradient-based optimization
RL side
Convey common useful knowledge by
the shared state representation


Swimmer
Ant
Walker
Humanoid
Hopper
HalfCheetah
Undiscounted Return
Time Steps (1e6)
ERL-Re^2(TD3)
CEM-TD3
CERL(TD3)
ERL(TD3)
PDERL(TD3)
PPO
SAC
TD3
EA

How best to combine known components and approaches?

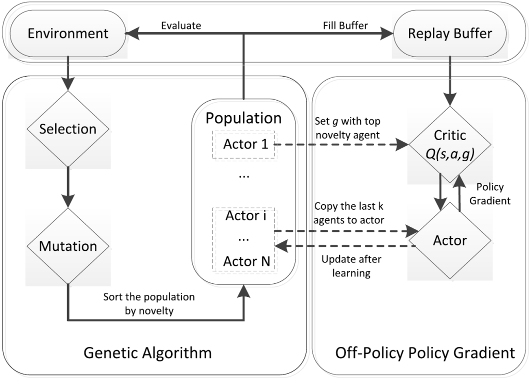


MERL Basic Strategy.

Two levels of a cooperative multiagent RL:

- Single agent
  - Gradient RL
  - Basic skills
- Team
  - Evolutionary RL
  - The overall task

How to utilize gradients to improve novelty?
Generally, we still don't know. But...



Quo vadis, ERL?