Derivative-Free Optimization

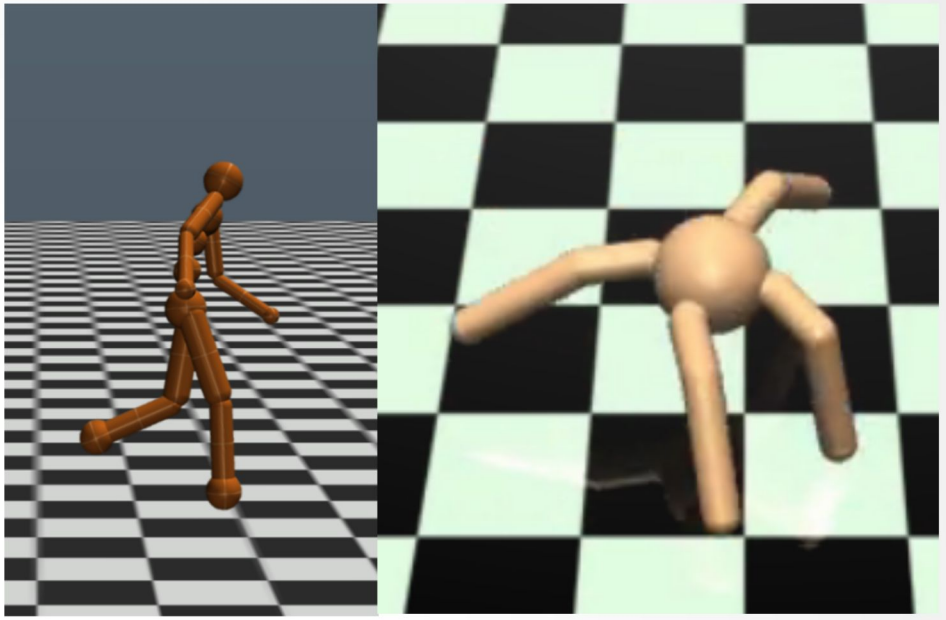
CMA-ES
Covariance Matrix Adaptation
(Evolution Strategy)

Cross-Entropy Method

Genetic Algorithms

Reinforcement Learning

Atari



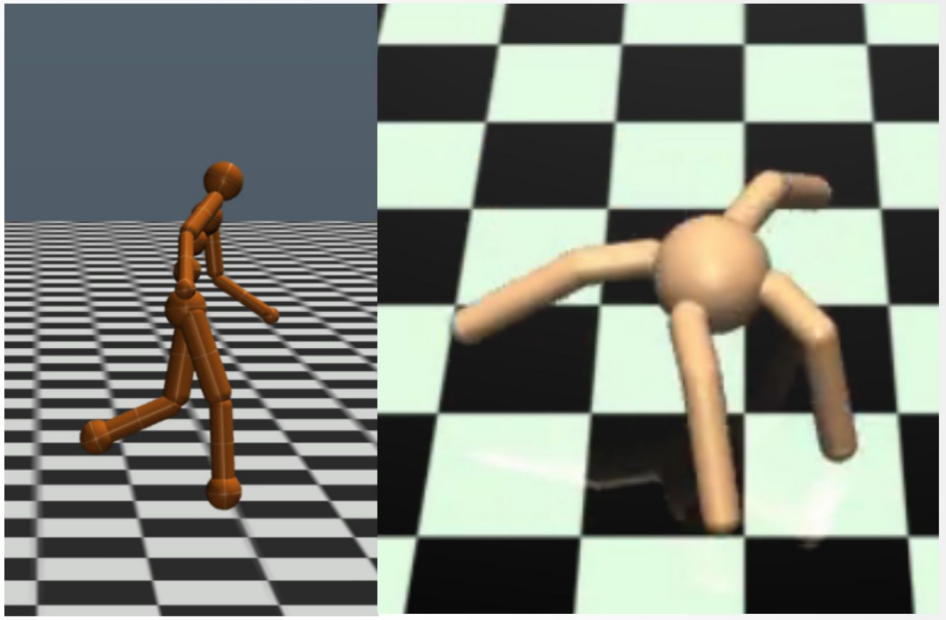
MuJoCo

- Methods: DQN, A3C, TRPO, PPO, ...

Sources: https://eng.uber.com/deep-neuroevolution/ and https://gym.openai.com/envs/

Reinforcement Learning

Atari



MuJoCo

- Methods: DQN, A3C, TRPO, PPO, ...
- ...or a simple GA.

Sources: https://eng.uber.com/deep-neuroevolution/ and https://gym.openai.com/envs/

Simple GA

1. Initialize 1’000 NNs ~4M weights (Atari)
    2. Evaluate
    3. Select best 2% as parents
    4. Generate 1’000 offspring
        - Copy a parent
        - Add noise ~ $N(0, \sigma^2)$

Simplified; see https://arxiv.org/abs/1712.06567 (Such et al., 2018)

Evolution Strategy

Update distribution (not individuals)

1. Generate Random Samples
   - e.g. $\theta_i \sim N(\boldsymbol{\mu}, \boldsymbol{\Sigma})$
2. Evaluate
3. Update Random Mechanism
   - e.g. $\boldsymbol{\mu}$ and $\boldsymbol{\Sigma}$ of best 25%

Intro: http://blog.otoro.net/2017/10/29/visual-evolution-strategies/

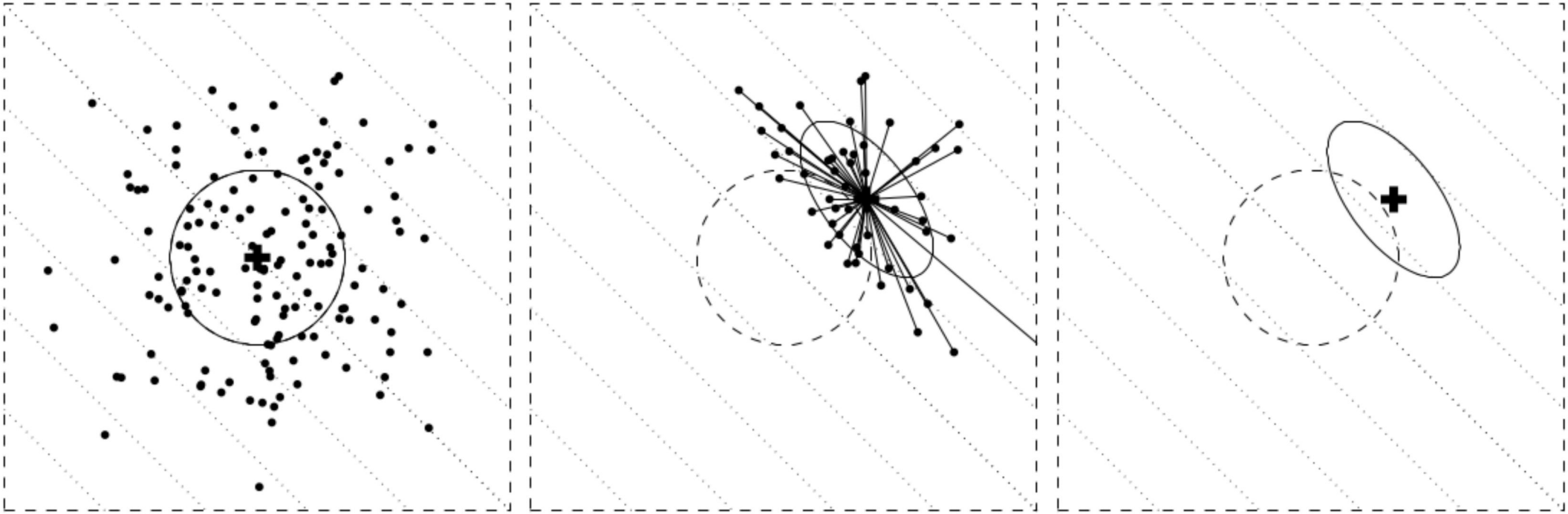


Image: https://arxiv.org/abs/1604.00772 (Hansen, 2016) CMA-ES tutorial, Rank-μ-Update

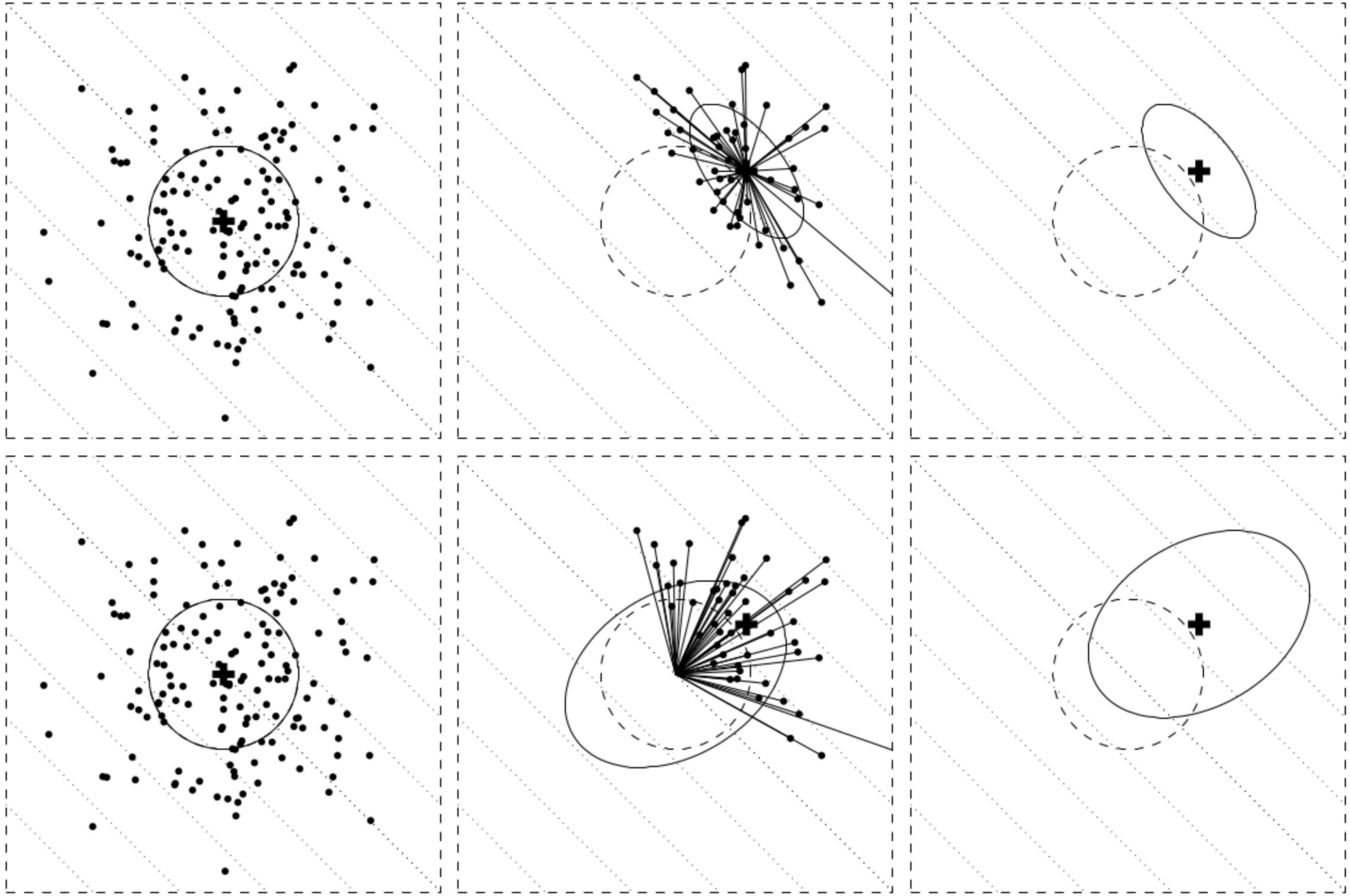


Image: https://arxiv.org/abs/1604.00772 (Hansen, 2016) CMA-ES tutorial, Rank-μ-Update

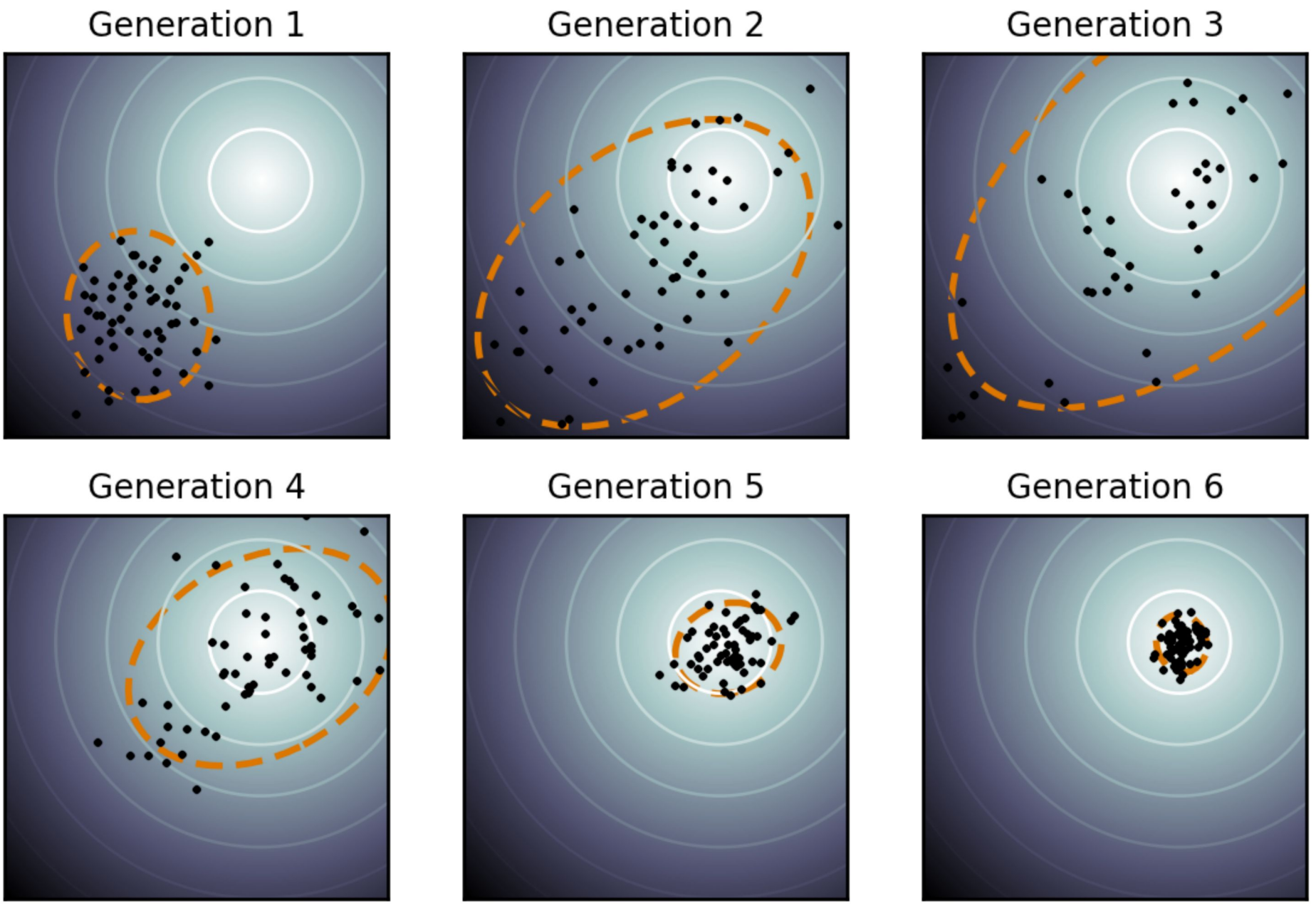


Image: https://en.wikipedia.org/wiki/CMA-ES (by Sentewolf)

CMA-ES

"extremely successful solving problems in low to medium dimension"

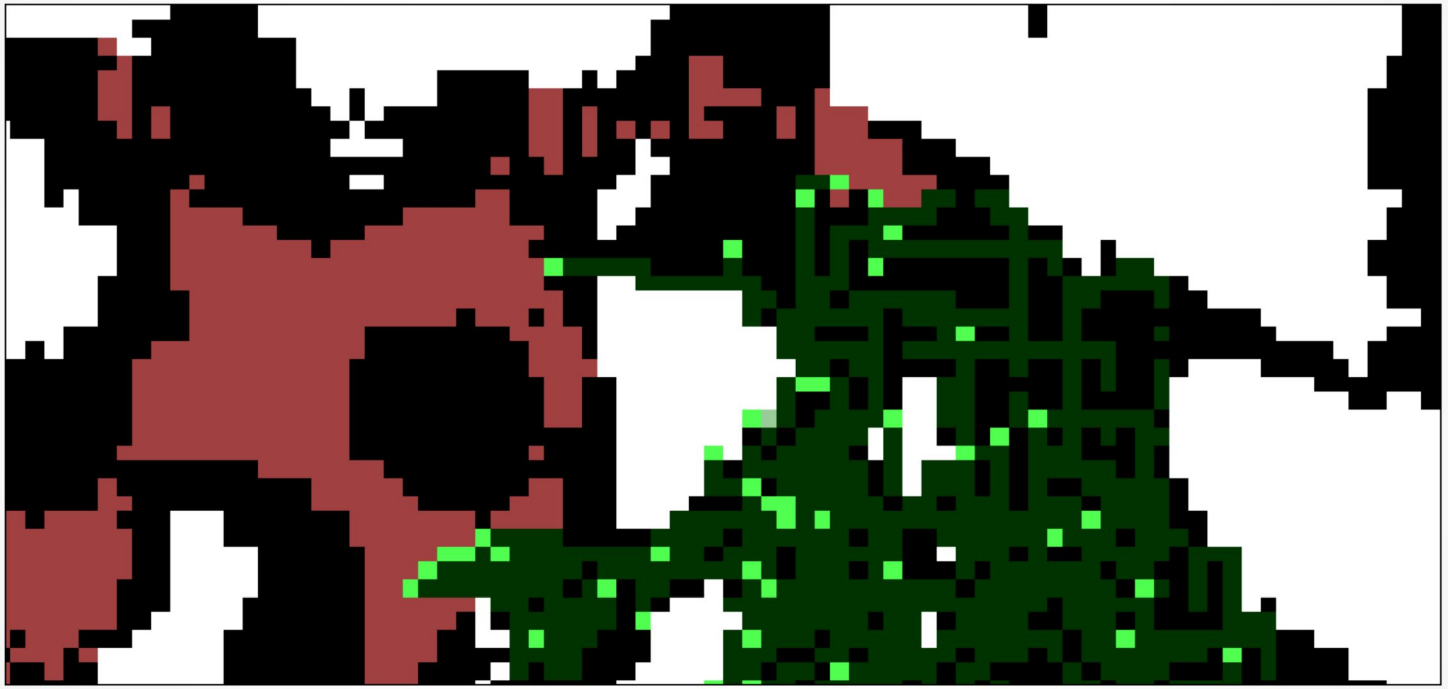
(Salimans, 2017)

- Almost parameter-free (for user)
- Fast on black-box benchmarks
  - when no (good) gradient is available

Quote: https://arxiv.org/abs/1703.03864 (Salimans et al., 2017)

CMA-ES: Experiment

- Small NN (~600 weights), pycma
- Cooperative swarm: eat red food



- CMA-ES “just worked”; consistent results
- Tuning: model size, recurrent state

Summary

CMA-ES

- Easy to use
- Continous only
- Small/medium size (< 5’000 params)

Alternatives

- Cross-Entropy Method
- Genetic Algorithms
- Anything more specific (e.g. using analytical gradient)

References!

- **A Visual Guide to Evolution Strategies**
  http://blog.otoro.net/2017/10/29/visual-evolution-strategies/
- **Evolution Strategies as a Scalable Alternative to Reinforcement Learning**
  https://blog.openai.com/evolution-strategies/
  https://arxiv.org/abs/1703.03864 (Salimans et al., 2017)
- **Deep Neuroevolution**
  https://eng.uber.com/deep-neuroevolution/
  https://arxiv.org/abs/1712.06567 (Such et al., 2018)
- **Trust Region Policy Optimization** (CMA-ES/CEM baseline)
  https://arxiv.org/abs/1502.05477 (Schulman et al., 2015)