Estimation of Distribution Algorithms

•••

Jan Waltl

Overview

- Distribution of what exactly?
- Overview of developed and used EDAs.
- What do EDAs have in common and how they differ from Genetic Algorithms.
- Examples of (simple) discrete and continuous problems solved by EDAs.
- Recent developments.

Quick Recap of Genetic Algorithms

The Task

...

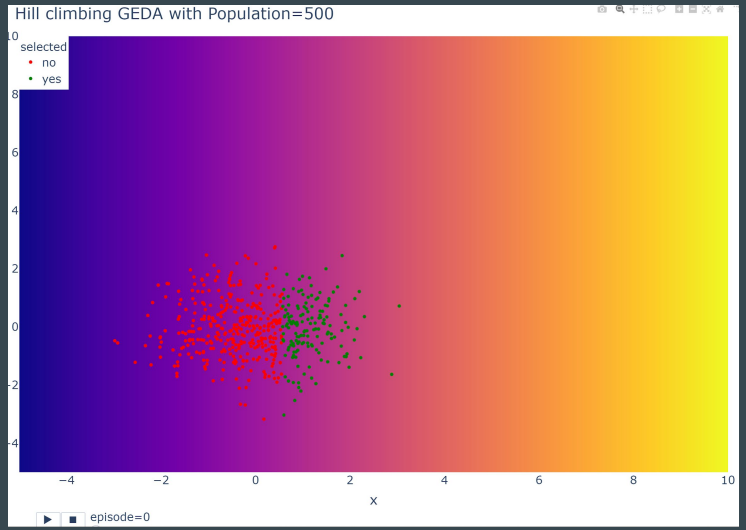
Given a search space and a fitness function, find the point of global maximum.

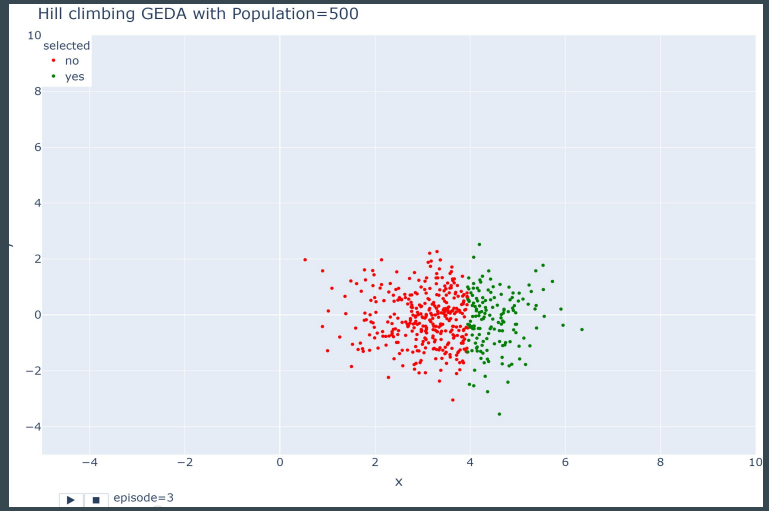
Genetic Algorithm

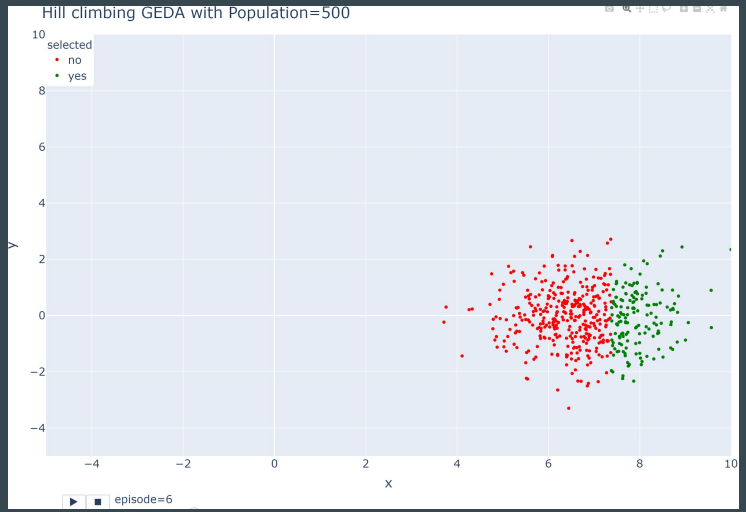
- Evolve population of individuals = elements of search space.
- Generate a new population by:
  - Selecting promising individuals to breed.
  - Creating children by recombination of parents.
  - Mutating children.
- Children hopefully inherit good traits from parents.
- Mutation serves as exploration.
- The hope is that by repeating this process, the evolution will guide the search to better solutions and weed out the bad ones.

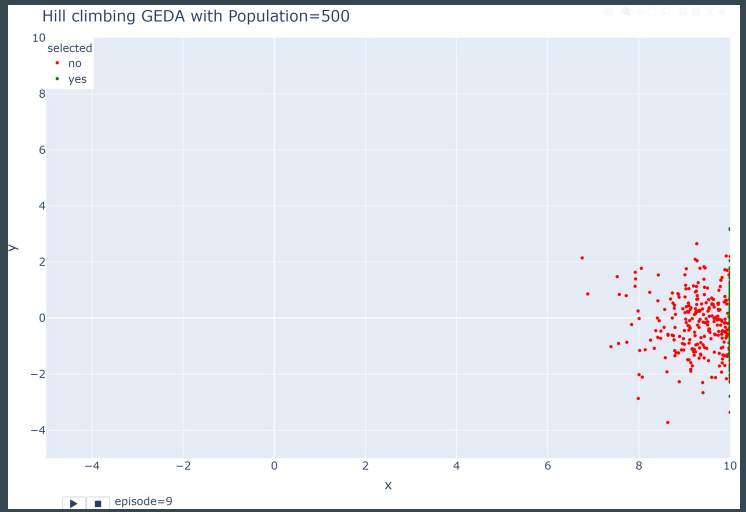
Estimation of Distribution Algorithm

- Population* of individuals.
- Generate a new generation by:
    - Selecting promising individuals based on fitness.
    - Creating a probabilistic model over the search space based on the selected individuals.
    - Using this model to sample a new generation.
- The hope here is that the model can extract the information needed to create a diverse population similar to selected individuals.
- The goal is to create a model generating globally optimal solutions.


Hill climbing GEDA with Population=500
selected
no
yes
x
episode=0

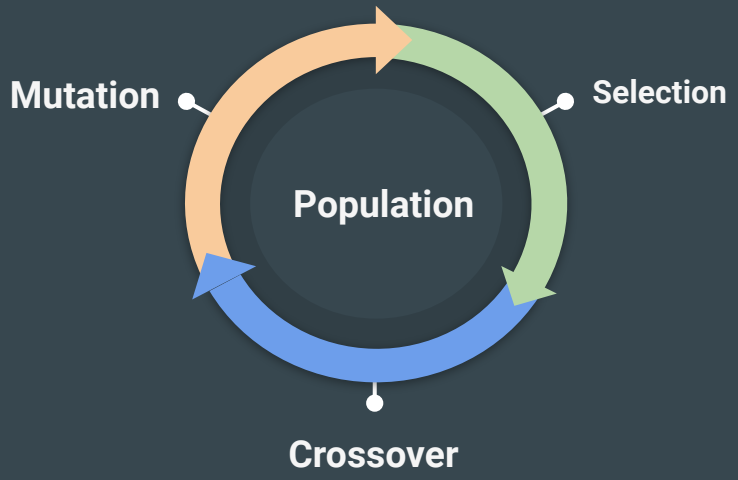

Hill climbing GEDA with Population=500
selected
no
yes
x
episode=3


Hill climbing GEDA with Population=500
selected
no
yes
x
episode=6


Hill climbing GEDA with Population=500
selected
no
yes
x
episode=9

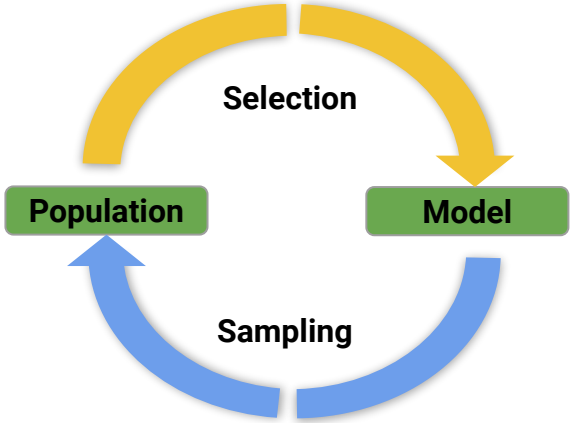
GA

- Population based
- Mutation provides exploration
- Crossover inherits good traits
- Selection for breeding guides the search



EDA

- Model based, still uses population
- Random sampling provides exploration
- Model captures good traits
- Selection for modeling guides the search



Matching Binary Strings

Matching Binary Strings

Task:

- Search space: $S = \{0,1\}^d$
- Binary string $s \in S$
- Fitness function: $f(x) = - \#\ (s_i \neq x_i)$ for $i \in [1,d]$

Goal:

- Find global optimum $x=s$ with $f(x)=0$

Univariate Marginal Distribution Algorithm

Selected population K:

- Top % in the current population (10%-50%).

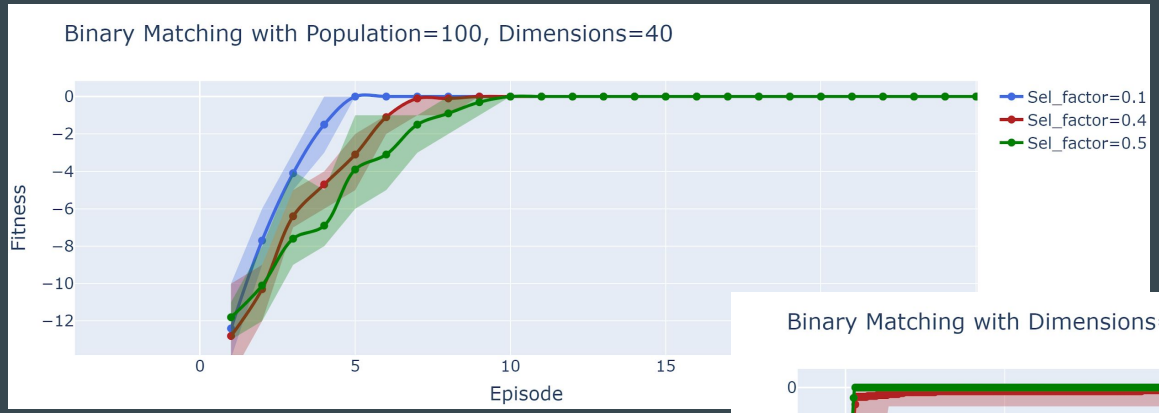
Model:

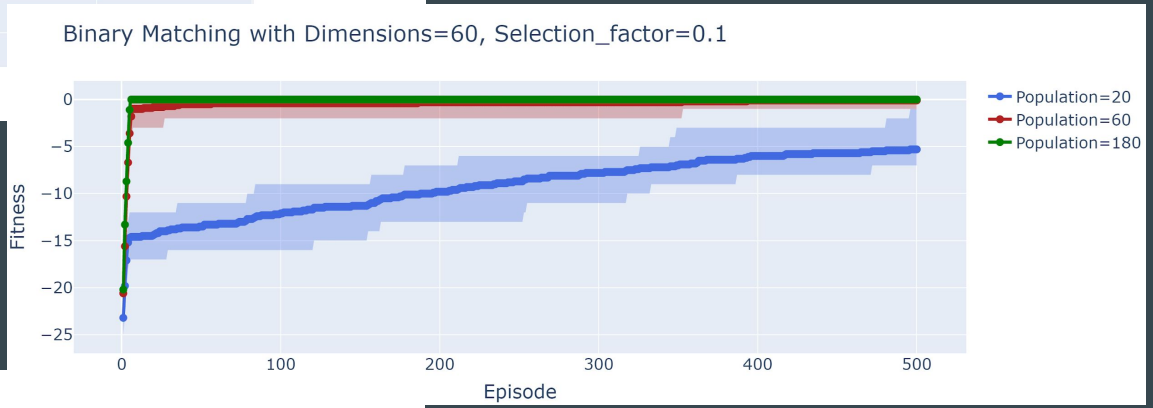
- Model $p(x_i = 1)$ for each position in the individual separately, $p(x_i = 0) = 1 - p(x_i = 1)$.
- An individual $x \in S$ is assigned probability $p(x) = \prod p(x_i)$.
- Estimate the model's parameters as $p(x_i = 1) = \#(x_i = 1) / |K|$ for $x \in K$.

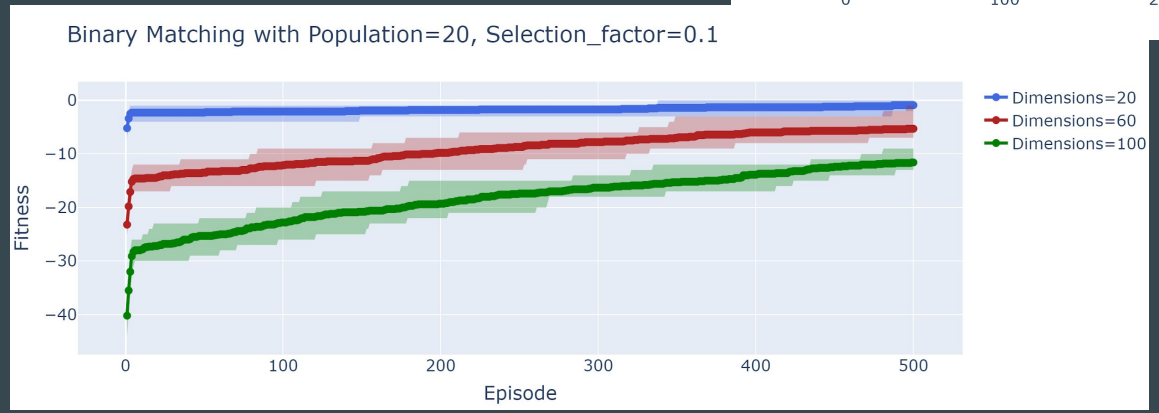
$$K = \left\{ \begin{array}{l} 1110 \\ 0100 \\ 0100 \\ 1100 \\ 0000 \end{array} \right\} \quad \begin{array}{l} p(x_1=1) = 2/5 \\ p(x_2=1) = 4/5 \\ p(x_3=1) = 1/5 \\ p(x_4=1) = 0 + \varepsilon \end{array}$$

Retain Population Diversity - Premature Convergence

[Mühlenbein, Heinz.]


Binary Matching with Population=100, Dimensions=40
Fitness
Episode
Sel_factor=0.1
Sel_factor=0.4
Sel_factor=0.5


Binary Matching with Dimensions=60, Selection_factor=0.1
Fitness
Episode
Population=20
Population=60
Population=180


Binary Matching with Population=20, Selection_factor=0.1
Fitness
Episode
Dimensions=20
Dimensions=60
Dimensions=100

Binary Traps

Binary Traps

Assignment:

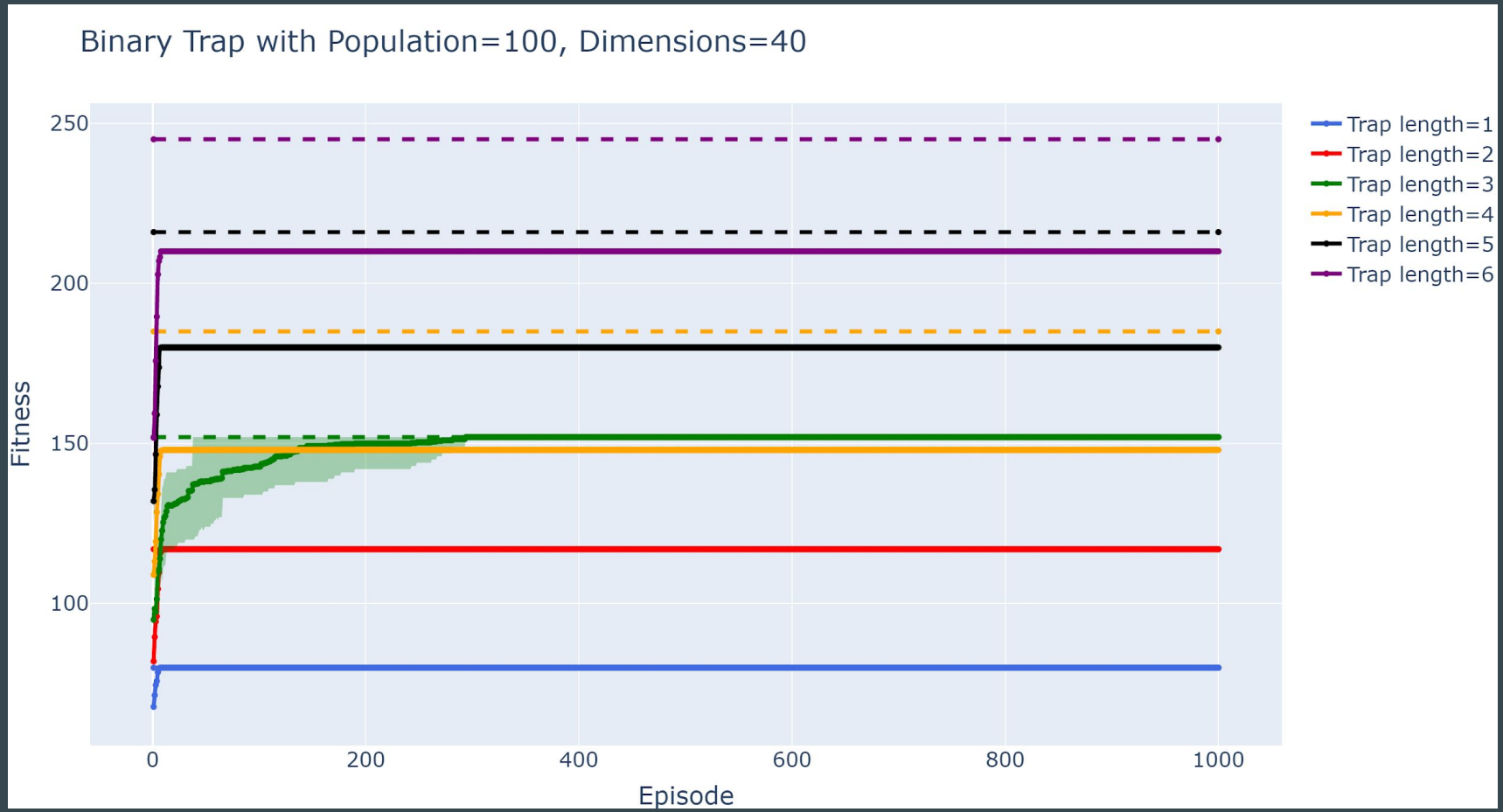
- Search space: $S = \{0,1\}^d$
- Binary string $s \in S$
- Trap length L
- Fitness function: $f(x) = \sum_i \text{score}(x,i)$

$$\text{score}(x,i) = \begin{cases} L + 1 & \text{if } xi...xi+L-1 = 1...1 \\ L - \sum_{k=i}^{i+L-1} x_k & \text{otherwise} \end{cases}$$

Goal:

- Global optimum x=1...1 with $f(x) = (d+1-L)(L+1)$,
- $f(0...0)=(d+1-L)(L+0)$.

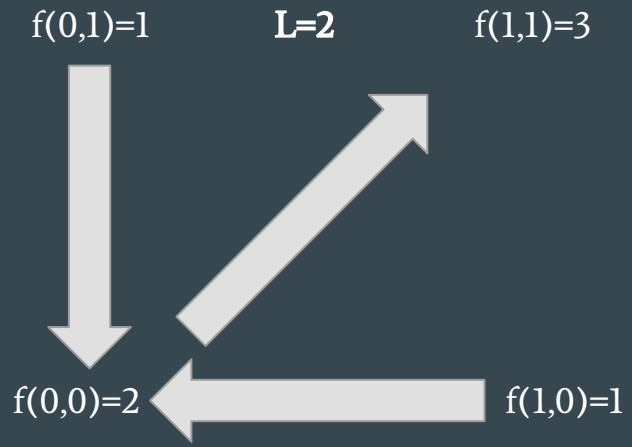
[Hauschild,Pelikan]


Binary Trap with Population=100, Dimensions=40
Fitness
Episode
250
200
150
100
0
200
400
600
800
1000
Trap length=1
Trap length=2
Trap length=3
Trap length=4
Trap length=5
Trap length=6

Gradient vs individual coordinates and epsilon

- For an individual variable it is always more profitable to go to zero
- Only if the whole group decides to switch to ones, it can profit
- Our model cannot capture this relationship.
- Holland's schema theorem

"Schemata with above-average fitness (especially short, low order schemata), increase their frequency in the population each generation at an exponential rate when rare."-[Altenberg,Lee]



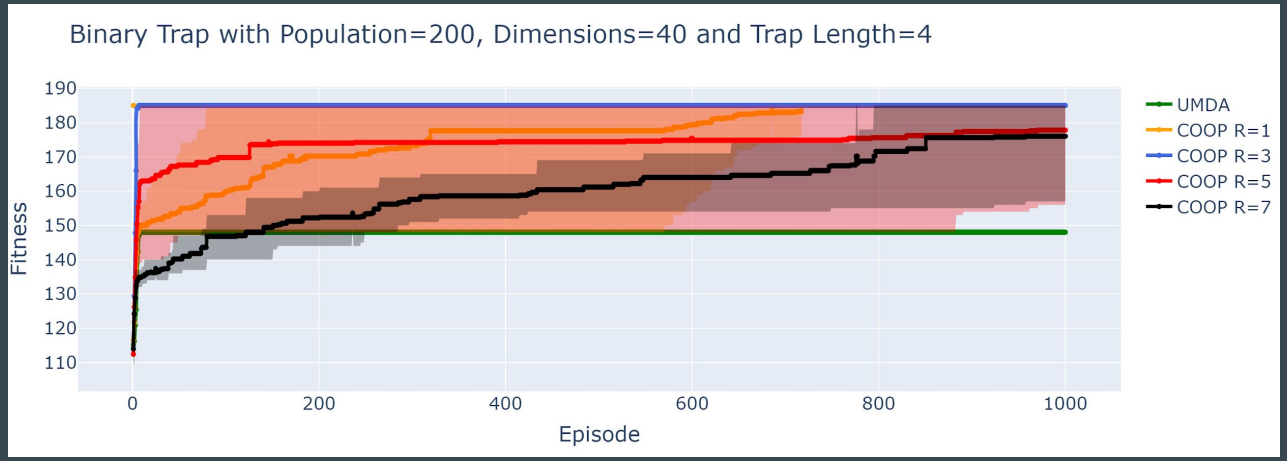
Cooperation Model

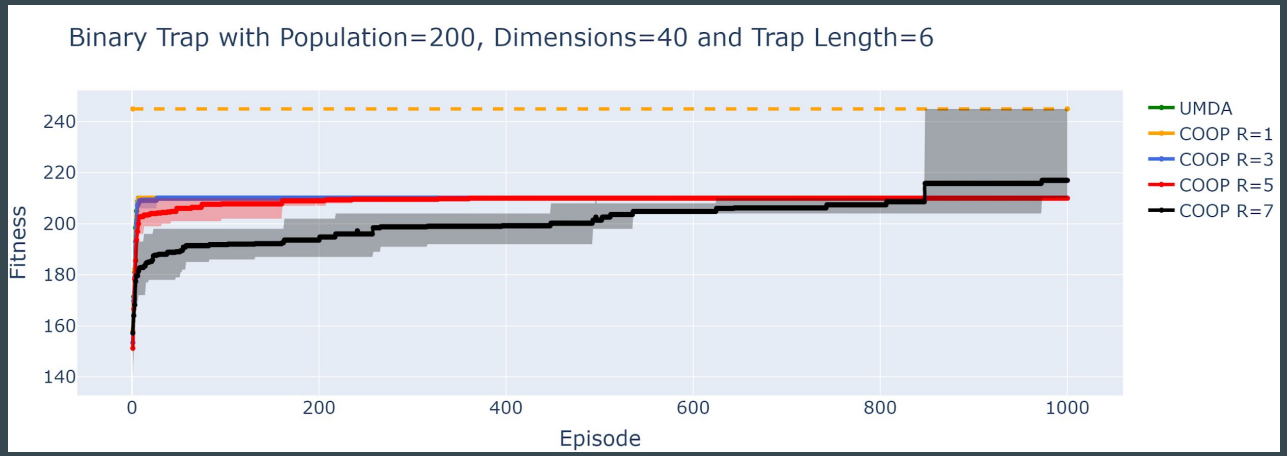
Selected population K:

- Top % in the current population

Model:

- Estimate $p(x_i = 1 \mid x_{i-1}, x_{i-2}, \ldots, x_{i-R})$ - probability depends on the previous R variables
- It is not an exact model!
- Given values $x_{i-1}, x_{i-2}, x_{i-R}$ we can sample x_i - individual can be sampled from left to right.
- Estimate $p(x_i = 1 \mid x_{i-1} = y_1, \ldots, x_{i-R} = y_R)$ as $\#(x_i = 1, x_{i-1} = y_1, \ldots, x_{i-R} = y_R) / \#(x_{i-1} = y_1, \ldots, x_{i-R} = y_R)$ for $x \in K$
- Or 0.5 if the denominator is zero.
- $+ \varepsilon$ if numerator is zero.


Binary Trap with Population=200, Dimensions=40 and Trap Length=4
Fitness
Episode
UMDA
COOP R=1
COOP R=3
COOP R=5
COOP R=7


Binary Trap with Population=200, Dimensions=40 and Trap Length=6
Fitness
Episode
UMDA
COOP R=1
COOP R=3
COOP R=5
COOP R=7

Continuous Space

Point Attraction

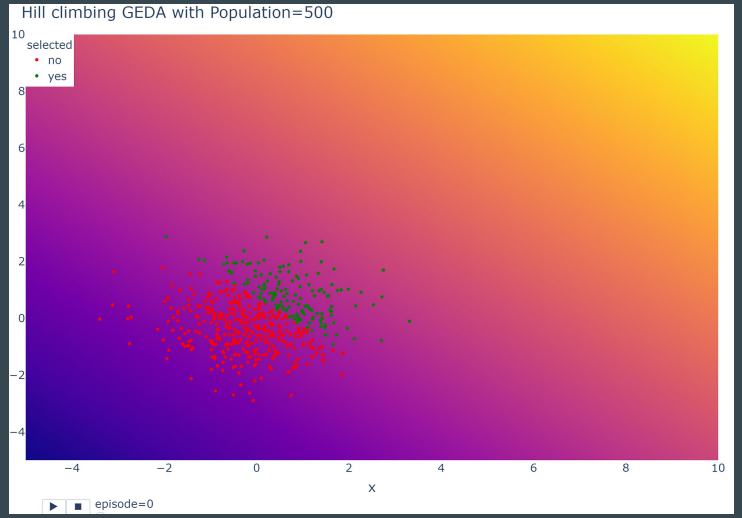
Point Attraction

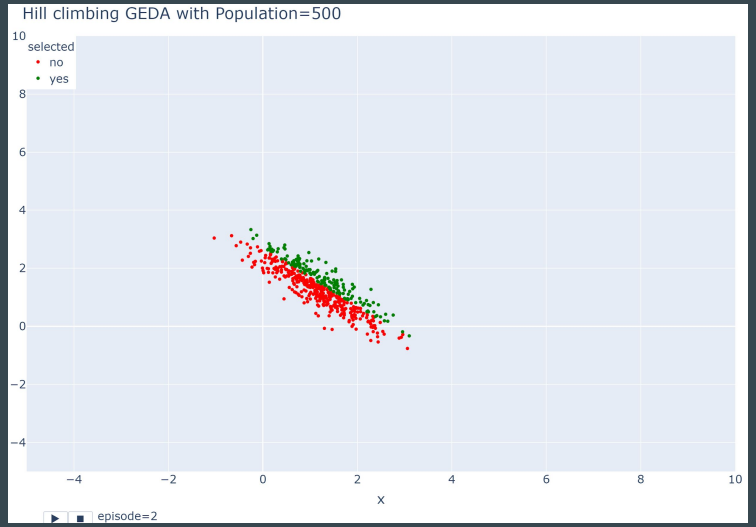
Assignment:

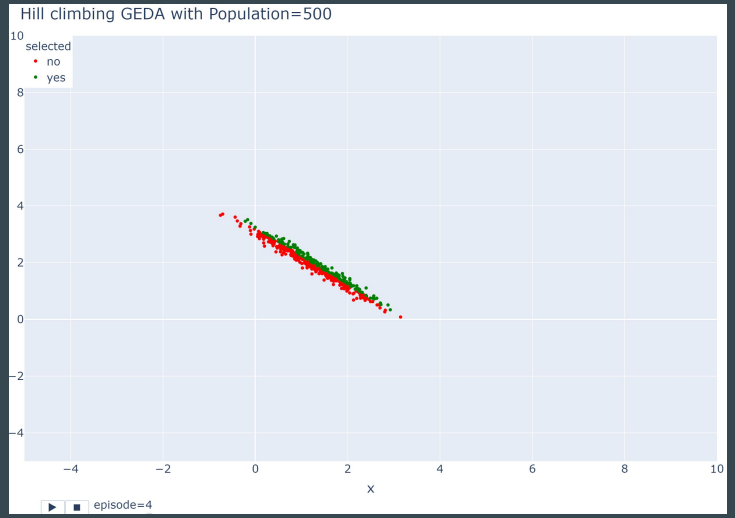
- Search space: $S \subseteq R^d$
- Point $s \in R^d$
- Fitness function: $f(x) = - \sum_i |s_i - x_i|$.

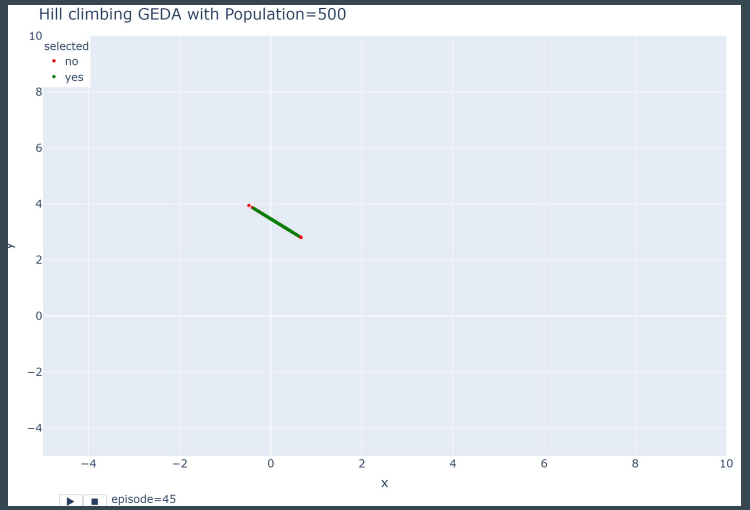
Goal:

- Global optimum x=s with f(x) = 0.









Premature Convergence

Gaussian Estimation of Distribution Algorithm (GEDA)

Selected population K:

- Top % in the current population.

Model:

- Model selected population using multivariate normal distribution
- Parameters to estimate: mean $\mu \in R^d$ and covariance matrix Σ
- $P(x) = ...$
- Parameters can be estimated by maximum likelihood estimation:

$$f_{\mathbf{X}}(x_1, \ldots, x_k) = \frac{\exp\left(-\frac{1}{2}(\mathbf{x}-\boldsymbol{\mu})^{\mathrm{T}}\boldsymbol{\Sigma}^{-1}(\mathbf{x}-\boldsymbol{\mu})\right)}{\sqrt{(2\pi)^k|\boldsymbol{\Sigma}|}}$$

en.wikipedia.org

$$\overline{\boldsymbol{\mu}}^{t+1} = \frac{1}{|\boldsymbol{S}^t|}\sum_{i=1}^{|\boldsymbol{S}^t|}\boldsymbol{S}_i^t$$

$$\overline{\boldsymbol{C}}^{t+1} = \frac{1}{|\boldsymbol{S}^t|}\sum_{i=1}^{|\boldsymbol{S}^t|}\left(\boldsymbol{S}_i^t - \overline{\boldsymbol{\mu}}^{t+1}\right)\left(\boldsymbol{S}_i^t - \overline{\boldsymbol{\mu}}^{t+1}\right)^T$$

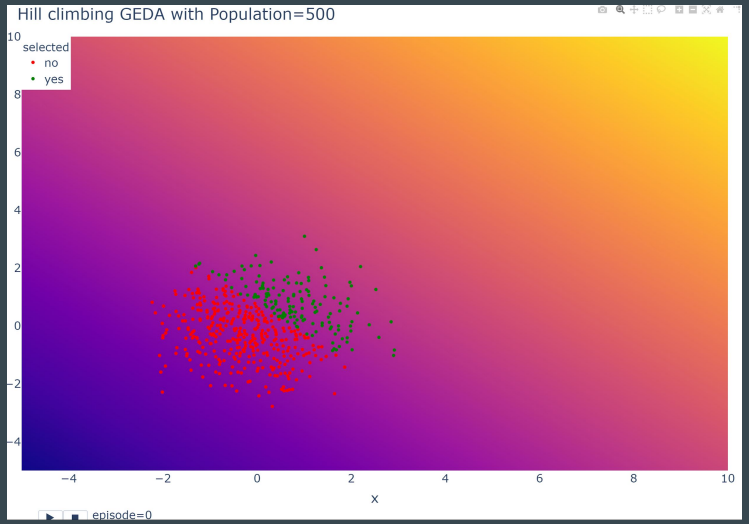
[Liang,Yongsheng] [Liang,Yongsheng]

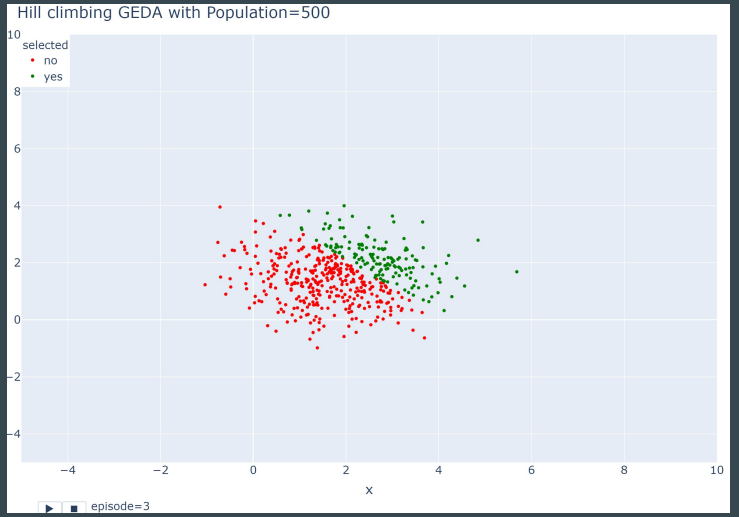
GEDA^2

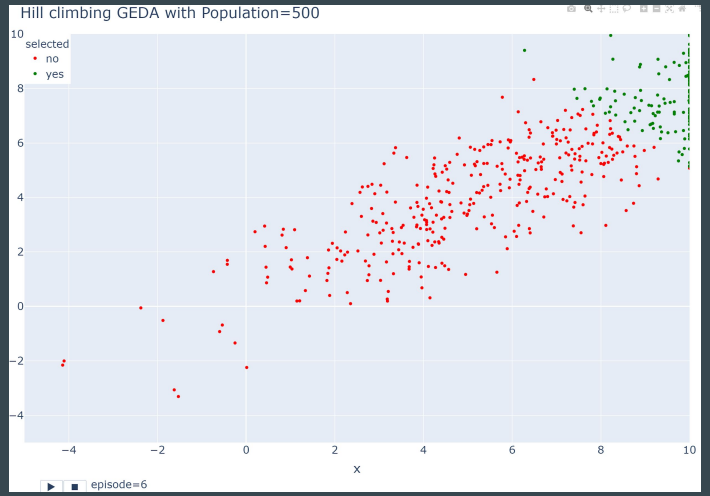
- Algorithm proposed by Lian et al., 2018.
- Not only that the variance vanishes, it it also orthogonal to gradient.
- Keep an archive of selected individual from previous generations.
- Estimate mean still only from the current population.
- Estimate variance from the current population and the archive.

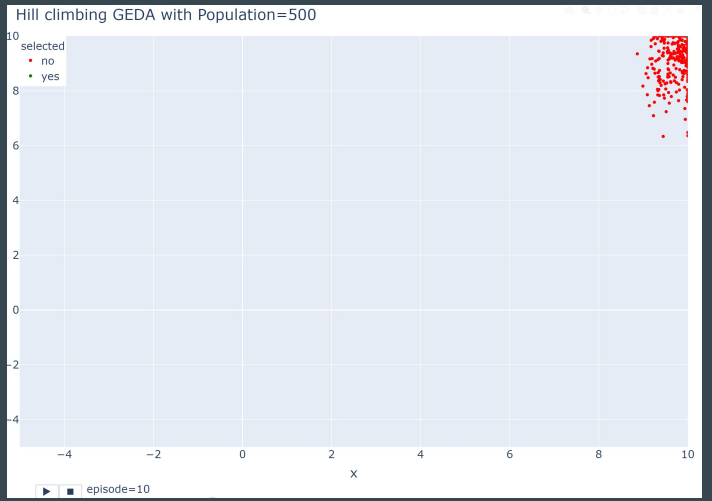
> *"Experimental results on two sets of benchmark functions demonstrate that the new developed archive-based covariance matrix estimation method is effective and EDA^2 is robust to its parameters and different problem dimensions. Compared with the traditional and six efficient EAs, EDA^2 exhibits the overall best performance."*-[Liang,Yongsheng]

[Liang,Yongsheng]


Hill climbing GEDA with Population=500
selected
no
yes
x
episode=0


Hill climbing GEDA with Population=500
selected
no
yes
x
episode=3


Hill climbing GEDA with Population=500
selected
no
yes
x
episode=6


Hill climbing GEDA with Population=500
selected
no
yes
x
episode=10

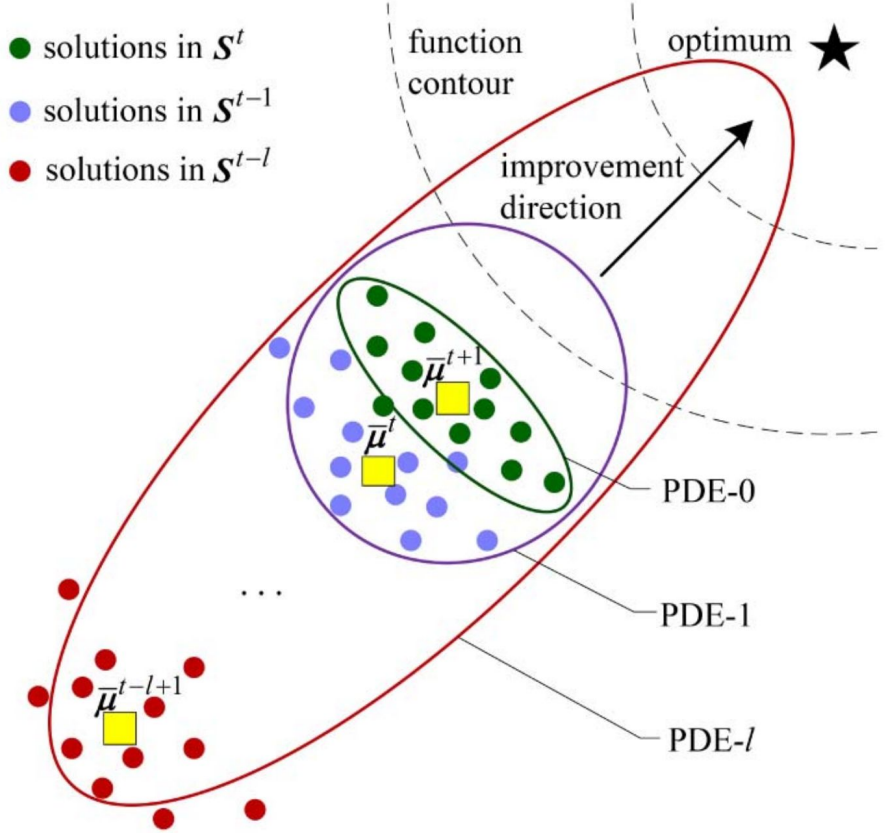
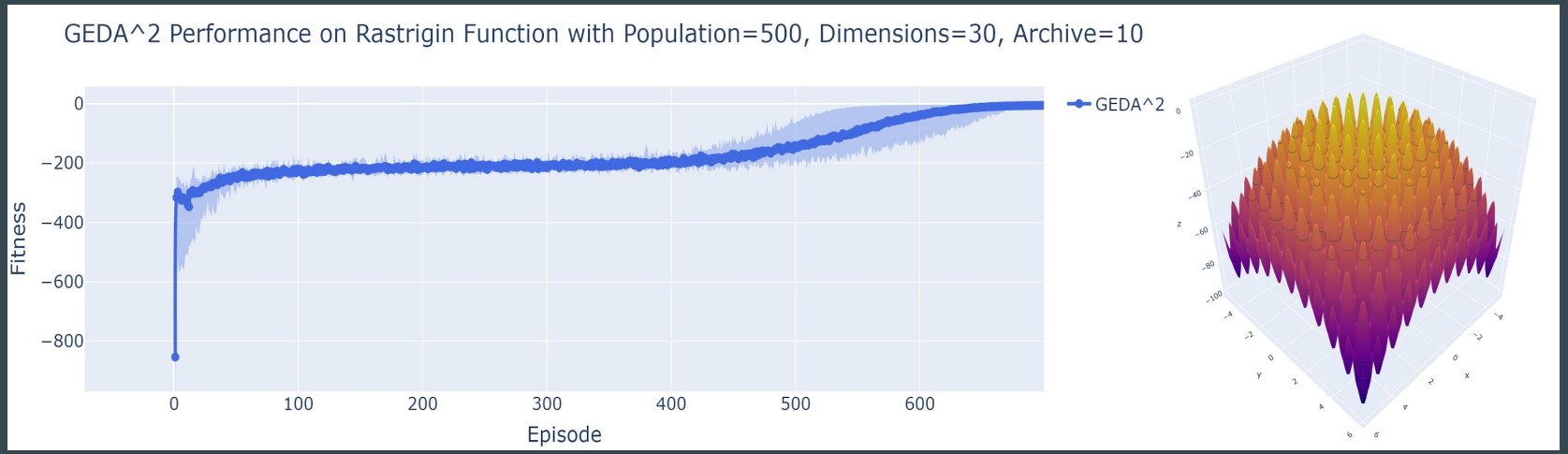
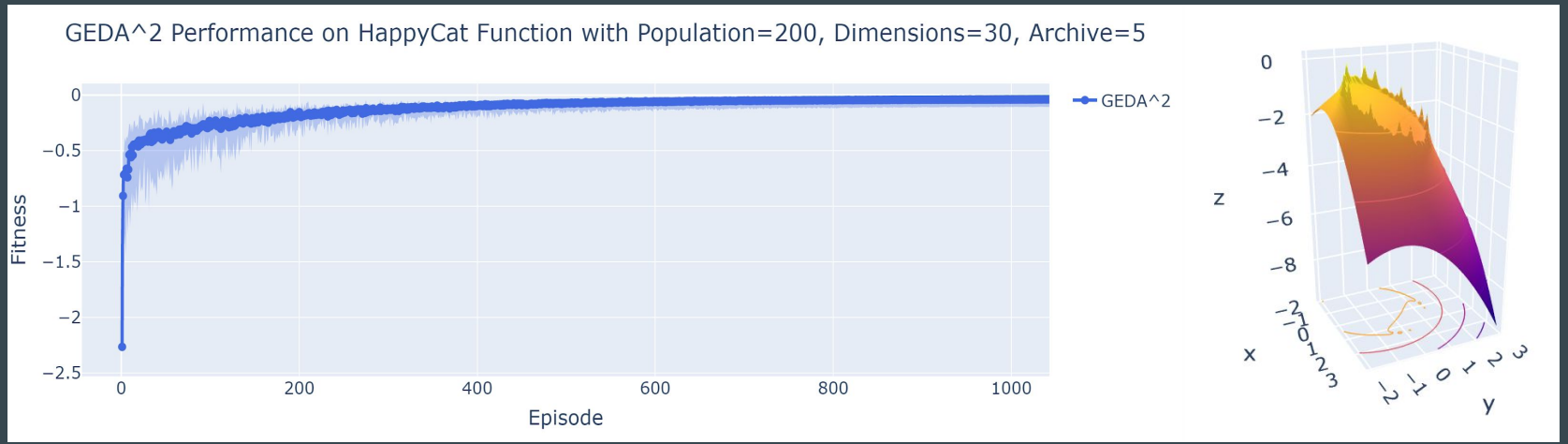


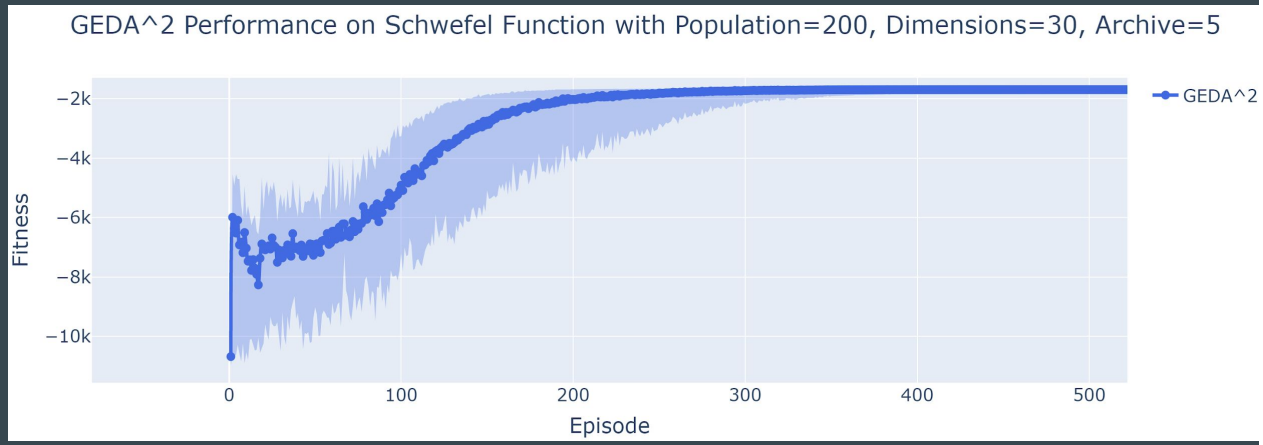
Fig. 2. Schematic of PDEs estimated with different archive lengths in EDA^2.

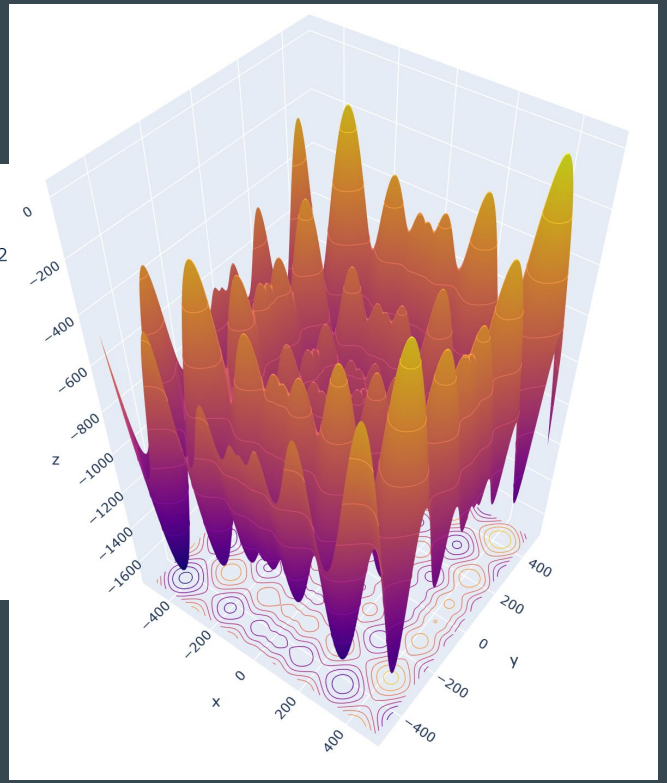
[Liang,Yongsheng]


GEDA^2 Performance on Rastrigin Function with Population=500, Dimensions=30, Archive=10
Fitness
Episode
GEDA^2
0
−200
−400
−600
−800
0
100
200
300
400
500
600


GEDA^2 Performance on HappyCat Function with Population=200, Dimensions=30, Archive=5
Fitness
Episode
GEDA^2
0
−0.5
−1
−1.5
−2
−2.5
0
200
400
600
800
1000

0
−2
−4
−6
−8
z
x
y





www.benchmarkfcns.xyz

Overview of other EDAs

- PBIL - Population Based Incremental Learning [Baluja, Schumeet]
  - Interpolate UMDA between generations.
- ECGA - Extended Compact Genetic Algorithm [Harik, Georges]
  - Group variables into clusters and computer joint probability.
  - Balance between exponential probability table size and dependencies.
- BOA - Bayesian Optimization algorithm[Pelikan, Goldberg]
  - Build (greedily) Bayesian Network over variables.
  - Optimize for some network score.
  - hBOA - Hierarchical decomposition.
- GEDA variants dealing with vanishing variance:
  - [Tamayo-Vera] - Thresholding convergence.
  - [Cai, Yunpeng] - Adaptive variance.

Challenges of EDAs

- Model choice
- Population size and curse of dimensionality
- Bigger population is not always better - GEDA^2
- Premature convergence
  - What happens when the population effectively contains only one individual?
  - How do GA deal with that?
- Constraints modeling - see [Ceberio] slides
- **(Almost) all papers I found deal only with suspiciously low dimensionality and benchmark functions.**

What About Using Generative Neural Networks as Models?

Variational Autoencoder with Population Queue

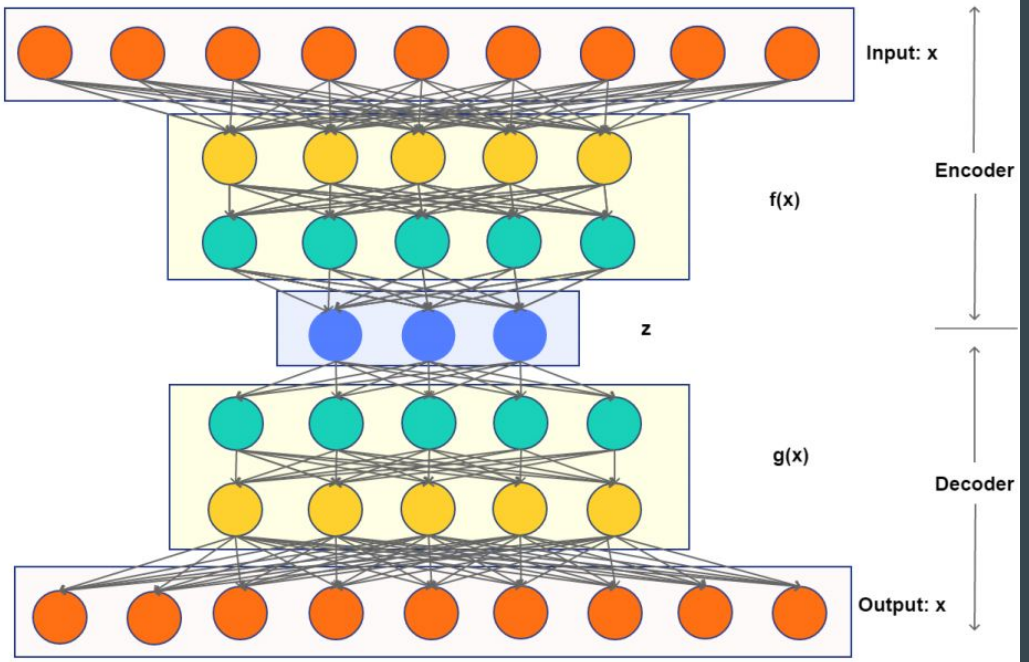


Fig. 2. Variational Autoencoder structure showing the parent vectors on top layer mapped to higher dimensional spaces in encoder f(x). The z layer provides the probability distribution sampling function which is sampled by the decoding layers g(x) to yield the output in real dimension at the bottom layer.

[Bhattacharjee,Sourodeep]

Variational Autoencoder with population Queue

- Proposed by Bhattacharjee et al., 2019
- Train network enough so it reasonably approximate selected individuals
- By not training on the old ones, network will forget them
- Population Queue=Archive
- The paper claims it outperforms other EDAs both in time and fitness evaluation
- I did not manage to get it past hill climbing problem and even that fails to converge.
    - Very thin line between underfitting and overfitting the network.
    - Failure to really converge to single best solution.
    - How to optimize the architecture if the network never stops learning
- **[Garciarena, Unai] -** VAE also predicts fitness, preceded VAE-Q

[Bhattacharjee, Sourodeep]

Generative Adversarial Networks

GANs

- One paper - Probst & Malte, 2016
- Worse and slower than other approaches:

  “GAN-EDA is not competitive, neither in the number of fitness evaluations required, nor in the computational effort. On the tested benchmark problems, it was unable to reliable find the respective global optima with reasonable population sizes. A reason for this bad performance could be the noisy training data”

- Authors, too, struggled with tuning their network and the proper amount of training.
- Since 2013 GANs have evolved
  - Training is more stable, e.g. Progressively grown GANs

[Probst,Malte]

References

- **[Mühlenbein, Heinz.] - Mühlenbein, Heinz. (1997). The Equation for Response to Selection and Its Use for Prediction. Evolutionary computation. 5. 303-46. 10.1162/evco.1997.5.3.303.**
- **[Hauschild,Pelikan]** - Hauschild, Mark and Martin Pelikan. "An introduction and survey of estimation of distribution algorithms." *Swarm Evol. Comput.* 1 (2011): 111-128.
- **[Altenberg,Lee] -** Altenberg, Lee. (2002). The Schema Theorem and Price's Theorem. Foundations of Genetic Algorithms. 3. 10.1016/B978-1-55860-356-1.50006-6.
- **[Liang,Yongsheng]** - Liang, Yongsheng, et al. 'Enhancing Gaussian Estimation of Distribution Algorithm by Exploiting Evolution Direction with Archive'. *ArXiv:1802.08989 [Cs]*, July 2018. *arXiv.org*, http://arxiv.org/abs/1802.08989.
- **[Bhattacharjee, Sourodeep]** - Bhattacharjee, Sourodeep & Gras, Robin. (2019). Estimation of Distribution using Population Queue based Variational Autoencoders. 1406-1414. 10.1109/CEC.2019.8790077.
- **[Baluja, Shumeet]** - Baluja, Shumeet "A Method for Integrating Genetic Search Based Function Optimization and Competitive Learning." (1994).
- **[Harik, Georges]** - Harik, Georges. (1997). Learning Gene Linkage to Efficiently Solve Problems of Bounded Difficulty Using Genetic Algorithms.
- **[Pelikan, Goldberg]** - Pelikan, Martin & Goldberg, David & Cantu-Paz, Erick. (2000). Linkage Problem, Distribution Estimation, and Bayesian Networks. Evolutionary computation. 8. 311-40. 10.1162/106365600750078808.
- **[Tamayo-Vera]** - D. Tamayo-Vera, A. Bolufé-Röhler and S. Chen, "Estimation multivariate normal algorithm with thresheld convergence," *2016 IEEE Congress on Evolutionary Computation (CEC)*, Vancouver, BC, 2016, pp. 3425-3432.
- **[Cai, Yunpeng]** - Cai, Yunpeng & Sun, Xiaomin & Xu, Hua & Jia, Peifa. (2007). Cross entropy and adaptive variance scaling in continuous EDA. Proceedings of GECCO 2007: Genetic and Evolutionary Computation Conference. 609-616. 10.1145/1276958.1277081.
- **[Ceberio]** - https://www.slideshare.net/InformaticaUCM/dealing-with-constraints-in-estimation-of-distribution-algorithms
- **[Garciarena, Unai]** - Garciarena, Unai & Santana, Roberto & Mendiburu, Alexander. (2018). Expanding variational autoencoders for learning and exploiting latent representations in search distributions. 849-856. 10.1145/3205455.3205645.
- **[Probst, Malte]** -Probst, Malte 'Generative Adversarial Networks in Estimation of Distribution Algorithms for Combinatorial Optimization'. *ArXiv:1509.09235 [Cs]*, 2, Aug. 2016. *arXiv.org*, http://arxiv.org/abs/1509.09235.

“This is a super-important quote”

- From an expert