Efficient Global Optimisation in Dynamic Environments

Juergen Branke, Sergio Morales-Enciso

Outline

- Motivation
- Evolutionary optimisation in dynamic environments
- Efficient Global Optimisation (EGO)
- Extensions to dynamic environments
- Experimental results
- Conclusion

Motivation

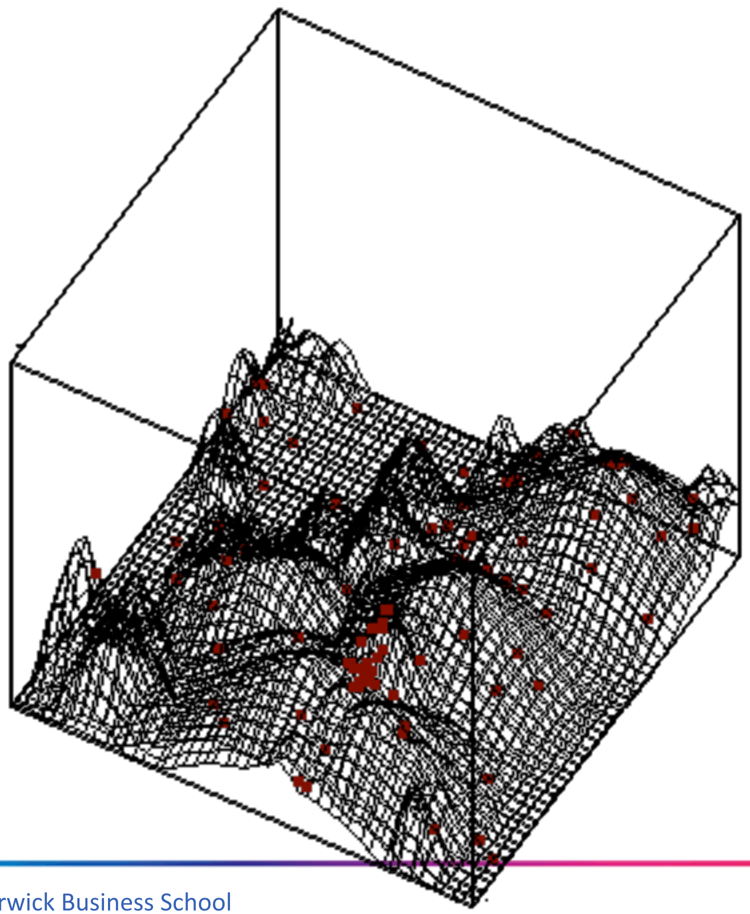
- Many optimisation problems are dynamic
  - Scheduling
  - Pickup & delivery
  - Changing quality of raw material
  - …
- Problem changes from finding the optimum to tracking the optimum

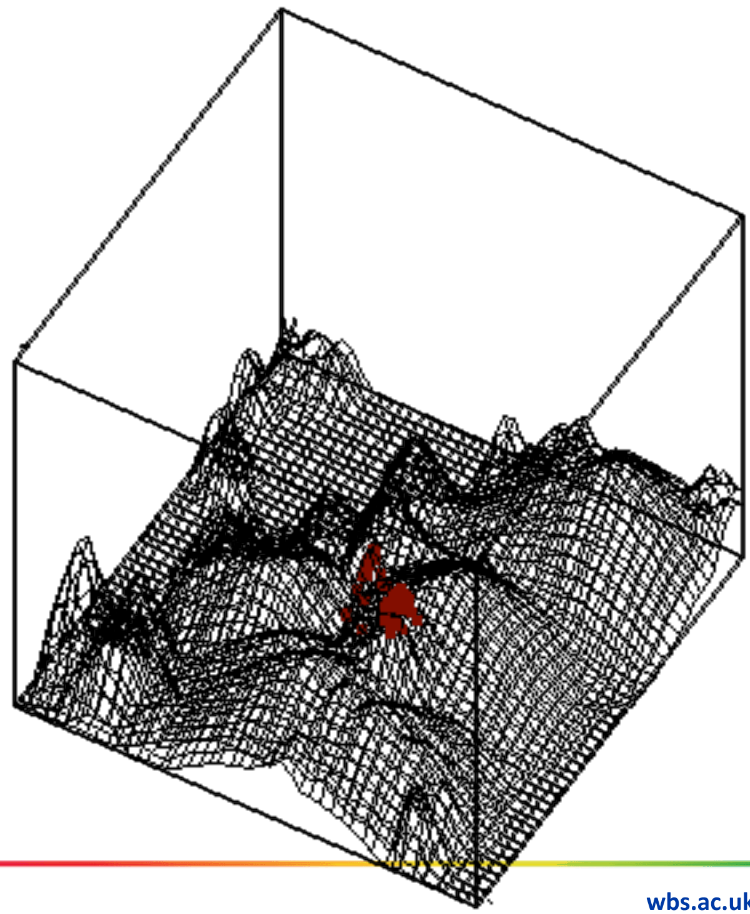
Nature is able to adapt

Evolutionary algorithms seem promising!

Evolutionary algorithms are not

Convergence of population limits adaptability





Possible Remedies

1. Restart after a change
   (only choice if changes are too severe)

But: Too slow

2. Generate diversity after a change
   – Hypermutation [Cobb 1990]

But: Randomisation destroys information,
only local search or similar to restart

Possible Remedies (2)

3. Maintain diversity throughout the run
   - Random Immigrants [Grefenstette 1992]
   - Thermodynamical GA [Mori et al. 1996]

But: Disturbes optimisation process

4. Memory-enhanced EAs
   – Implicit memory [Goldberg & Smith 1987, Lewis et al. 1998]
     - Redundant genetic representation (e.g. diploid)
   – Explicit memory [Ramsey & Grefenstette 1993, Branke 1999, Yang 2008]
     - Explicit rules which information to store in and retrieve from the memory

But: Only useful when optimum reappears at old location,
Problem of convergence remains

Possible Remedies (3)

5. Multi-Population approaches
   - Maintain different subpopulations on different peaks
     - adaptive memory
     - able to detect new optima
     - distance/similarity metric required
   - Self-Organizing Scouts [Branke et al. 2000
   - ClusteringPSO [Yang&Li 2010, Li & Yang 2012]

Maintains and updates memory of several good regions

Only few evaluations possible

- Limited time
- Expensive black-box optimisation problem

Efficient Global Optimisation (EGO)

- Fit a Gaussian Process (GP) to data
- Response model provides information about
  - expected value
  - uncertainty
- Use response model to determine next data point
- Expected improvement makes explicit trade-off between exploration and exploitation

Efficient Global Optimisation (EGO)

- Fit a Gaussian Process (GP) to data

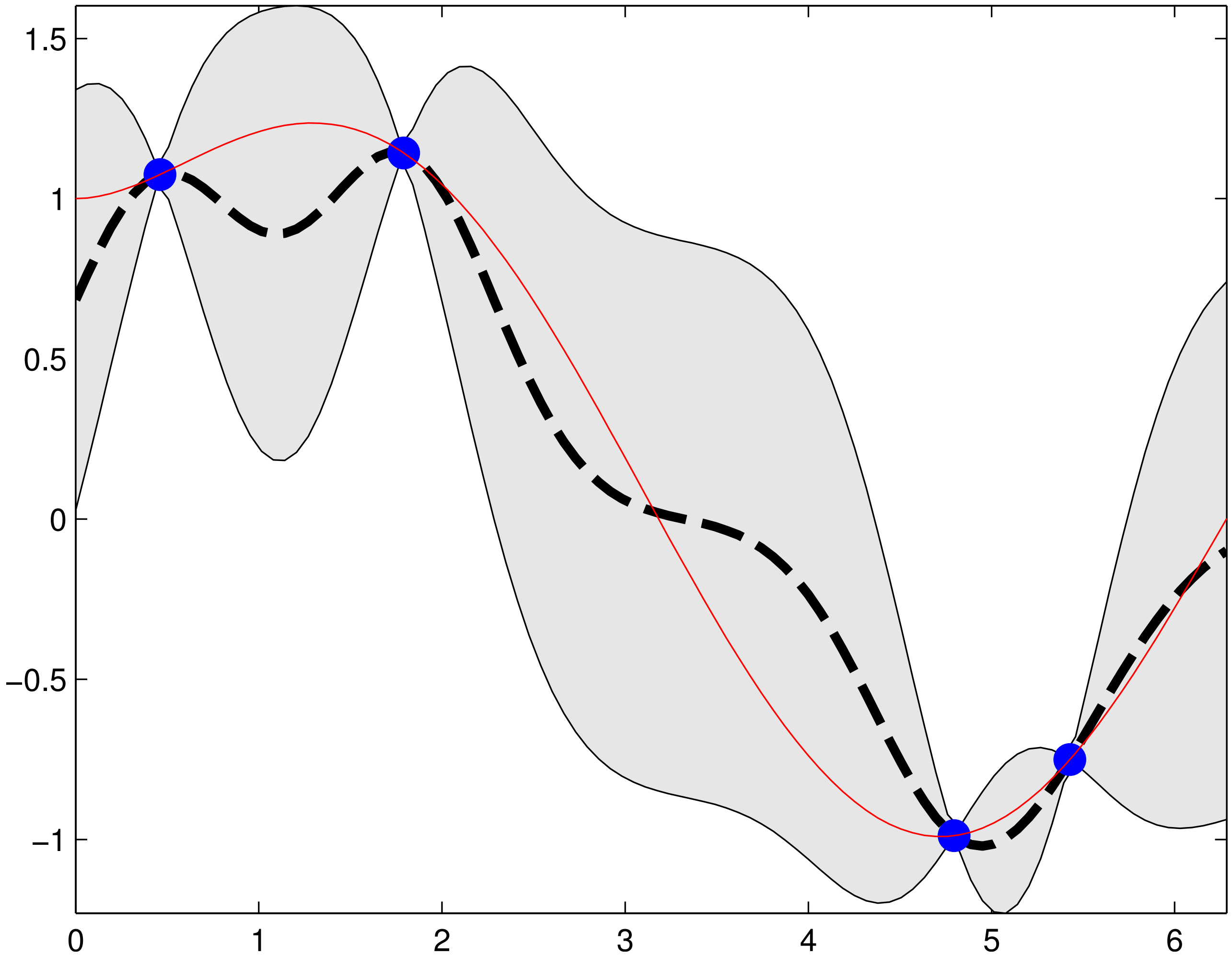
$$f(\vec{x}) \sim GP\big(m(\vec{x}), K(\vec{x}, \vec{x}')\big)$$

where $m(\vec{x}) = 0$

$$\vec{K} = \begin{bmatrix} k(\vec{x}_1, \vec{x}_1) & \cdots & k(\vec{x}_1, \vec{x}_n) \\ \vdots & \ddots & \vdots \\ k(\vec{x}_n, \vec{x}_1) & \cdots & k(\vec{x}_n, \vec{x}_n) \end{bmatrix}$$

$$\underbrace{k(\vec{x}, \vec{x}')}_{\text{kernel}} = \underbrace{\sigma_f^2}_{\text{max cov}} \; exp\left(- \sum_{d=1}^{D} \frac{(x_d - x'_d)^2}{2 \underbrace{\ell_d^2}_{\text{length scale}}} \right) + \underbrace{\sigma_n^2 \delta(\vec{x}, \vec{x}')}_{\text{measurement noise}}$$

Example: GP in 1 dimension



Adaptation to dynamic environments

How to integrate knowledge from history?

a) Designate old date as less reliable (noisy)

b) Add time as additional dimension

c) Modify mean prior

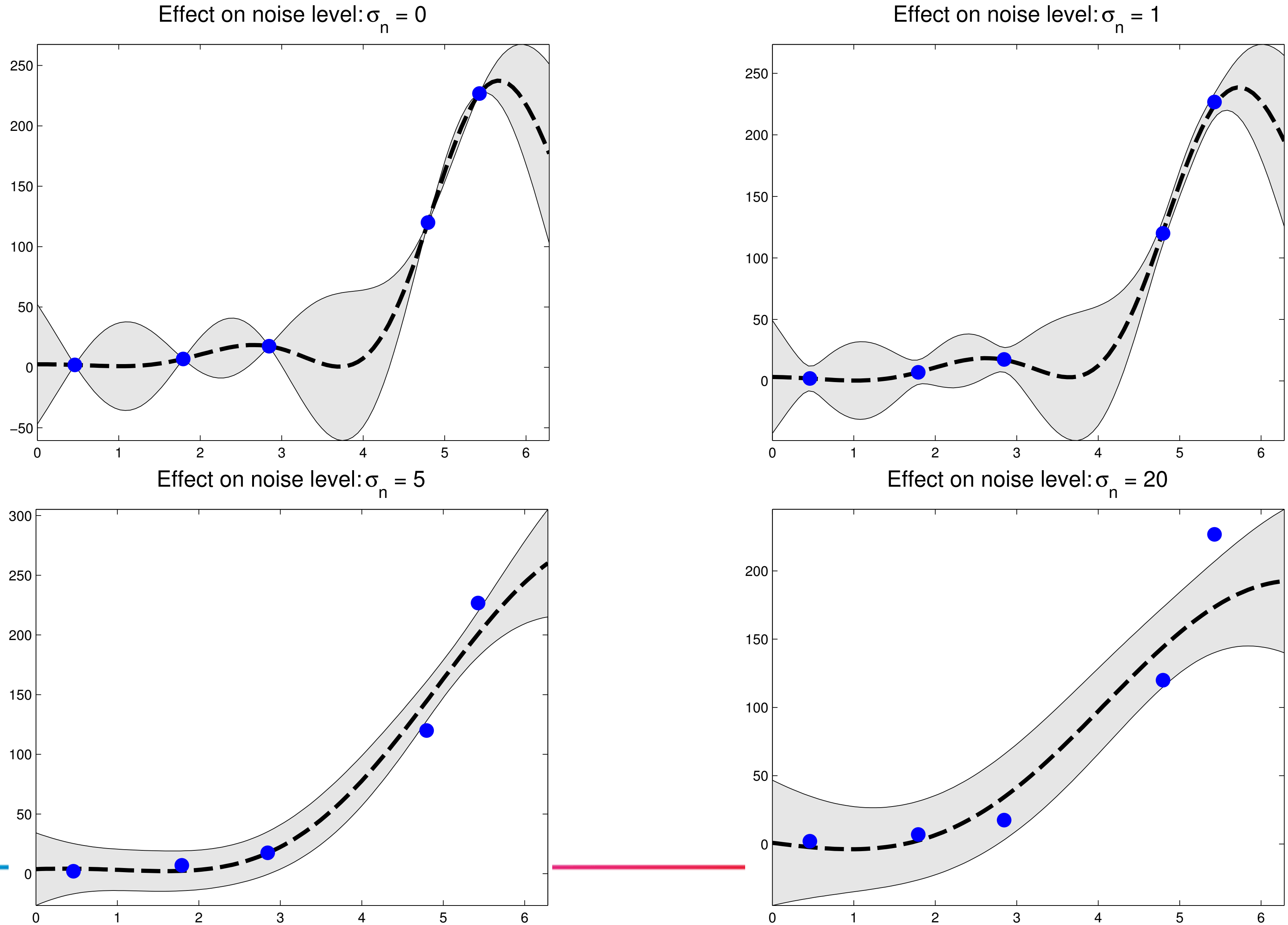
a) Add noise to old samples

$$k(\vec{x},\vec{x}') = \underbrace{\sigma_f^2}_{\text{max cov}} \; exp\left(-\sum_{d=1}^{D} \frac{(x_d - x'_d)^2}{2 \underbrace{\ell_d^2}_{\text{length scale}}} \right) + \underbrace{\sigma_n^2(\tau)\delta(\vec{x},\vec{x'})}_{\text{measurement noise}}$$

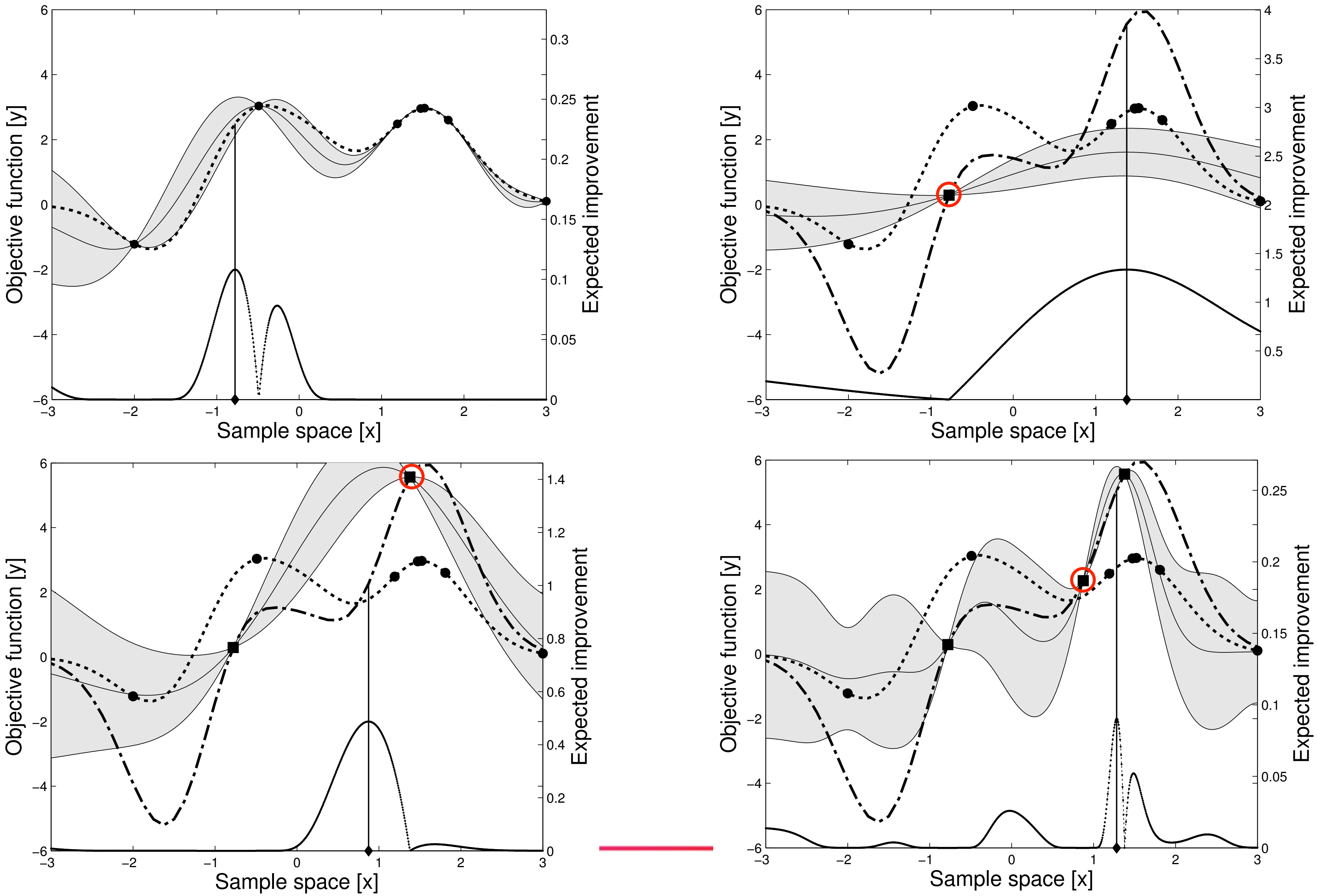
$$\sigma_n^2(\tau) = (\tau_c - \tau)s^2$$

- Noise level s^2 is user-specified parameter

Effect of noise



Example



Objective function [y]
Expected improvement
Sample space [x]
Objective function [y]
Expected improvement
Sample space [x]
Objective function [y]
Expected improvement
Sample space [x]
Objective function [y]
Expected improvement
Sample space [x]

b) Additional dimension

- Time stamp as additional dimension

$$k(\vec{x}, \vec{x}') = \underbrace{\sigma_f^2}_{\text{max cov}} \; exp\Big(- \sum_{d=1}^{D+1} \frac{(x_d - x'_d)^2}{2 \underbrace{\ell_d^2}_{\text{length scale}}} \Big)$$

- Length scale parameter is learned by GP

c) Transferring the mean prior

- Instead of a zero mean prior, take the best estimate of the previous epoch as a prior mean function

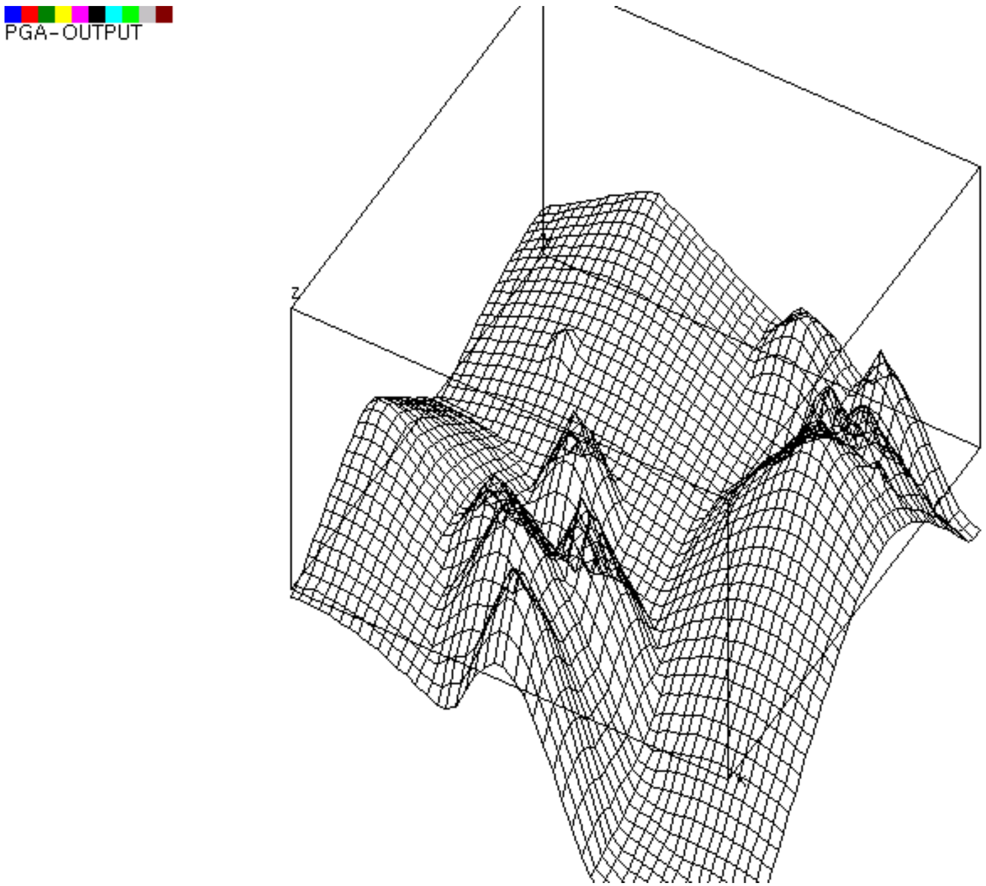
$$f(\vec{x}) \sim GP\big(f_{\tau-1}(\vec{x}), K_\tau(\vec{x}, \vec{x}')\big) \quad \forall\ \tau > 0$$

Measuring performance

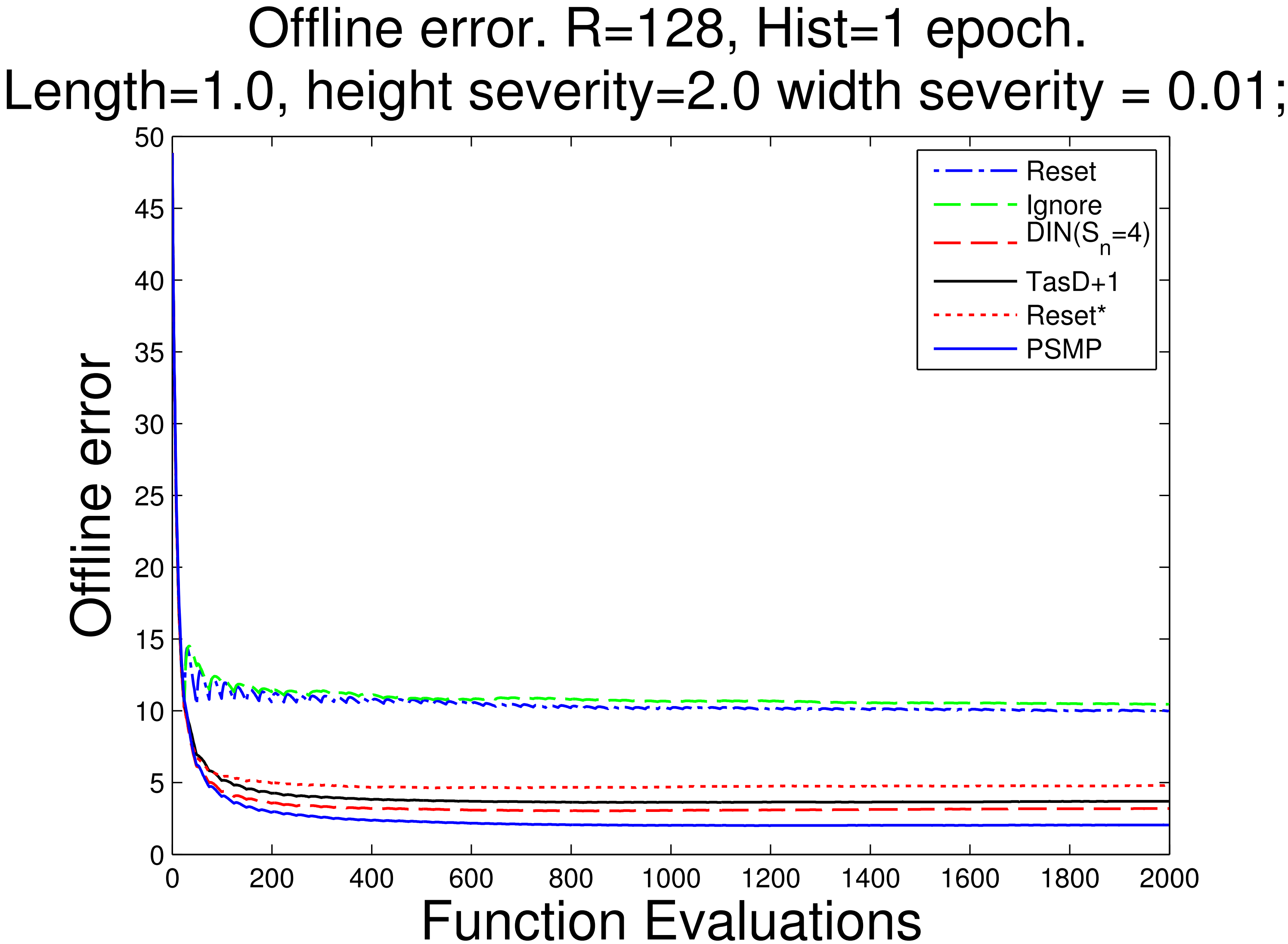
- Average error – If every solution is tested in real world
- Offline error – If best so far solution is tested in real world
- Best before change – If optimization is done before a solution is implemented

Moving Peaks Benchmark [Branke1999]

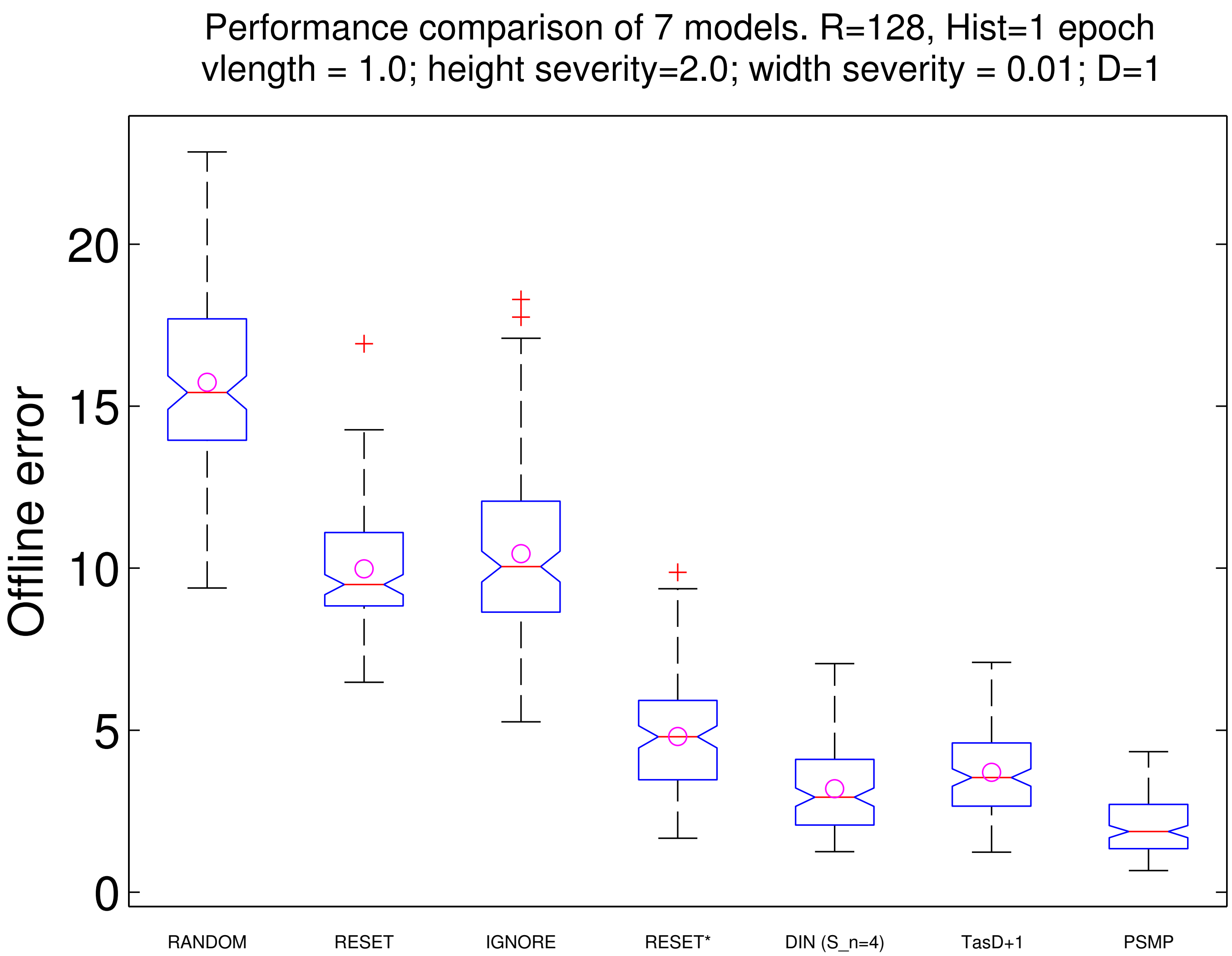
- Several peaks, changing in
  - Location
  - Height
  - Width
- Used here:
  - 5 peaks
  - 1 D
  - Change every 25 evaluations



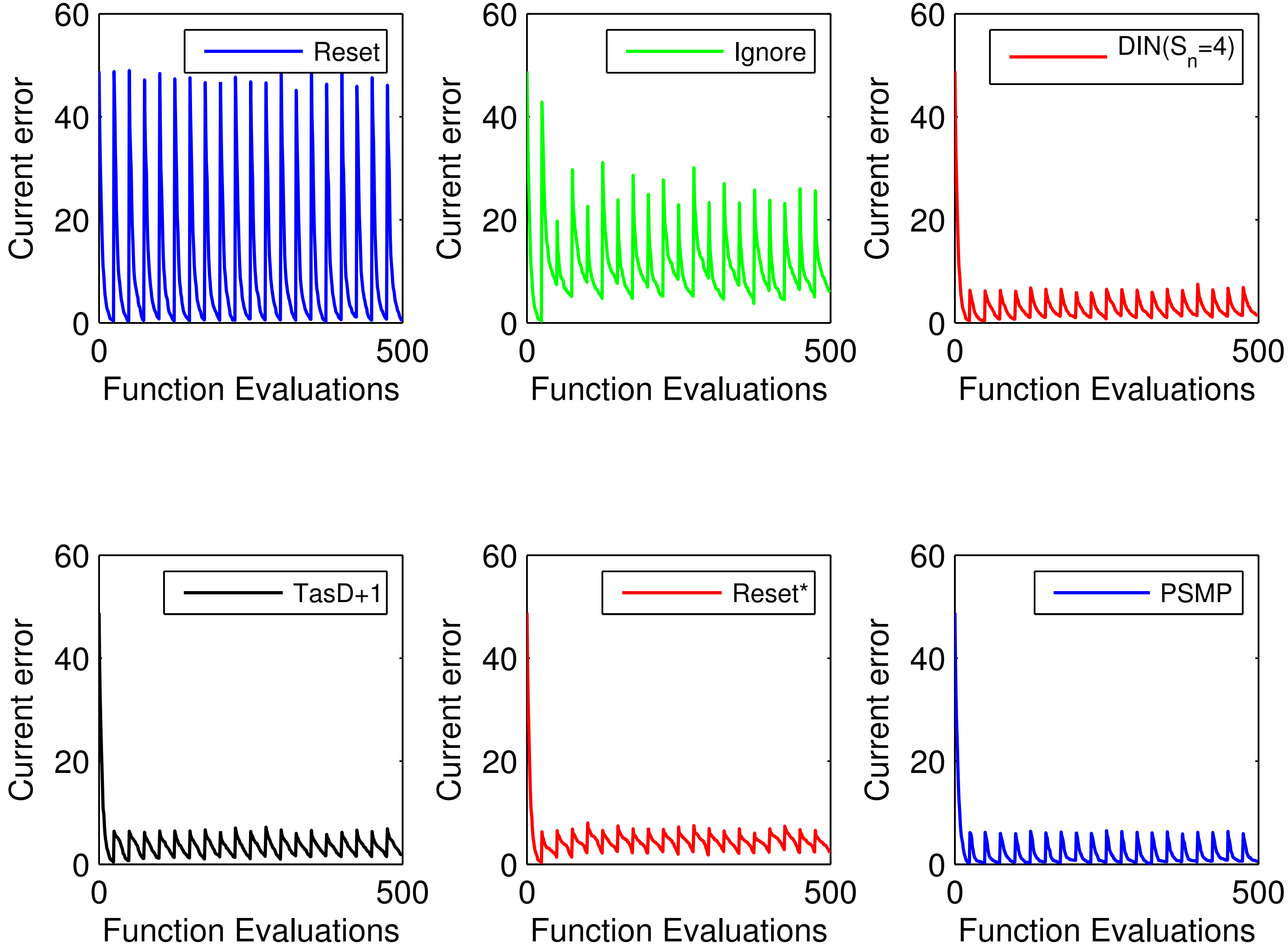
Offline error



Final offline error



Current error. R=128, Hist=1 epoch.
Length=1.0, height severity=2.0 width severity = 0.01;



Conclusion

- With appropriate adjustments, evolutionary computation is able to continuously adapt
- Proposed new variants of EGO for dynamic optimisation problems
- Results show benefit of transferring information from previous epochs
- Promising for applications where very few function evaluations are possible

Future work

- Move to GP variants that can deal with larger dimensionality and more datapoints
- Learning of noise parameter
- Test on other functions