High-dimensional sampling algorithms comparison for particle and astrophysics applications

Roberto Ruiz de Austri Bazan, Judita Mamužić

IFIC / CSIC - University of Valencia

28 April 2021, I Workshop de Computing y Software de la Red Española de LHC

IFIC INSTITUT DE FÍSICA CORPUSCULAR

CSIC CONSEJO SUPERIOR DE INVESTIGACIONES CIENTÍFICAS

VNIVERSITAT DE VALÈNCIA

UNIÓN EUROPEA
Fondo Social Europeo
El FSE invierte en tu futuro

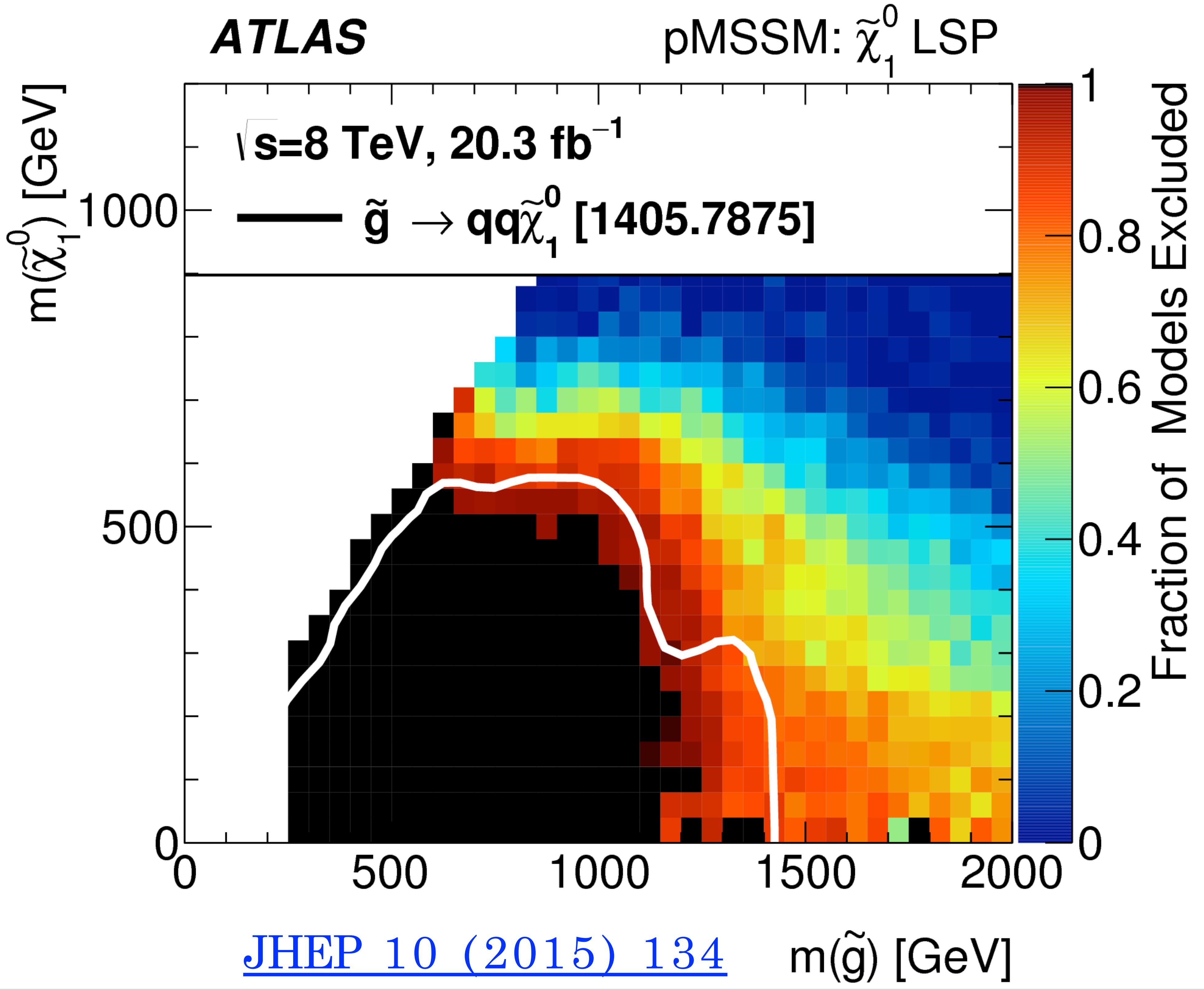
GENERALITAT VALENCIANA

EXCELENCIA SEVERO OCHOA

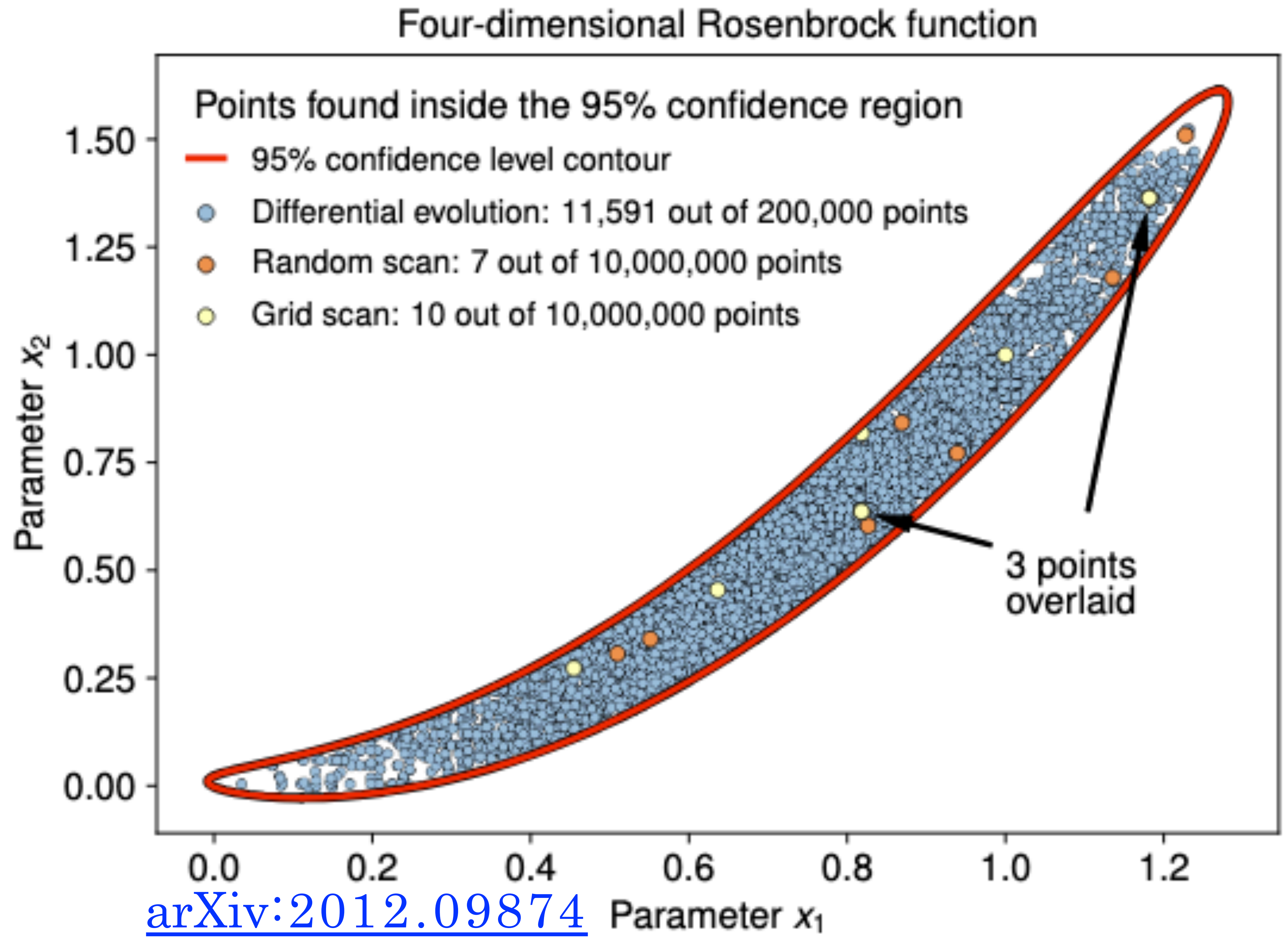
ATLAS EXPERIMENT

Introduction

- With increasing computing power, a number of new approaches in physics searches are being studied.
- Frequent requirement is sampling in a multi-dimensional parameter space, e.g. 19-dimensional pMSSM scan for interpretation of all Supersymmetry analyses.
- Random sampling selects points only at the boundary, better/smart sampling needed.
- First step is finding an optimal minimisation algorithm for physics analysis (particle, astro-particle physics).
- Comparison of minimisation algorithms performance for increasing number of dimensions in arXiv:2101.04525.



JHEP 10 (2015) 134



arXiv:2012.09874

Optimisation Algorithms

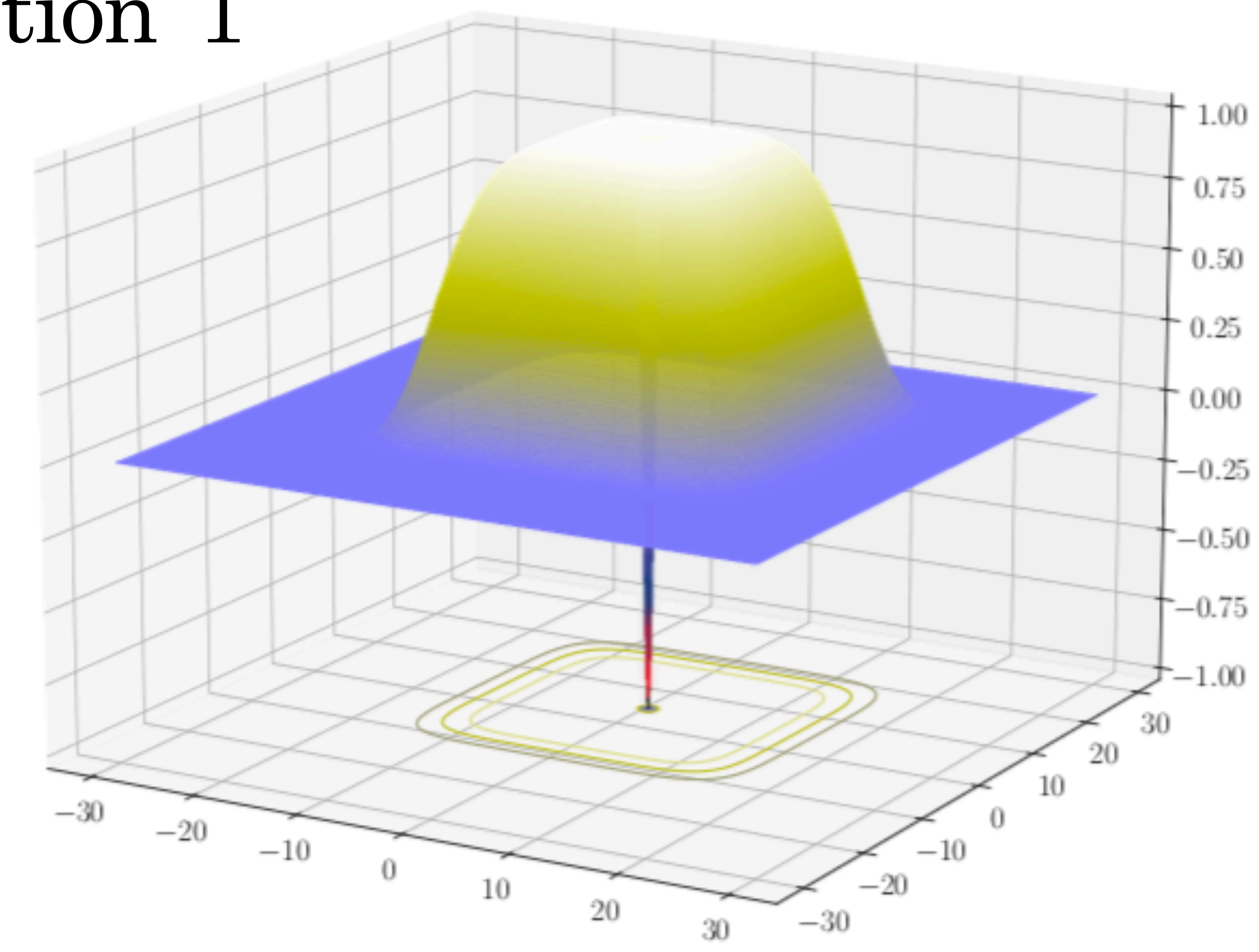
- Help user in choosing the sampling algorithm and show examples for comparison.
- Overview of state of the art minimisation algorithms, including also the ones not so frequently used in HEP.
- Considering gradient free algorithms (no e.g. Minuit).
- Compare their performance for analytic test functions and for a realistic physics example (MSSM7)

Contribution using Artemisa computing facility.

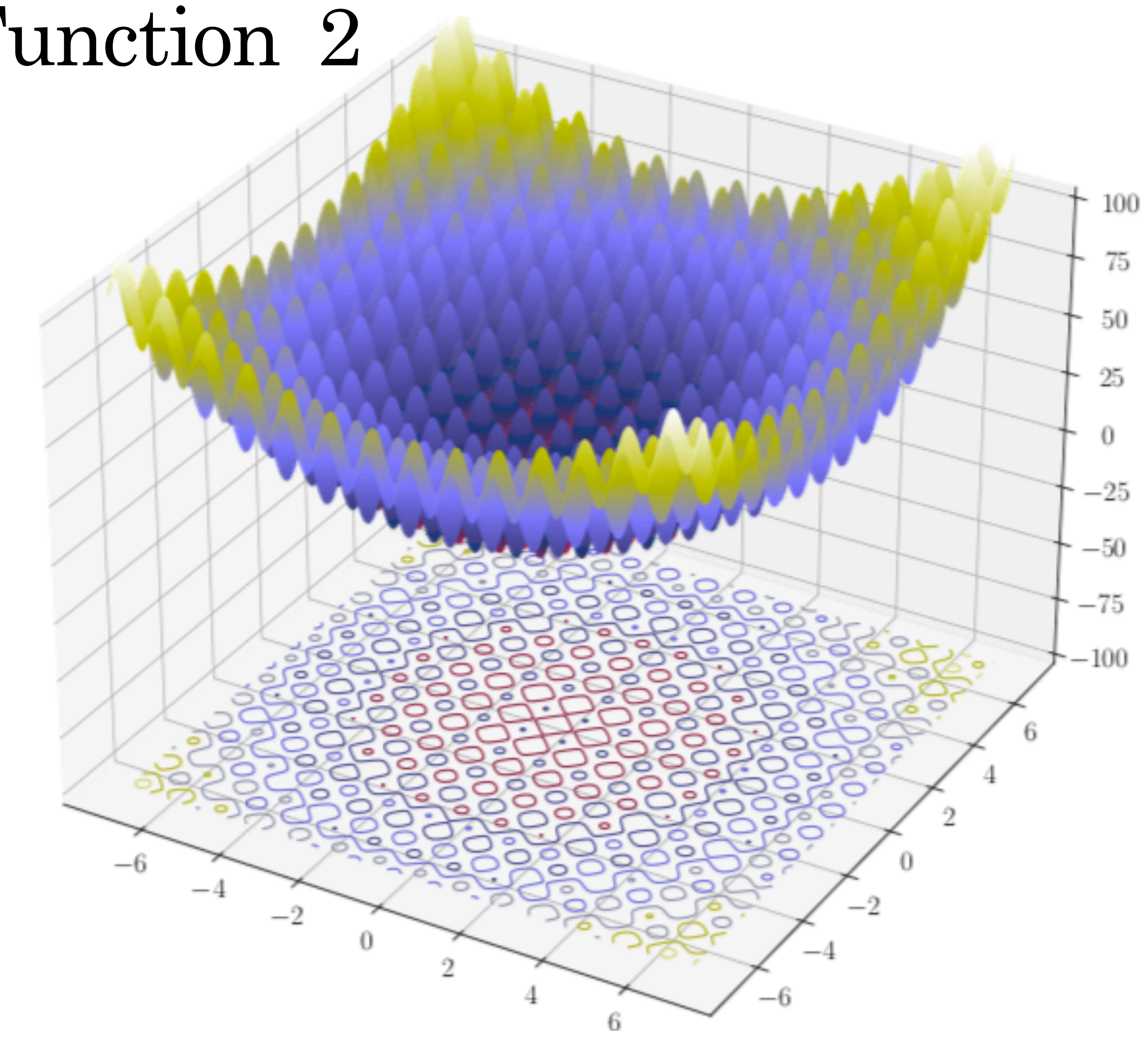
Parameter	Explored values	Type
AMPGO		
Number of sampled points	2000, 5000, 10000, 20000	Resolution
CMA-ES		
Function tolerance	10^{-11}, 10^{-7}, 10^{-4}, 10^{-1}	Convergence
Population size (λ)	20, 50, 100, 500	Resolution
Diver		
Threshold for convergence	10^{-4}, 10^{-3}, 10^{-2}, 10^{-1}	Convergence
Population size	2000, 5000, 10000, 20000	Resolution
Parameter adaptation scheme	λjDE	-
Gaussian Particle Filter		
Width decay	0.90, 0.95, 0.99	Convergence
Logarithmic sampling	`True`, `False`	Hint
Survival rate	0.2, 0.5	Reliability
Initial gaussian width	2	Reliability
GPyOpt		
Threshold for Convergence	10^{-6}, 10^{-5}, 10^{-4}, 10^{-3}, 10^{-2}, 10^{-1}	Convergence
Particle Swarm Optimisation		
Threshold for convergence	10^{-4}, 10^{-3}, 10^{-2}, 10^{-1}	Convergence
Population size	2000, 5000, 10000, 20000	Resolution
Adaptive ϕ	`True`	Reliability
Adaptive ω	`True`	Reliability
PyGMO Artificial Bee Colony		
Generations	100, 250, 500, 750	Resolution
Maximum number of tries	10, 50, 100	Reliability
PyGMO Differential Evolution		
Generations	100, 250, 500, 750	Resolution
Parameter adaptation scheme	iDE, jDE	-
PyGMO Grey Wolf Optimisation		
Generations	10, 50, 100, 1000	Resolution
random sampling		
Number of points	10, 50, 100, 500, 1000, 5000, 10000, 50000, 100000, 500000, 1000000	Resolution
Trust Region Bayesian Optimisation (TuRBO)		
Max #evaluations / iteration	64, 100	Convergence

Test Function: 4 Analytic

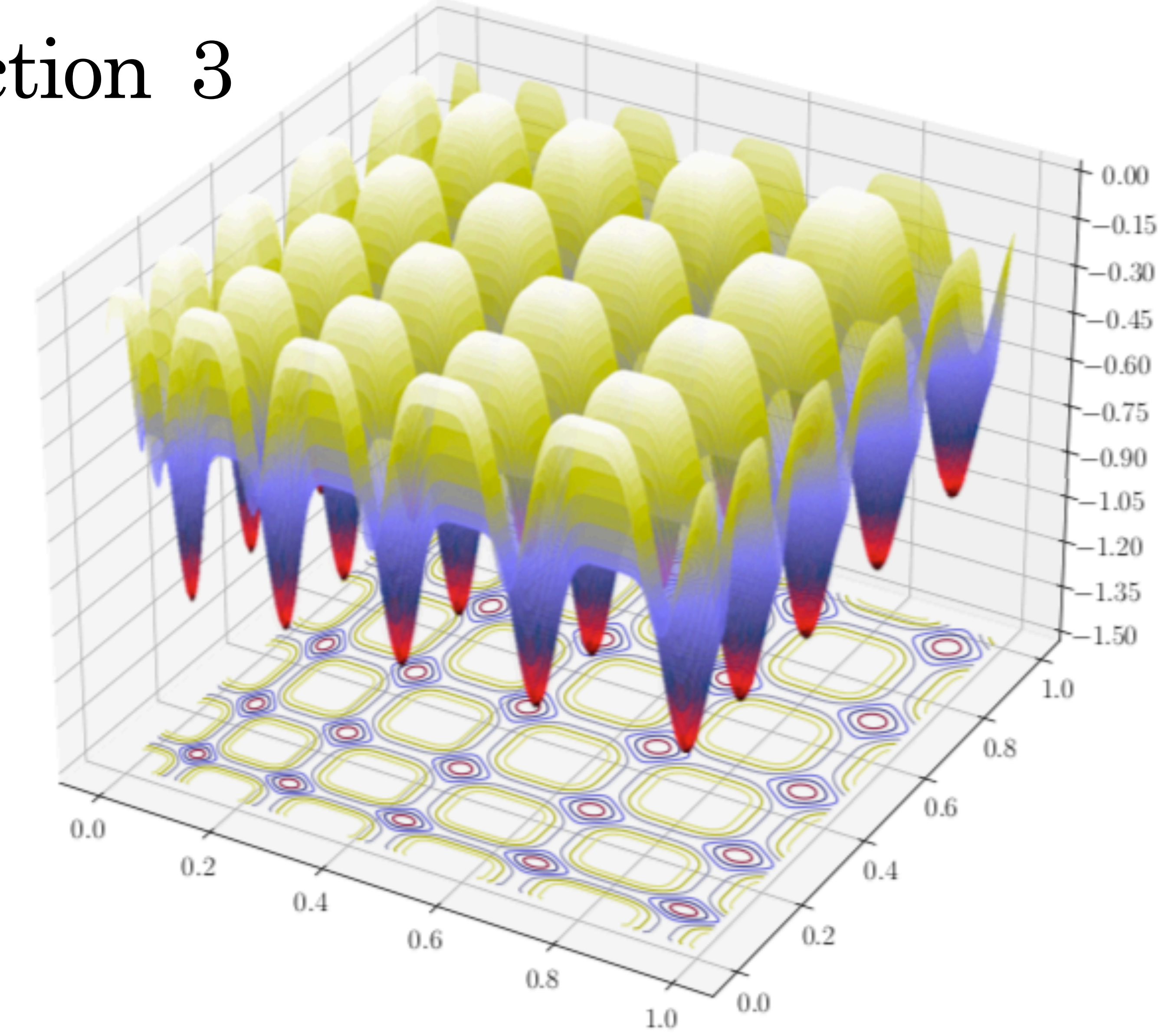
Function 1



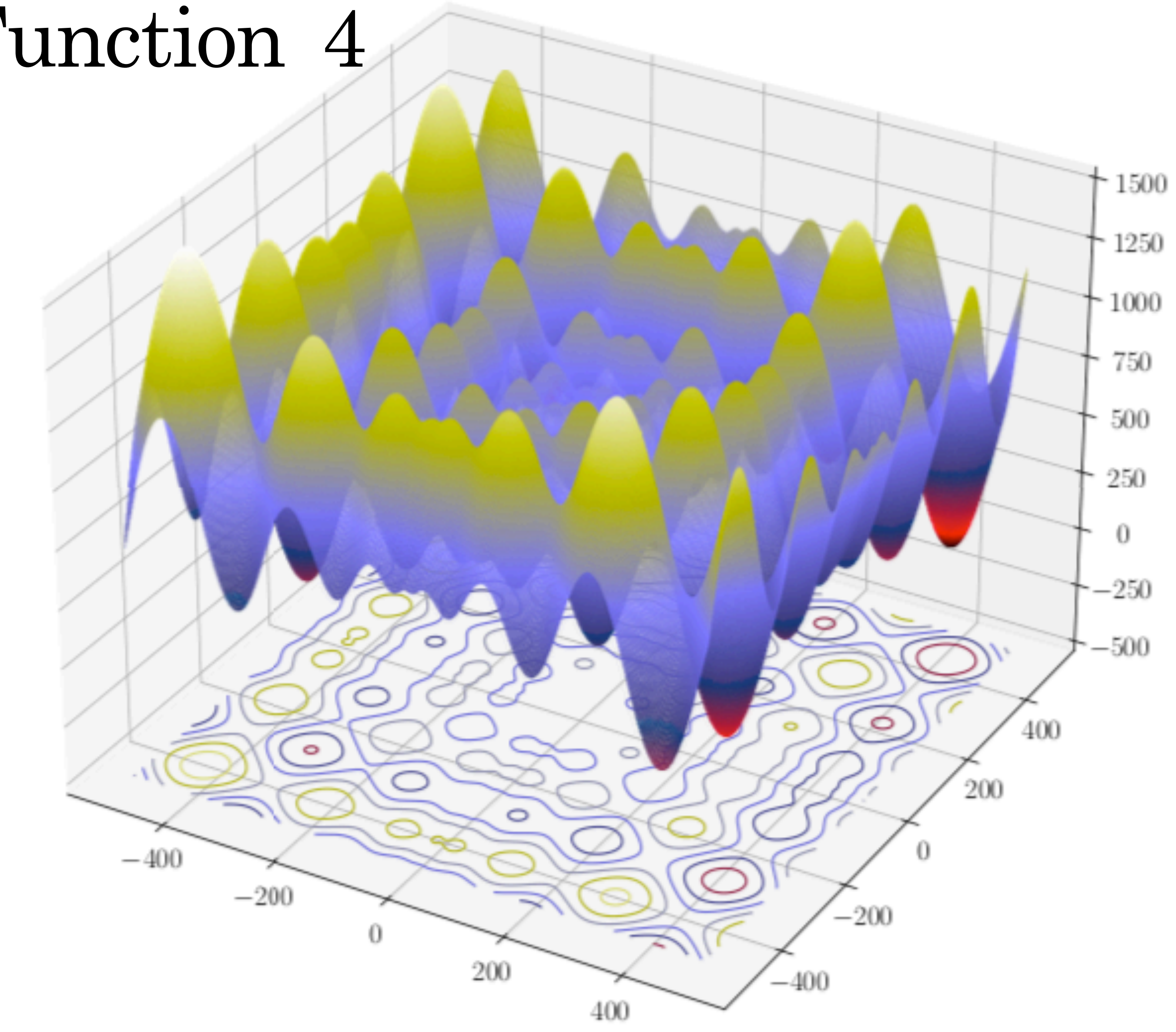
Function 2



Function 3



Function 4

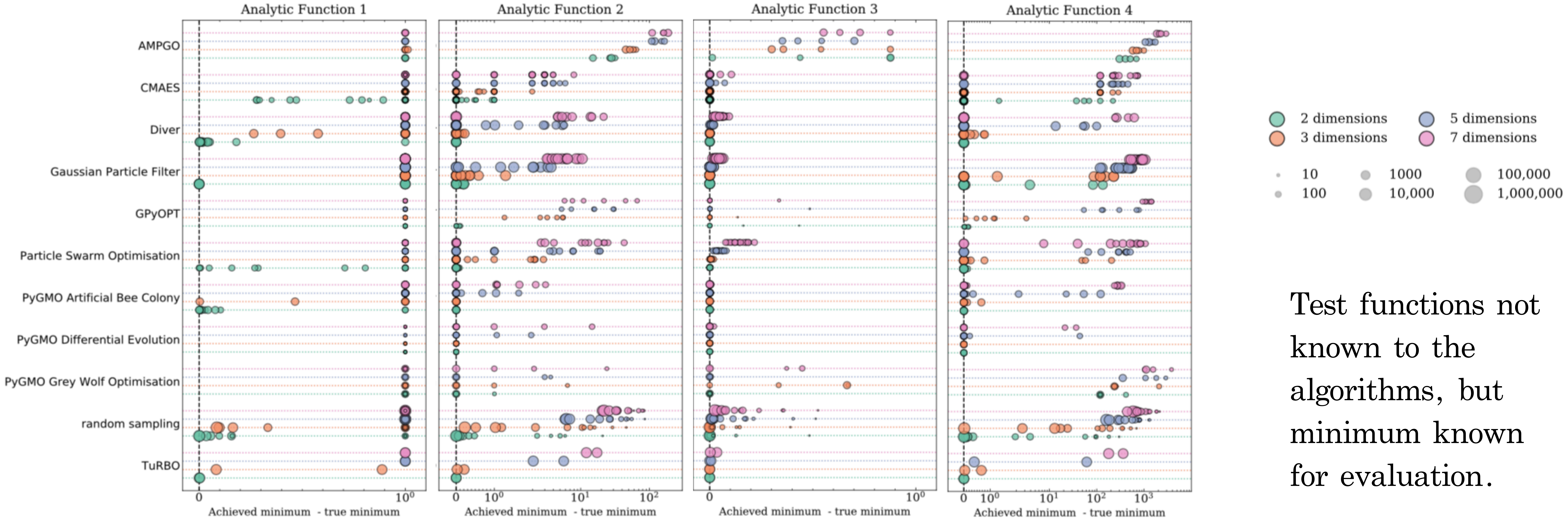


Compare algorithm performance using analytic test functions:

- Function 1: Sharp peak
- Function 2: High number of local minima
- Function 3: High number of global minima
- Function 4: High number of local minima

Functions shown in 2D, tests performed in 2-7D.

Performance: 4 Analytic



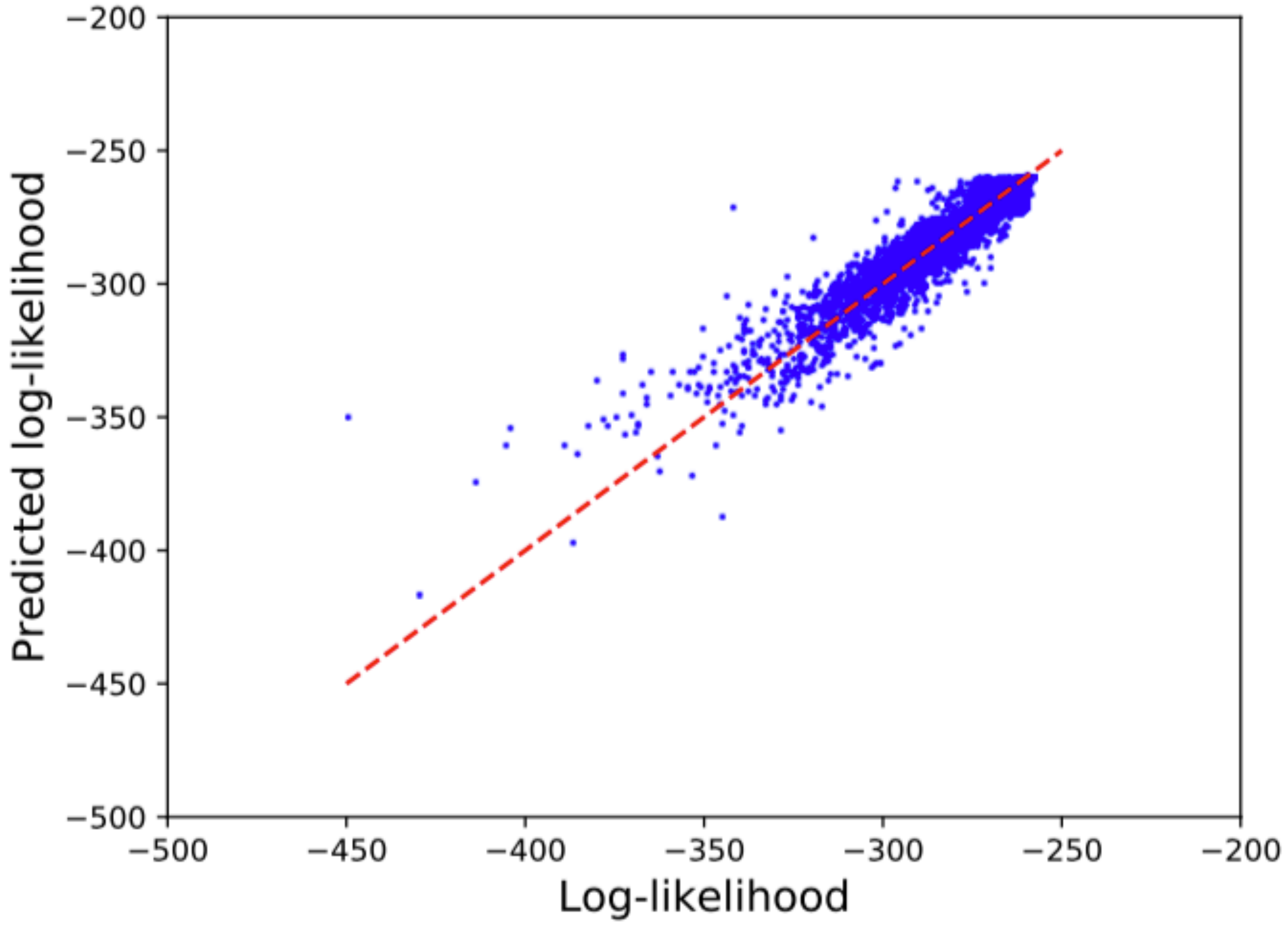
Test functions not known to the algorithms, but minimum known for evaluation.

- Performance with high dimensions (1-4): CMA-ES, Diver, PyGMO Artificial Bee Colony, PyGMO Differential
- Average number of evaluations (1-4): varies
- Finding a sharp minimum (1): Diver, Turbo and random sampling work better for low dimensions, but number of evaluations is relevant.
- Finding global minimum (2&4): CMA-ES, Diver, Particle Swarm Optimisation, PyGMO Artificial Bee Colony, PyGMO Differential evolution good for high dimensions.

==> Different algorithms need to be explored for different applications.

Test Function: MSSM7

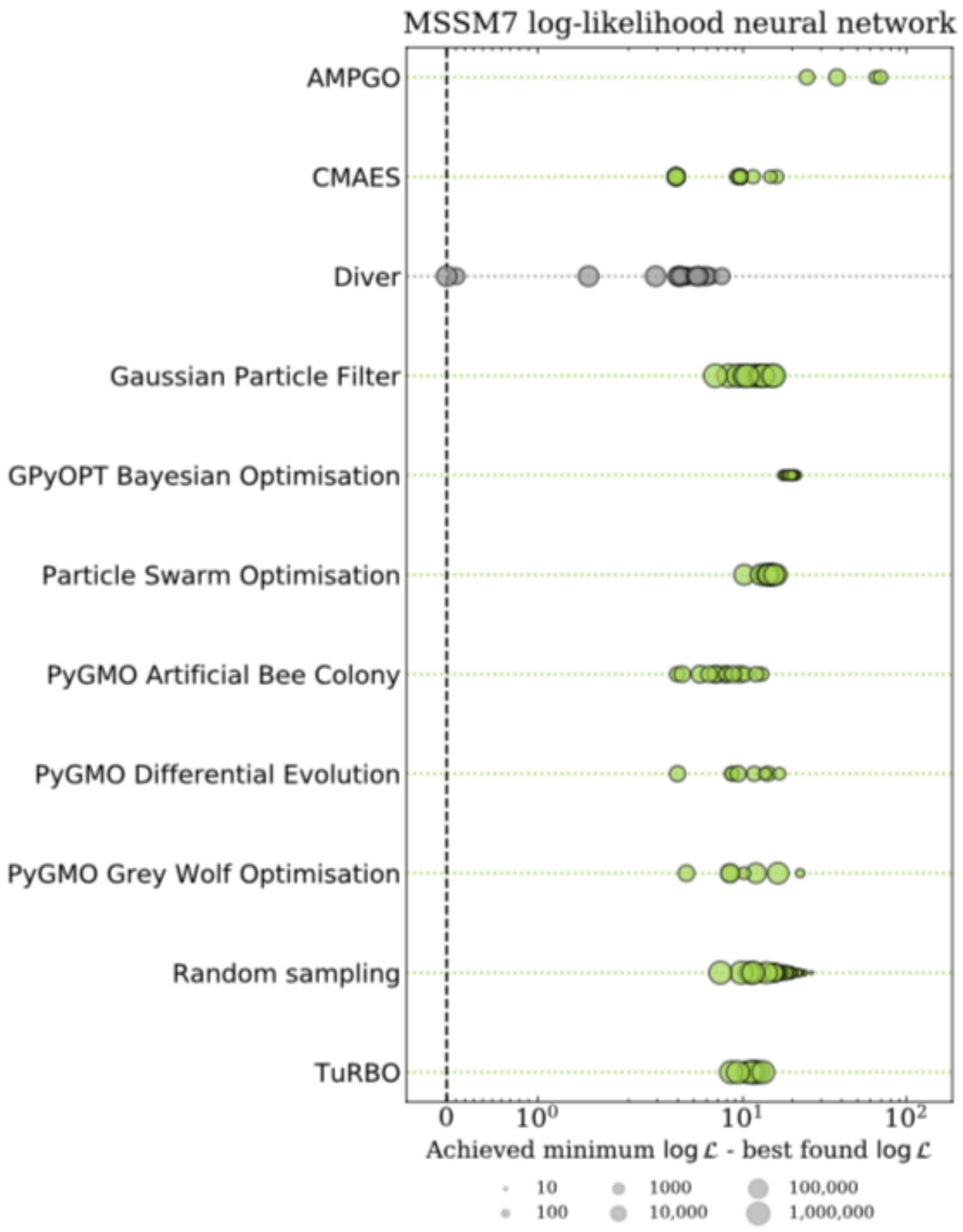
- Compare algorithms for high number of dimensions with a very complicated function.



MSSM7: 7-parameter pMSSM:

- Soft masses M_2, $m_{\tilde{f}}^2$, $m^2_{H_u}$, $m^2_{H_d}$
- Trilinear couplings for the third generation of quarks A_{u_3} , A_{d_3}
- $\tan\beta$ (plus the input scale Q = 1 TeV and the sign of μ , which was chosen to be positive)
- Strong coupling constant, the top quark mass, the local dark matter density, and the nuclear matrix elements for the strange, up and down quarks.
- Global fit on a 12-dimensional parameter space.
- Fast likelihood function interpolation using deep neural network.
- Generate train sample 2.3 x 10^7.
- Minimum of the function not known.

Performance: MSSM7



- Best performing algorithm is Diver (however the same algorithm was used to sample the test samples, gives unavoidable bias by construction).

==> Different algorithms have quite poor performance for high dimensional function.

==> Different algorithms should be tested for specific applications.

Summary

- Optimisation algorithms and sampling in high dimensions needed in most of the physics studies.
- Algorithm comparison done for 4 analytic test functions and realistic physics example.
- For a new study different algorithms should be compared, this study should help in the choice of an optimal algorithm and understanding of results.
- Framework (software and test functions) for the study publicly available at [high-dimensional-sampling]() (users can use the code for their optimisation).
- Results obtained at [DarkMachines.org]()

	Finding sharp minimum	Finding global minimum	Performance with high dimensions	Average number of evaluations
AMPGO	bad	bad	bad	low
CMA-ES	very low dimensions	good	good	medium
Diver	low dimensions	good	good	high
Gaussian Particle Filter	very low dimensions	low dimensions	highly configuration and function dependent	high
GPyOpt	bad	low dimensions	function dependent	low
Particle Swarm Optimisation	very low dimensions	good	configuration dependent	medium
PyGMO Artificial Bee Colony	low dimensions	good	good	medium
PyGMO Differential Evolution	bad	good	good	low
PyGMO Grey Wolf Optimisation	bad	bad	function dependent	medium
TuRBO	low dimensions	moderate dimensions	function dependent	high
random sampling	low dimensions	low dimensions	function dependent	high