Introduction to Blackbox Optimization Benchmarking

(original slides by Nikolaus Hansen)

Black-Box Optimization (Search)

Minimize an objective function (also: cost, loss, error, or fitness function)

$$f: \mathcal{X} \subset \mathbb{R}^n \to \mathbb{R}, \quad x \mapsto f(x)$$

in a black-box scenario (direct search, no gradients)

$$x \longrightarrow \int f(x)$$

where the black box can be

- non-linear, non-convex, discontinuous, dynamic, stochastic
- from milli-seconds to hours to evaluate

Objective:

- ullet convergence to a global essential infimum of f as fast as possible
- (informally, time-finite) find $x \in \mathcal{X}$ with small f(x) value using as few back-box calls (function evaluations) as possible

Why Do We Want to Evaluate Optimizers?

- understanding of algorithms
- algorithm selection
- putting algorithms to a standardized test
 - simplify judgement
 - simplify comparison
 - regression test under algorithm changes

How to Evaluate Algorithms

We can measure performance on

- real world problems
 - expensive
 - comparison is typically limited to certain domains
 - experts have limited interest to publish
- "artificial" benchmark functions
 - cheap
 - controlled
 - data acquisition is comparatively easy
 - problem of representativity
- caveat: parameter of algorithms

How to Evaluate Search Algorithms

we need

- Meaningful quantitative measure on benchmark functions or real world problems
- Account for meta-parameter tuning

tuning, if needed, can be quite expensive

Account for invariance properties

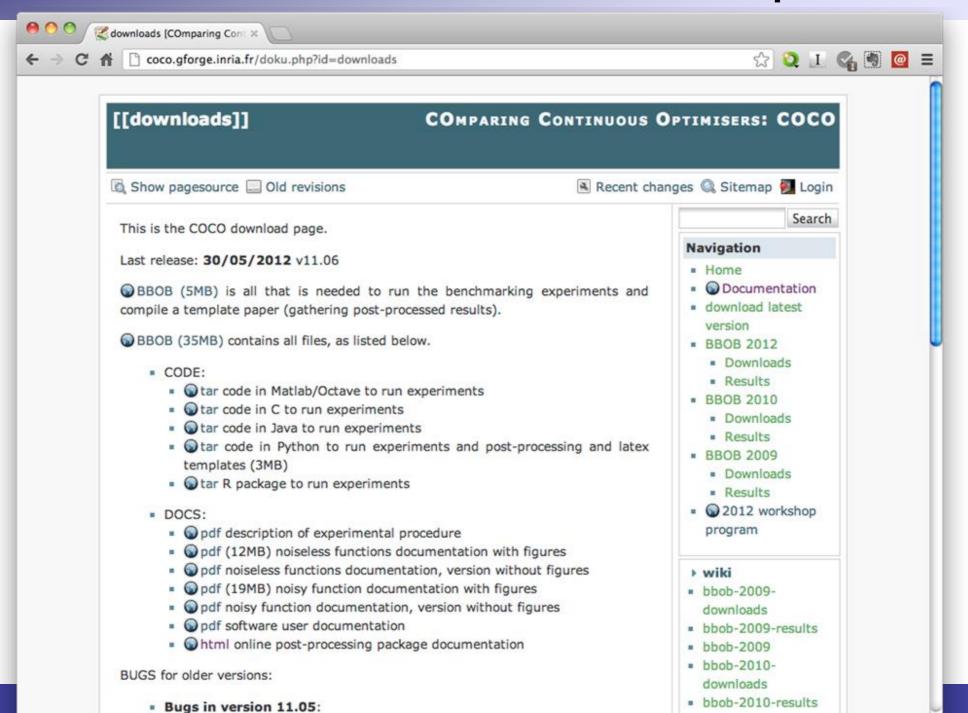
prediction of performance is based on "similarity", ideally equivalence classes of functions

Account for algorithm internal costs

often negligible, depending on the objective function cost

BBOB in Practice (using COCO)

COCO (COmparing Continuous Optimizers): a tool for black-box optimization benchmarking



Name	Date Modified	Size	Kind		
▼ 🛅 bbob.v11.06	Today, 1:15		Folder		
▶ 🛅 c	May 30, 2012 12:07		Folder		
docs 🚞 docs	October 27, 2012 0:58		Folder	•	
▶ 🛅 java	May 30, 2012 12:07		Folder		
latextemplates	May 30, 2012 12:06		Folder		
► 🚞 matlab	May 30, 2012 12:06		Folder		
▶ 🚞 python	May 30, 2012 12:06		Folder	Ĭ.	
▶ 🛅 r	May 30, 2012 12:07		Folder	▼	
Macintosh HD 🔃 Users 👚 hansen 🔯 Downloads 🧰 bbob.v11.06					

Name	▲ Date Modified	Size	Kind		
▼ 🛅 bbob.v11.06	Today, 1:15		Folder		
▶ 🛅 c	May 30, 2012 12:07		Folder		
▶ 🚞 docs	October 27, 2012 0:58		Folder		
▶ 🛅 java	May 30, 2012 12:07		Folder		
latextemplates	May 30, 2012 12:06		Folder		
▼ 🛅 matlab	May 30, 2012 12:06		Folder		
benchmarkinfos.txt	February 9, 2009 16:29	4 KB	Geditument		
m benchmarks.m	February 10, 2011 16:24	86 KB	Objecrce File		
m benchmarksnoisy.m	February 10, 2011 16:24	102 KB	Objecrce File		
m exampleexperiment.m	February 1, 2012 19:52	4 KB	Objecrce File		
m exampletiming.m	March 7, 2012 14:40	4 KB	Objecrce File		
m fgeneric.m	December 7, 2011 17:56	33 KB	Objecrce File		
□ LICENSE.txt □	February 1, 2012 20:23	4 KB	Geditument		
m MY_OPTIMIZER.m	February 14, 2011 19:30	4 KB	Objecrce File		
README.txt	May 19, 2011 10:47	4 KB	Geditument		
▶	May 30, 2012 12:06		Folder	Ă	
▶ i r	May 30, 2012 12:07		Folder	¥	
Macintosh HD → 🖭 Users → 🏠 hansen → 🔯 Downloads → 🧰 bbob.v11.06					

Matlab script (exampleexperiment.m):

```
dimensions = [2, 3, 5, 10, 20, 40]; % small dimensions first, for CPU reasons
functions = benchmarks('FunctionIndices'); % or benchmarksnoisy(...)
instances = [1:5, 31:40]; % 15 function instances
for dim = dimensions-
  for ifun = functions-
    for iinstance = instances-
     fgeneric('initialize', ifun, iinstance, datapath, opt); -
     MY_OPTIMIZER('fgeneric', dim, fgeneric('ftarget'), eval(maxfunevals) - f
     disp(sprintf([' f%d in %d-D, instance %d: FEs=%d with %d restarts, fbes
     fgeneric('finalize');¬
   end
   disp([' date and time: ' num2str(clock, ' %.0f')]);¬
 end-
 disp(sprintf('---- dimension %d-D done ----', dim));-
end⊸.
```

Interface: MY_OPTIMIZER(function_name, dimension, optional_args)

Running the experiment at an OS shell:

```
$ nohup nice octave < exampleexperiment.m > output.txt &
$ less output.txt
GNU Octave, version 3.6.3
Copyright (C) 2012 John W. Eaton and others.
This is free software; see the source code for copying conditions.
[...]
Read http://www.octave.org/bugs.html to learn how to submit bug reports.
For information about changes from previous versions, type `news'.
  f1 in 2-D, instance 1: FEs=242, fbest-ftarget=-8.1485e-10, elapsed time [h]: 0.00
  f1 in 2-D, instance 2: FEs=278, fbest-ftarget=-6.0931e-09, elapsed time [h]: 0.00
  f1 in 2-D, instance 3: FEs=242, fbest-ftarget=-9.2281e-09, elapsed time [h]: 0.00
  f1 in 2-D, instance 4: FEs=302, fbest-ftarget=-4.5997e-09, elapsed time [h]: 0.00
  f1 in 2-D, instance 5: FEs=230, fbest-ftarget=-9.8350e-09, elapsed time [h]: 0.00
  f1 in 2-D, instance 6: FEs=284, fbest-ftarget=-7.0829e-09, elapsed time [h]: 0.00
  f1 in 2-D, instance 7: FEs=278, fbest-ftarget=-6.5999e-09, elapsed time [h]: 0.00
  f1 in 2-D, instance 8: FEs=272, fbest-ftarget=-8.7044e-09, elapsed time [h]: 0.00
  f1 in 2-D, instance 9: FEs=248, fbest-ftarget=-2.6316e-09, elapsed time [h]: 0.00
  f1 in 2-D, instance 10: FEs=302, fbest-ftarget=-4.6779e-09, elapsed time [h]: 0.00
  f1 in 2-D, instance 11: FEs=272, fbest-ftarget=-5.1499e-09, elapsed time [h]: 0.00
[...]
      date and time: 2013 3 29 19 59 26
  f2 in 2-D, instance 1: FEs=824, fbest-ftarget=-7.0206e-09, elapsed time [h]: 0.00
  f2 in 2-D, instance 2: FEs=572, fbest-ftarget=-9.2822e-09, elapsed time [h]: 0.00
[...]
```

Name A	Date Modified	Size	Kind	
▼ 🚞 bbob.v13.05	March 8, 2013 13:04		Folder	<u>*</u>
▶	March 5, 2013 23:04		Folder	
▶ 🛅 docs	March 6, 2013 13:56		Folder	
▶ 🛅 java	March 5, 2013 23:04		Folder	
latextemplates	March 5, 2013 23:04		Folder	
▶ matlab	March 5, 2013 23:02		Folder	
▼ mathematical python	Today, 20:08		Folder	
▶ 🛅 bbob_pproc	March 5, 2013 23:03		Folder	
bbobbenchmarks.py	November 12, 2012 16:56	74 KB	Python script	
benchmarkinfos.txt	February 9, 2009 16:29	4 KB	Geditument	
exampleexperiment.py	February 22, 2013 14:26	4 KB	Python script	
exampletiming.py	November 12, 2012 16:56	4 KB	Python script	
fgeneric.py	March 3, 2013 19:33	25 KB	Python script	
LICENSE.txt	February 1, 2012 20:23	4 KB	Geditument	
README.txt	November 12, 2012 16:56	4 KB	Geditument	
▶ = r	March 5, 2013 23:05		Folder	*
Macintosh HD → 🔃 Users → 🏠 hansen → 🔯 Downloads → 🧰 bbob.v13.05 → 🧰 python → 🚞 bbob_pproc				

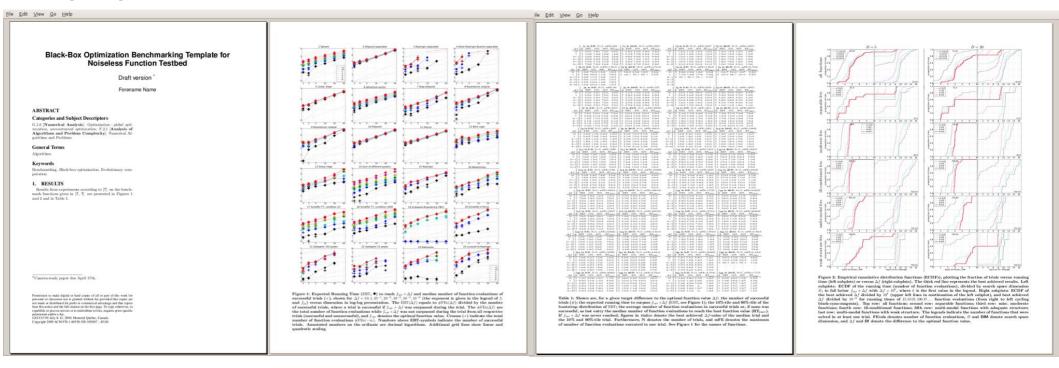
Post-processing at the OS shell:

```
$ python codepath/bbob_pproc/rungeneric.py datapath
```

[...]

\$ pdflatex templateACMarticle.tex

[...]



Black-Box Optimization Benchmarking Template for Noiseless Function Testbed

Draft version *

Forename Name

ABSTRACT

Categories and Subject Descriptors

G.1.6 [Numerical Analysis]: Optimization—global optimization, areometrained optimization; F.2.1 [Analysis of Algorithms and Problem Complexity]: Numerical Algorithms and Problems

General Terms

Algorithms

Keywords

Benchmarking, Black-box optimization, Evolutionary computation

1. RESULTS

Results from experiments according to [7] on the benchmark functions given in [7, 7] are presented in Figures 1 and 2 and in Table I.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without for provided that copies are not made or described for posti or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, to republish, so post on servers or to reduction to lists, requires grior specific permission anality a fee.

permission analor a for. GECCO'09; July 8-12, 2009, Montrial Québoc, Canada. Copyright 2009 ACM 978-1-60538-505-509-07 _S5.00.

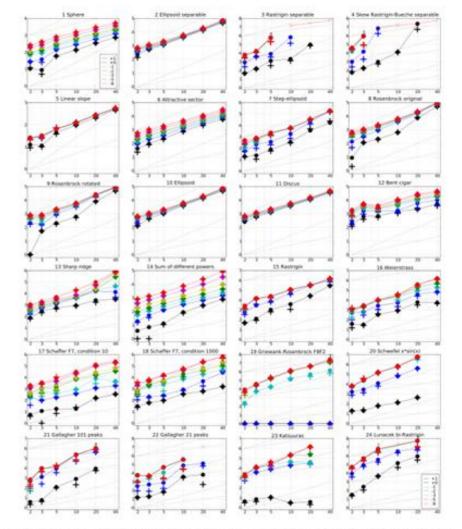


Figure 1: Expected Running Time (ERT, •) to reach f_{eqt} + Δf and median number of function evaluations of successful trials (+), shown for Δf = 10, 1, 10⁻¹, 10⁻¹, 10⁻¹, 10⁻¹, 10⁻¹, 10⁻¹ (the exponent is given in the legend of f_t and f_{fat}) versus dimension in log-log presentation. The ERT(Δf) equals to #FE₂(Δf) divided by the number of successful trials, where a trial is successful if f_{eqt} + Δf was surpassed during the trial. The #FE₂(Δf) are the total number of function evaluations while f_{eqt} + Δf was not surpassed during the trial from all respective trials (successful and unsuccessful), and f_{eqt} denotes the optimal function value. Crosses (×) indicate the total number of function evaluations #FE₂(−∞). Numbers above ERT-symbols indicate the number of successful trials. Annotated numbers on the ordinate are decimal logarithms. Additional grid lines show linear and quadratic scaling.

^{*}Camera-ready paper due April 17th.

			BYST-LEWIS CONTROL OF THE STREET
Ar of BET 103 000 BTs	of the lot bot by by	At # 1807 1005 1005 BT cars	# SHT 10% 20% 0Trace
10 13 6.0×1 4.8×1 7.3×1 6.0×	111 141 104 141 141	10 13 14-3 1.1-3 14-3 14-3	1 1 4c4 1 4c4 1 5c4 1 4c4
1 5 1449 1342 1442 144	6 T. R. OVE S. THE R. ROLE M. SHIP	1 3 3.5ct 1.2ct 1.5c2 1.5cb	5. J. Dr. S. Don S. Ave. S. Ave.
34-1 5 3246 5/042 5/46 5/46		N-1 5 15+8 1-3+8 16+2 15+2	5 1 444 1 544 1 744 1 444
10-3 A ANG ANG ANG ANG 10-3 A SAGE SAGE ANG SAGE		h-1 1 17e2 14e4 18e2 17e2	5 1.8+1 1.7+1 1.8+1 1.8+6 5 1.8+1 1.8+1 1.8+1 1.8+1
10-1 1 1402 1 102 1702 1 10 10-4 1 7 202 7 802 1 102 7 20	9 5 2003 2740 2003 2003		1 1 3e4 13e4 2.0e4 1.9e4
	Jam 20-D. Not. or Fine Steel		I to be being with appropriation
AT IN SHIP LOSS, NO. 1075.	S FROT 1070 0070, BT 1000	- AT US REF. 10% NOT WELLIAM	S ERY 1073 NOS STARS
		10 1 1 1 10 1 1 1 1 1 1 1 1 1 1 1 1 1 1	I BANT KANK DANT KANK
1 5 7.805 6505 1.504 7.605 2-14 5.505 1.505 1.605 1.505	th discrete special arrest	1 1 7 tol 1 and 57 of 7 and 5	D. SRIEF PR. C. Spinit B. Sell-
h-3/4 3/30 1/40 1/40 1/40	1523 35 II S	W-11	15 D3 TH 53 TT
8-5 4 5 Set 5344 1 846 1 345		b - b	
B-8 4 5.005 SOVE 1.600 1.605	And the opposite and the second	(a-4)	Act and the second second
At 8 60, Not. office)	A DET MAN AND STORY	At a last too, well are start	Fig. in 20-D. Nich, of the 1964.
Af # ENT 10% 00% BT.,	of H S Total S	AT # ERT 10% NOT BY 1001	# ERT ON NO. RT. NO.
1 5 6346 5.341 7.242 6.54	1 1 162 142 162 162	1 1 10-2 11-2 11-2 11-2	5 3.362 2340 3.763 3.340
Sec-11 5 0.541 5.341 7.041 0.54	4 1 5 2 7 2 2 5 42 2 5 43 2 7 7 2	hr-1 5 5.0+2 5.0+2 0.2+2 5.0+2	3 4.3e3 3.8e5 4.7e3 4.5e5
14-1 3 0.341 0.441 7.44 0.34			5 6 345 5.840 6.743 6.348
10-1 A 0.341 0.841 7.841 0.8 10-1 5 0.341 0.841 7.041 0.8		h-5 5 1,0x3 1,0x3 1,4x3 1,5x3 h-4 5 1,0x3 1,7x3 1,0x3 1,0x3	5 S. Long V. Cod S. South St. Long
I ft in Sep. Nach. williams		/s to delb, Not, of Bollett	for the 20-40, No. 5, softengalet
AT DE RET. 1070 1070 RTu-	of R. WHY NOW MOVE BY AND THE	AJ R RET. 10% 10% RTown	at REST 1075 NOTS STrange
30 [A 7.641 3.441 1.040 7.64	6 1 5 5 5 c 5 2 c 6 c 5 4 5 c 5 5 5 c 5	10 55 S. S. See S. Polic S. Aug. S. See	5. T.549. 6.Te3 T.948. T.849.
1. [5	B. I.A. Willed L. Rott. Ballet . B. Do. 4	1 5 King Tiles Link King	5 1-7 of 1-7 of 1-5 of 1-5 of
h-1 1 104 114 214 114 114 h-2 1 114 114 114 114		h-1 5 1.463 1.56 1.763 1.463 h-1 5 1.863 1.563 2.063 1.863	5 1.7e4 1.5e4 1.9e8 3.7e4 5 1.9e4 1.8e4 2.0e8 4.0e4
h -5 5 2 1st 1.5s3 2.7s1 2.1s	A Alest Direct Adopt Shiet	2c-1 5 2.0c3 1.7c3 2.2c3 2.0c3	5 2.0ml 1.6ml 2.0ml 2.0ml
h 14 2 2.54 1.742 3.045 2.34	5 1 32c4 27c6 18c4 2.2c4		5 E.let 2016 2 lot 2 lot
[/a to 5-0. No.h. of No.407	To in 90-D. 20-5. inFile 60010	foote 6-th, history	Fac in 20-D. Not. of No. 10030
A/ 2 EAT 100 900 EF.	2 ERT 10'S 90'S RT 10'S	A7 8 8NY 100 000 8F and 10 5 1003	S PART LON SON BY
1 2 2012 7 642 1 643 8 644		1 5 13-0 13-0 14-0 14-0 13-0	5. 1.444 1.544 1.544 1.544 5. 1.544 1.544 1.544 1.544
h-1 5 1.4d 1.5d 1.5d 1.4d			5 1.744 1844 1.744 1.744
h-3 2-1.8c2 1.7ct 1:3c3 1.8cd	5. E-4e4 1.0e4 3.0e8 2.4e4		2 Liket Act Liket Liket
\$6 - 5 5 2.002 1.000 E.563 E.563			5 1.4×6 1.6×4 1.0×4 1.4×4
h-4 5 8.002 8.102 8.502 9.302			5 1/544 1/544 1/544 1/544
A/ # ERT 10% SUS MEANS	fig in 20-D. Not, militariotes	Af R ERT SON WITH PERSON	Fig. in 20-D. Not. officettly of SET and SET Miles
A/ 8 ERT 10% NOS ME-100 10 3 13-0 13-0 13-0 13-0 13-3	S ERT 10% 90% RY-00%	Af 2 EHT 100 Mch Brown	2 EST 201 MAN STORE
1 1 1 1 4 4 1 1 24 2 1 24 3 1 24 4 1	S. S.Rob Elled S.Rod Like E.	1 3. 1 dect 1.5ut 1.5ub 1.6eb	3. 3.0x4 3.3x6 3.3x5 3.4x4
10-1 5 1603 1.5c3 1.7c3 1.6c3 10-2 5 1802 1.7c3 1.8c7 1.4c3	5 1344 1344 1344 1344 5 1444 1344 1444 1444	h - 1 A 9.303 1.705 8.605 9.205 h - 2 5 9.605 8.105 9.705 9.305	5 1744 7.140 1344 1.144 5 1744 1.144 1854 1.744
5-1 1 1840 1741 1847 1843 5-1 1 2041 1841 2041 2041	5 1.4st 1.5st 1.4st 1.4st 5 1.5st 1.4st 1.5st 1.5st	h-1 5 3 h-2 2 0-2 4 h-3 3 3-2	5 2304 1 204 2 det . 2 208
to-8 5 9205 2.163 2363 - 2463			
		h-9 5 3.942 2.945 5.942 3.942	5 2844 2344 3344 2344
Triale 5-0, Not settlem	Jan to posts, Nach, softwareness	I fag to 6-D, Work or Physical	I find by 20-th, No.5, or Workship
Af a cur see see are	Jan in 1980. Nath of the beaute	At g say you see are	I find by 20-th, No.5, or Workship
A/ # 607 105 005 0701700 00 5 550 1501 7201 1342	fig to \$0-D. Not. of Woldships g EUF LOT Solt STreet LASS LASS LASS LASS	At # 600 Not see By 200 10 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1	
A/ # 607 105 005 0701700 00 5 550 1501 7201 1342	Jan in 1980. Nath of the beaute	At g say you see are	I find by 20-th, No.5, or Workship
Af p sur int son stricts Af p sur int son stricts 1 1 1.24 1.34 1.34 1.44 1.34 1 1 1.44 1.34 1.34 1.44 1.34 i-1 2 1.44 1.34 1.34 1.34 i-1 2 1.44 1.34 1.34 1.34 1.34	fig to \$00-D. No.h. on Fig. 1000000 gr SHF tork onth AF rect 0 5.75 1.75 1.50 1.50 1.50 1.50 0 1.50 1.50 1.50 1.50 1.50 1.50 0 1.50 1.50 1.50 1.50 1.50 1.50 1.50 1.5	### 15 1.5	Fig 00 40 - No. 00 No. 10 No.
712 bs 5-D. No.5. of	fig to 20-0. Note of the images graph land dark defined 5 2702 1702 2-00 2005 5 1.004 1004 2-00 1004 5 2.004 1004 2-00 1004 5 2.004 1704 2-00 2-00 5 2.004 1704 2704 2504 5 2.004 1704 2704 2504 5 2.004 1704 2704 2504	## 86 6-D. No. 1. arPin 2004 ## 88 8 100 100 100 100 100 100 100 100 10	fig be 20-10, No.5, orf Ecosetti graph art art
fab is 50, Nu.5, nd3, if 30, 100, 100, 100, 100, 100, 100, 100,	fig to 20-45, No.5, as Fincients Fine Property Section Sec	\$\begin{array}{cccccccccccccccccccccccccccccccccccc	Fig. 10 20-45 Xinh. of Hispanish
Tin be 5-D. Not. of Notice	fig. 80-45, Mark, self-minimum mark ma	$ \begin{array}{c ccccccccccccccccccccccccccccccccccc$	Fig. 10 20-45, No.5, or World (1) Fig. 10 20-45, No.5, or World (1) Fig. 12 20-5, No.5, Ord (1) Fig. 12 20-5, Ord (1) Fi
Fig. 10 - 50, No. 5, orFile 1740 A	Fig. 10 20-10, 70-11, 10-70-100-100-100-100-100-100-100-100-10		Fat 10 - 20 - 10 No. 1. art No. 2 + 10
Fig. bs. 5-D, Nucl., articularly	Fig. 8: 26-10, No. 1, ac Fin commun. 2. ENT 10: 60% ST Print; 3. ENT 10: 60% ST Print; 3. ENT 10: 60% ST Print; 5. ENT 10: 60% ST Print; 6. ENT	### ### ### ### #### #################	Fat 10 20 - 10 No. 1. art No. 10 No. 1. art No. 10 No. 1. art No. 10 No. 1 No. 10 No
Fig. bs. 50, Nucl., astronome 1	Fa No. 100-10, No. 1, No. 100-100-100-100-100-100-100-100-100-100	fat he b-D. Sinch as Philadelle fat he b-D. Sinch as	Fat 10 - 10 - 10 No. 1. arXiv:24 kill Fat 10 - 10 No. 1. arXiv:25 kill Fat No. 1. arXiv:25 kill Fat 10 - 10 No. 1. arXiv:25 kill Fat 10 - 10 No. 1. arXiv:25 kill No. 25 kill
Fig. bs. 5-D, Nucl., articularly	Fig. 8: 26-10, No. 1, ac Fin commun. 2. ENT 10: 60% ST Print; 3. ENT 10: 60% ST Print; 3. ENT 10: 60% ST Print; 5. ENT 10: 60% ST Print; 6. ENT	### ### ### ### #### #################	Fat 10 20 - 10 No. 1. art No. 10 No. 1. art No. 10 No. 1. art No. 10 No. 1 No. 10 No
Fig. bs. 5-D, Nucl., netter 1784	Fig. 10-10, No.5, ar Fig. 10-10, Fig. 10-10, No.5, ar Fig. 10-10, Reg. 20, Reg	fat he 5-D, brick as Philadelle de de de de de de de d	Fat 10 - 20 - 10 No. 1. art No. 2 + 10
Fig. bs. 5-D, Nucl., activation	Fig. 10 10 10 11 11 11 11 11	fat 6-50 such as Fin fat 6-50 such as 6	Fat 10 20 10 10 10 10 10 10
Fig. bs. 5-D, Nucl., activation	Fa 10 10 10 10 10 10 10 1	fat 6-50 such as Fin fat 6-50 such as 6	
Fig. bs. 5-D, Nucl., article Principle Principle	F ₂ is 19-10. No.5. No.7. No.700 beams F ₂ 1977 100 1	fat is 5-D. Such as Fill about April Apr	
Fig. bs. 5-D, Nucl., article	Fa	fat No. 1-0. No. 1- N	
Fig. bs - 50, Nucl., activation	Fig. 10 10 10 11 11 12 12 13 14 14 15 14 15 15 15 15	fat is 0-0. Such as Plus 1900	Fat 10 - 20 - 10 No. 1 No. 10 + 10
Fig. bs -60, Nucl., asPrint284	Fa 10 10 10 10 10 10 10 1	fig. 6-50, tech as Fig. 1	
Fig. bs. 5-D, Nucl., article Prop. Prop.	Fig. 10 10 10 11 11 12 12 13 14 15 15 15 15 15 15 15	fat is 0-D. Such as Plus 1940	Fat 10 - 20 - 10 No. 1 Fit 10 - 10 No. 1 Fit 10 - 10 No. 1 Fit 10 - 10 No. 1
Fig. bs. 5-D, Nucl., activation	Fat 10 10 10 10 10 10 10 1	faq is 5-D. Such as Fill about April Apr	Fat 10 20 10 10 10 10 10 10
Fig. bs. 5-O. Nucl., activation	Fa 10 10 10 10 10 10 10 1	fig. 6-50, with a FF	Fat 10 20 10 10 10 10 10 10
Fig. bs -60, Nucl., activation	Fat 10 10 10 10 10 10 10 1	fig. 16 - 5-0 tenth of Fig. 18 - 5-0 tenth	
Fig. bs. 5-O. Nucl., activation	Fa 10 10 10 10 10 10 10 1	fat is 0-0. Such as First 1884	Fat 10 20 10 10 10 10 10 10
Fig. bs -50, Nucl., activation	Fig. 10 10 10 10 10 10 10 10	fig. 6-50, tech as Fig. 1 fig. 6-50, tech as Fig. 6-50, te	Fat 10 - 20 - 10 No. 1 Fit 10 - 10 No. 1 Fit 10 - 10 No. 1 Fit 10 - 10 No. 1
Fig. 10. 5-D. Nucl., activation	Fig. 10 10 10 10 10 10 10 10	fat is 5-D. Strike in Fluctuation A	Fat 10 - 20 - 10 No. 1. art No. 2 + 10
Fig. 16 - 6-0, Nucl., activation	Fat 10 10 10 10 10 10 10 1	fat is 5-D. Sinch as Fill about A	Fat 10 - 20 - 10 No. 1
Fig. to 5-D, Nucl., activation	Fig. 10 10 10 10 10 10 10 10	faq is 6-0. Such as Fill about A A A A A A A A A	Fat 10 20 40 10 10 10 10 10 10 1
Fig. 16 - 6-0, Nucl., activation	Fat 10 10 10 10 10 10 10 1	fat is 5-D. Sinch as Fill about A	Fat 10 - 20 - 10 No. 1
Fig. 10. 5-D. Nucl., activation	Fig. 10 10 10 10 10 10 10 10	fat is 5-D. Sinh a Plus 1990	
Fig. 16 - 6-0, Nucl., activation	Fat 10 10 10 10 10 10 10 1	Fig. 18 - 5-D. Str. b. arthur 1970 1840	Fat 10 20 10 10 10 10 10 10
Fig. 1s. 5-O. Nucl., activation	Fig. 10 10 10 11 10 10 10 10	Fig. 18 - 5-D. Str. b. art 1-1	Fat 10 20 10 10 10 10 10 10
Fig. 16 - 6-0, Nucl. of Fig. 17	Fat 10 10 10 10 10 10 10 1	fat is 6-0.0 tenth of Final State Add	Fat to 20-0-1, No. 1, art No. 20-01
Fig. 10. 5-0. Nucl., activation	Fat 10 10 10 10 10 10 10 1	fat is 5-0, with a Plus 1990	Fat 10 20 10 10 10 10 10 10
Fig. 16 - 6-0, Nucl. article Fig. 2 Fig. 2	Fall 10 10 10 10 10 10 10	Fig. 16 - 5-0 Such as Floridation April	Fat bot Bot State bot Bo
Fig. 10. 5-0. Nucl., activation	Fig. 10 Fig. 11 Fig. 12 Fig. 12 Fig. 12 Fig. 13 Fig.	Fig. 6: 5-D. Str. by Physics April 1999 1	Fat 10 20 10 10 10 10 10 10
Fig. 16 - 60, Nucl. activation	Fig. 10 Fig. 11 Fig. 12 Fig. 12 Fig. 13 Fig. 13 Fig. 13 Fig. 14 Fig.	Fig. 18 - 5-D. Str. b. arthur arthu	Fat to 20-40, No.11, and No.19 to
Fig. 16 - 60, Nucl. activation	Fig. 10 Fig. 11 Fig. 12 Fig. 12 Fig. 13 Fig. 13 Fig. 13 Fig. 14 Fig.	Fig. 18 - 5-D. Str. b. arthur arthu	Fat 10 20 10 10 10 10 10 10

Table 1: Shown are, for a given target difference to the optimal function value Δf : the number of successful trials (#); the expected running time to surpass $f_{eqs} + \Delta f$ (ERT, see Figure 1); the 10%-tile and 90%-tile of the bootstrap distribution of ERT; the average number of function evaluations in successful trials or, if none was successful, as last entry the median number of function evaluations to reach the best function value (RT_{succ}). If $f_{eqs} + \Delta f$ was never reached, figures in italics denote the best achieved Δf -value of the median trial and the 10% and 90%-tile trial. Furthermore, N denotes the number of trials, and mFE denotes the maximum of number of function evaluations executed in one trial. See Figure 1 for the names of functions.

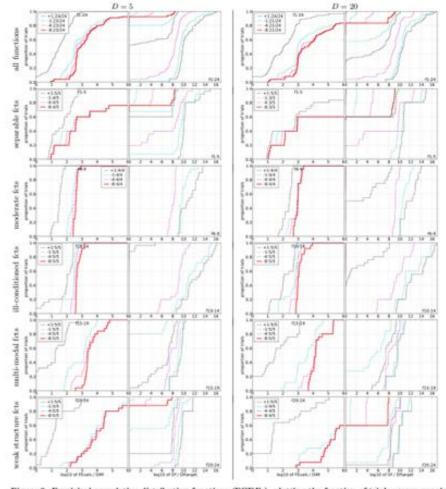


Figure 2: Empirical cumulative distribution functions (ECDFs), plotting the fraction of trials versus running time (left subplots) or versus Δf (right subplots). The thick red line represents the best achieved results. Left subplots: ECDF of the running time (number of function evaluations), divided by search space dimension D, to fall below $f_{\rm ep} + \Delta f$ with $\Delta f = 10^3$, where k is the first value in the legend. Right subplots: ECDF of the best achieved Δf divided by 10^3 (upper left lines in continuation of the left subplot), and best achieved Δf divided by 10^{-4} for running times of D, $10\,D$, $10\,D$. In function evaluations (from right to left cycling black-cyan-magenta). Top row: all functions; second row: separable functions; third row: misc. moderate functions; fourth row: ill-conditioned functions; fifth row: multi-modal functions with weak structure. The legends indicate the number of functions that were solved in at least one trial. FEvals denotes number of function evaluations, D and DM denote search space dimension, and Δf and Df denote the difference to the optimal function value.

Test Functions

Test Functions

Test functions

define the "scientific question"

the relevance can hardly be overestimated

- should represent "reality"
- are often too simple?

remind separability

a number of testbeds are around

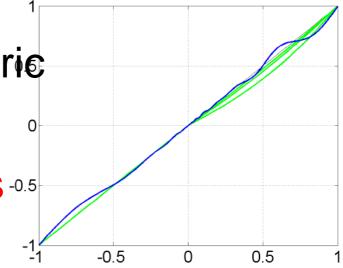
BBOB: the noiseless functions

functions are not perfectly symmetric and are locally deformed





- Essential unimodal functions
- III-conditioned unimodal functions
- Multimodal structured functions
- •Multimodal functions with weak or without structure



ı		arable functions
	1.1	Sphere Function
	1.2	Ellipsoidal Function
	1.3	Rastrigin Function
	1.4	
	1.5	Linear Slope
2	Fun	ctions with low or moderate conditioning
	2.6	Attractive Sector Function
	2.7	Step Ellipsoidal Function
	2.8	Rosenbrock Function, original
	2.9	Rosenbrock Function, rotated
3	Fun	ctions with high conditioning and unimodal
	3.10	Ellipsoidal Function
	3.11	Discus Function
	3.12	Bent Cigar Function
		Sharp Ridge Function
		Different Powers Function
4	Mul	ti-modal functions with adequate global structure
		Rastrigin Function
		Weierstrass Function
		Schaffers F7 Function
		Schaffers F7 Function, moderately ill-conditioned
		Composite Griewank-Rosenbrock Function F8F2
5	Mul	ti-modal functions with weak global structure
	5.20	Schwefel Function
		Gallagher's Gaussian 101-me Peaks Function
		Gallagher's Gaussian 21-hi Peaks Function
		Katsuura Function
		Lunacek bi-Rastrigin Function

GECCO-BBOB

Test Functions

http://coco.gforge.inria.fr

Submitted Data Sets

- 2009: 31 noiseless and 21 noisy "data sets"
- 2010: 24 noiseless and 16 noisy "data sets"
- 2012: 30 noiseless and 4 noisy "data sets"
- 2013: 31 noiseless "data sets"
- 2015: 26 noiseless "data sets"
- Algorithms: RCGAs (e.g. plain, PCX), EDAs (e.g. IDEA), BFGS, NEWUAO, Simplex & (many) other "classical" methods, ESs (e.g. CMA), PSO, DE, Memetic Alg, Ant-Stigmergy Alg, Bee Colony, EGS, SPSA, Meta-Strategies...

How do we measure performance?

Evaluation of Search Algorithms

Behind the scene

a performance measure should be

- quantitative on the ratio scale (highest possible)
 - "algorithm A is two times better than algorithm B" is a meaningful statement
- assume a wide range of values
- meaningful (interpretable) with regard to the real world possible to transfer from benchmarking to real world

runtime or first hitting time is the prime candidate (we don't have many choices anyway)

Measuring Performance

...empirically...

convergence graphs is all we have to start with

(recall) Black-Box Optimization

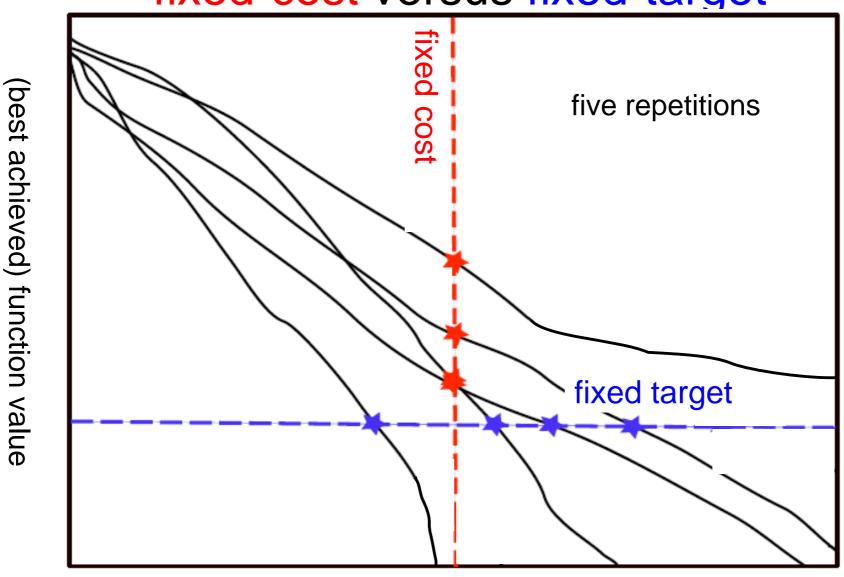
Two objectives:

•Find solution with small(est possible) function value

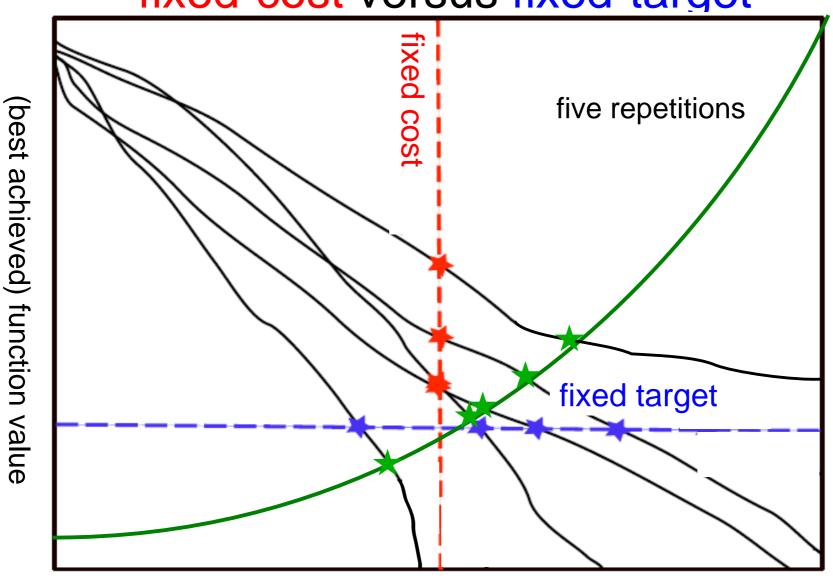
•With the least possible search costs (number of function evaluations)

•For measuring performance: fix one and measure the other

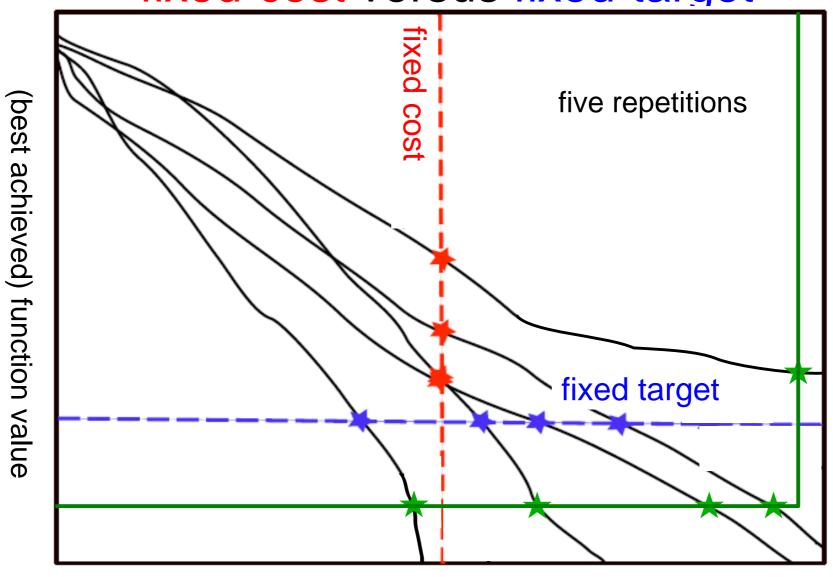
fixed-cost versus fixed-target



fixed-cost versus fixed-target



fixed-cost versus fixed-target



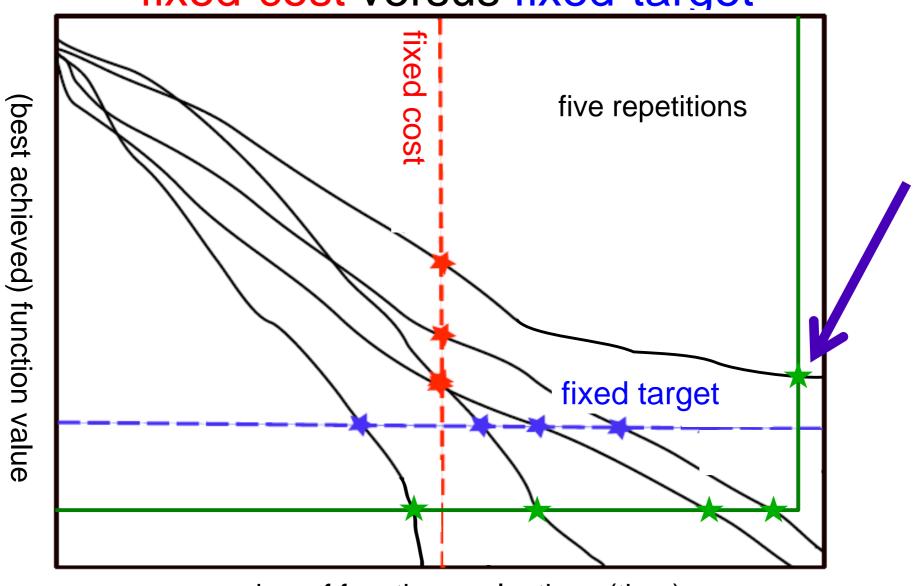
The performance measure we use

Run length or runtime or first hitting time to a given target function value measured in number of fitness function evaluations

equivalent to first hitting time of a sublevel set in search space

How can we deal with "missing values"?

fixed-cost versus fixed-target



Fixed-target: Measuring Runtime

- 1. Fix a target f-value (most difficult part)
- 2. Compute the success rate \hat{p} as

$$\widehat{p} = \frac{\text{\# of successful runs (that reached the target)}}{\text{\# of all runs}} \in [0,1]$$

$$\widehat{R} = \frac{1-\widehat{p}}{\widehat{p}} = \frac{\text{\# of unsuccessful runs}}{\text{\# of successful runs}} \in [0,\infty]$$

 \widehat{R} is the odds ratio to be unsuccessful

 \widehat{R} is the number of unsuccessful runs observed for each single successful run (i.e. normalized by # of successful runs)

Fixed-target: Measuring Runtime

$$\widehat{R} = \frac{1 - \widehat{p}}{\widehat{p}} = \frac{\text{\# of unsuccessful runs}}{\text{\# of successful runs}} \in [0, \infty]$$

 \widehat{R} is the odds ratio to be unsuccessful

 \widehat{R} is the number of unsuccessful runs observed for each single successful run (i.e. normalized by # of successful runs)

3. Compute "expected runtime" to hit the target

average runtime for a single successful run

$$\overline{\mathsf{ERT}} := \overline{\mathsf{RT}}_{\mathrm{succ}} + \widehat{R} \times \overline{\mathsf{RT}}_{\mathrm{unsucc}}$$

average runtime spent in unsuccessful runs to achieve one successful run

$$\mathrm{SP1} := \overline{\mathsf{RT}}_\mathrm{succ} + \widehat{R} imes \overline{\mathsf{RT}}_\mathrm{succ} = \overline{\mathsf{RT}}_\mathrm{succ} (1 + \widehat{R})$$
 disregarding runlength of unsuccessful runs

if
$$\widehat{R} < \infty$$
, else we can assume $\mathsf{ERT} \geq \sum \mathsf{RT}_{unsucc}$

Fixed-target: Measuring Runtime

 \widehat{R} is the number of unsuccessful runs observed for each single successful run (i.e. normalized by # of successful runs)

3. Compute "expected runtime" to hit the target

average runtime for a single successful run

$$\overline{\mathsf{ERT}} := \overline{\mathsf{RT}}_{\mathrm{succ}} + \widehat{R} \times \overline{\mathsf{RT}}_{\mathrm{unsucc}}$$

average runtime spent in unsuccessful runs to achieve one successful run

$$\mathrm{SP1} := \overline{\mathsf{RT}}_\mathrm{succ} + \widehat{R} imes \overline{\mathsf{RT}}_\mathrm{succ} = \overline{\mathsf{RT}}_\mathrm{succ} (1 + \widehat{R})$$
 disregarding runlength of unsuccessful runs

We can simulate a single runtime by "restarting" until the first success

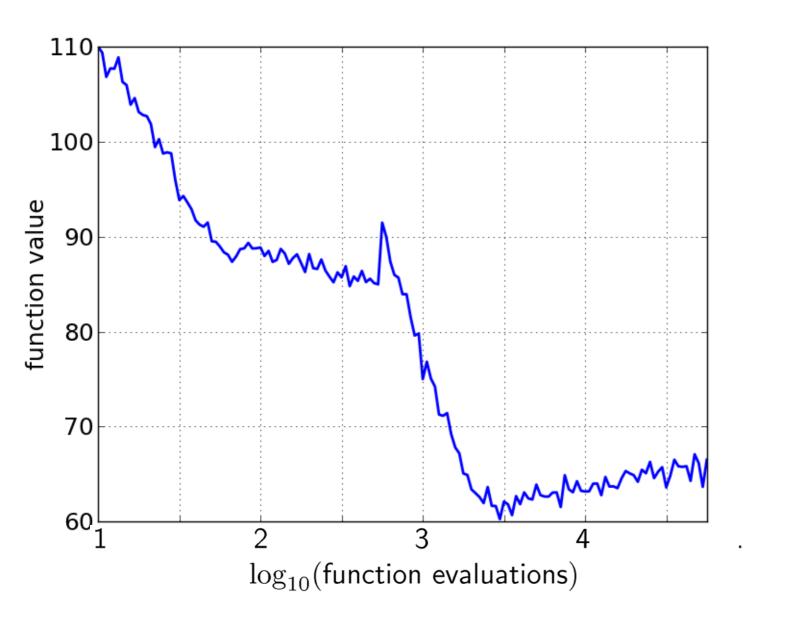
$$RT = RT_{\rm succ} + \sum RT_{\rm unsucc}$$

- ⇒ distribution of runtimes incorporating unsuccessful runs
- ⇒ display the distribution or a statistic of it

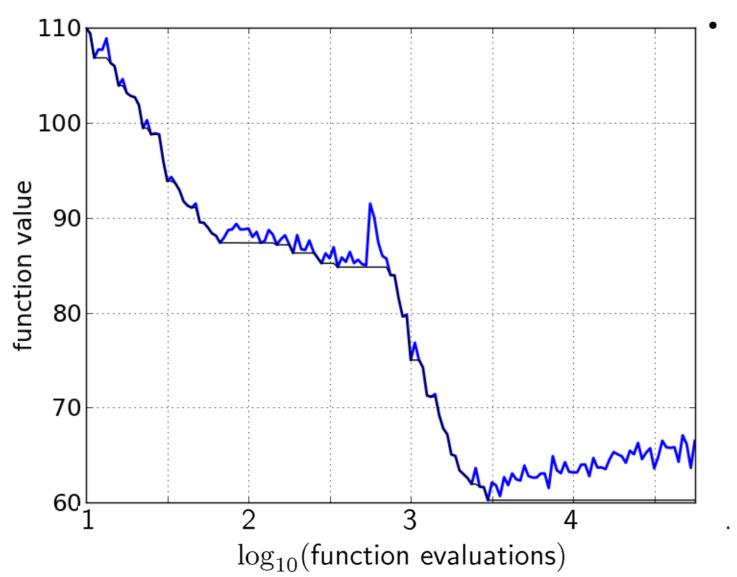
ECDF:

Empirical Cumulative Distribution Function of the Runtime

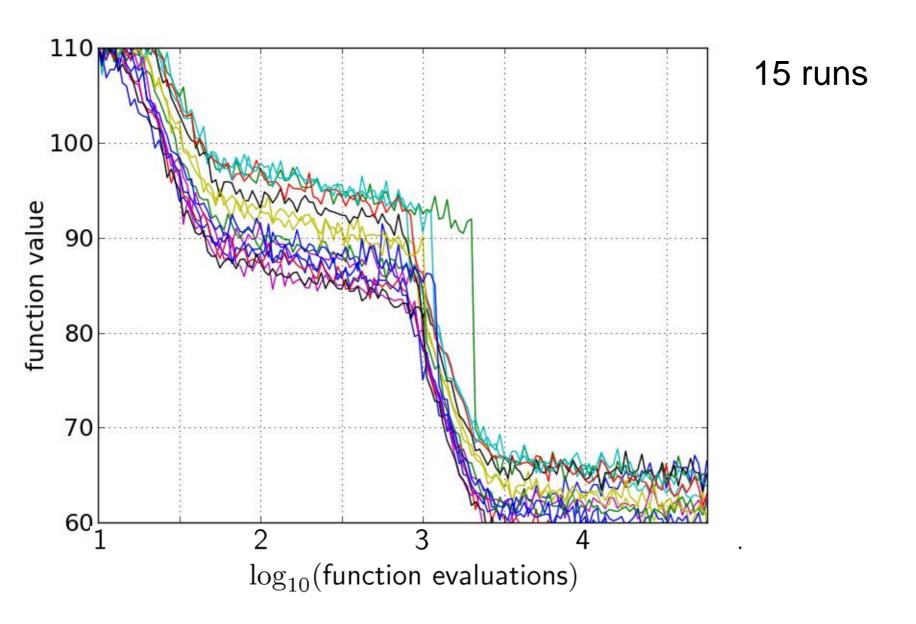
A Convergence Graph

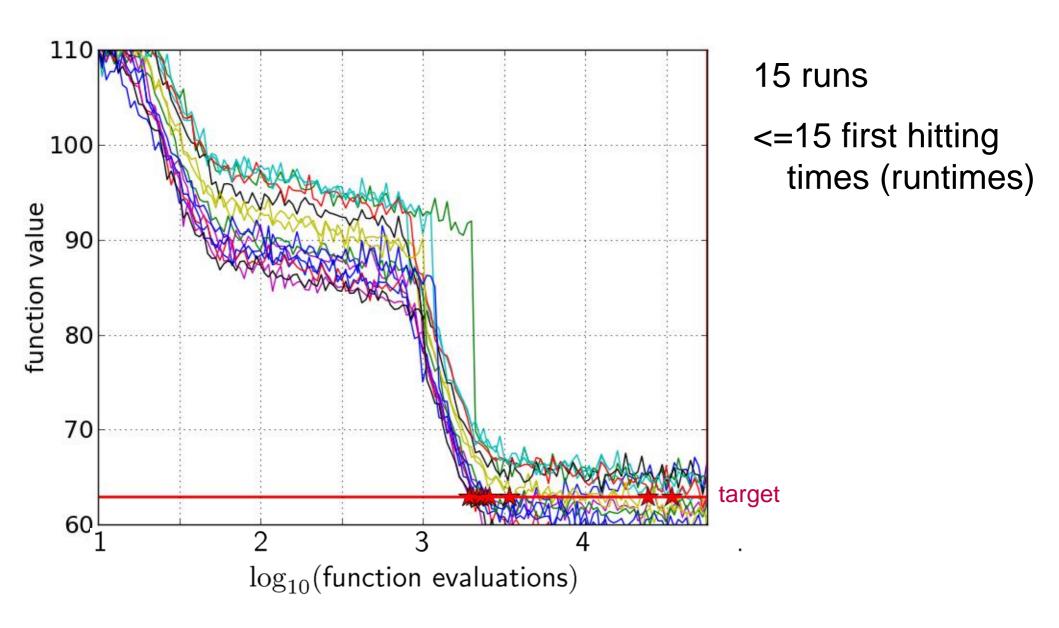


First hitting time is monotonous

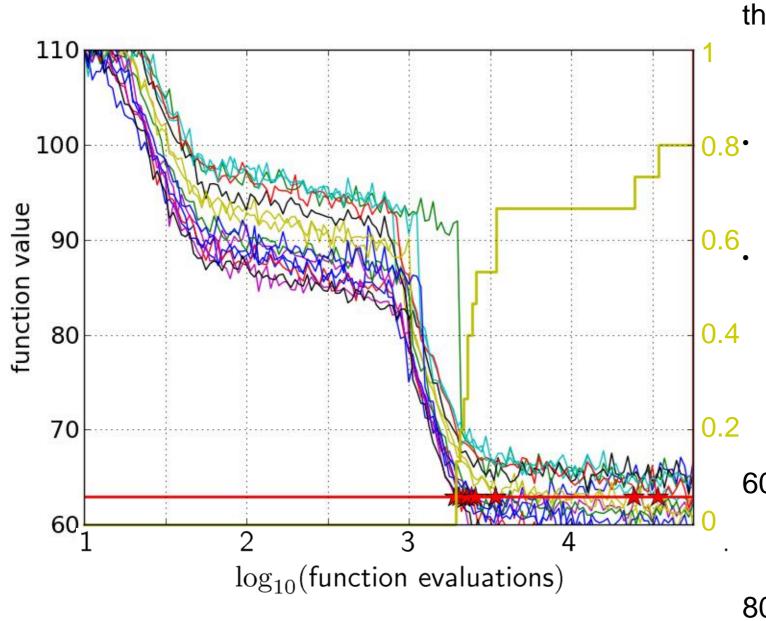


first hitting time: a monotonous graph





Empirical CDF



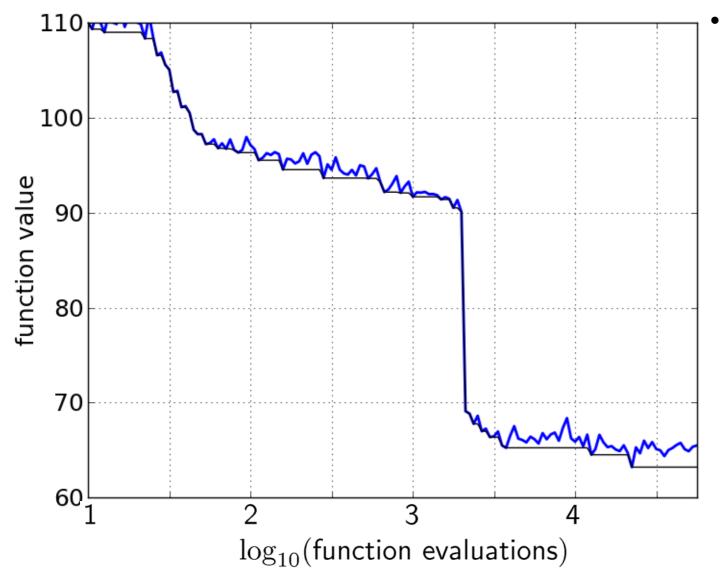
the ECDF of run lengths (runtimes) to reach the target

> has for each data point a vertical step of constant size

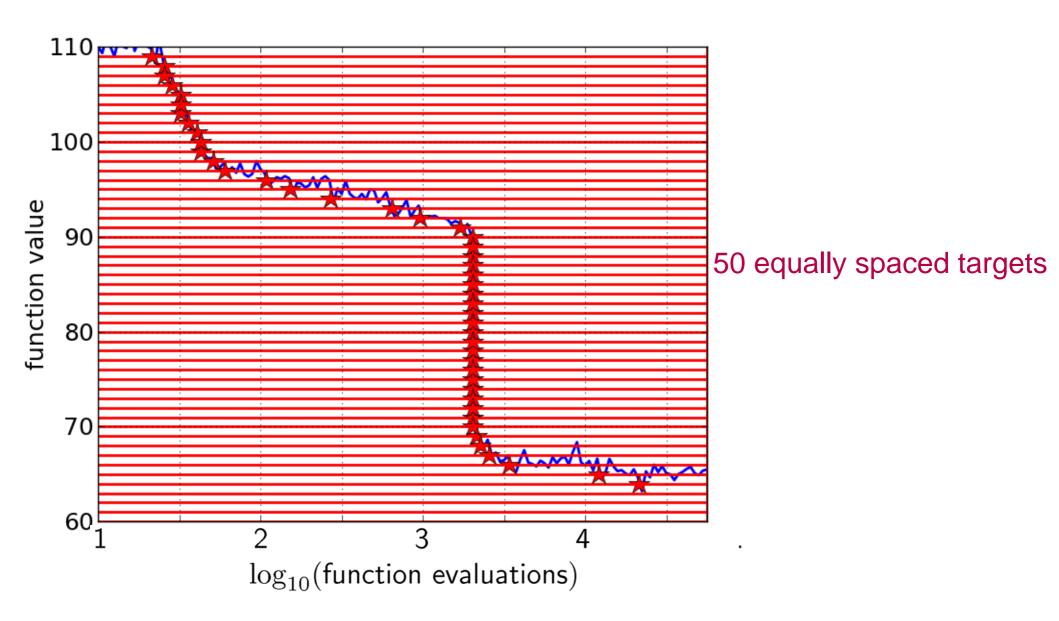
displays for each x-value (budget) the count of observations to the left (first hitting times)

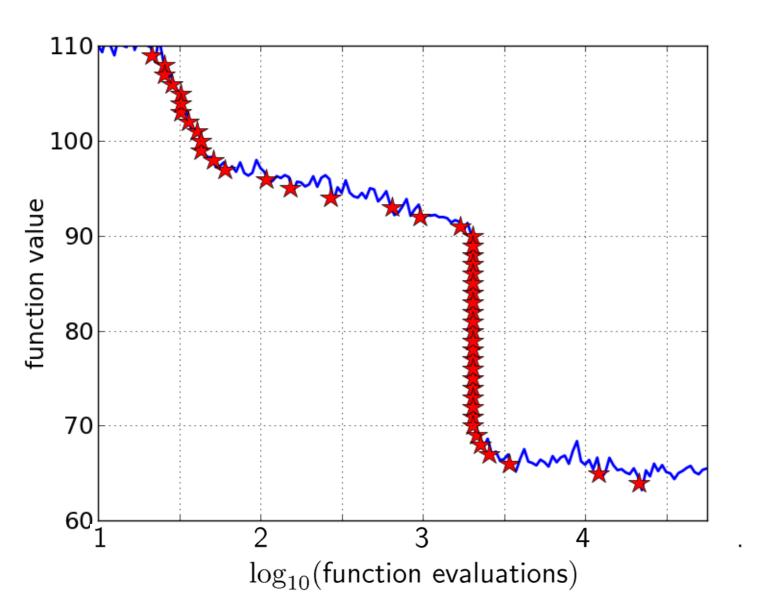
60% of the runs need between 2000 and 4000 evaluations

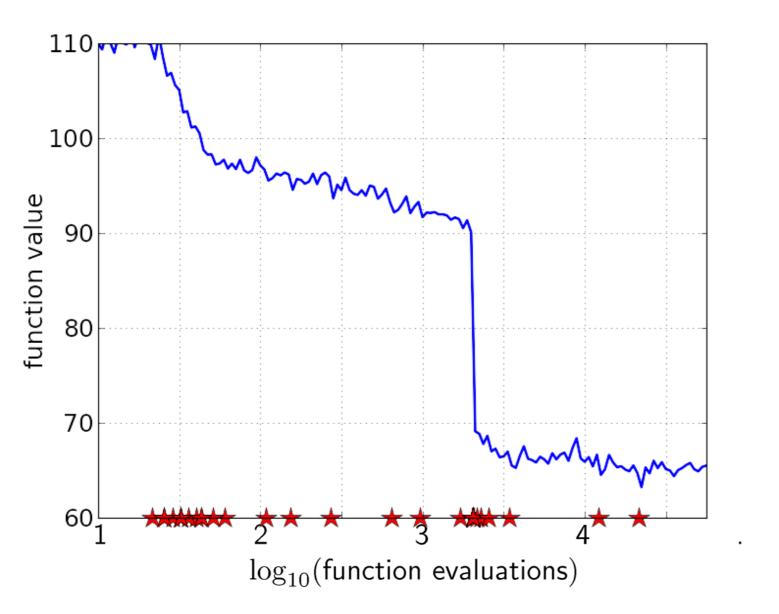
80% of the runs reached the target

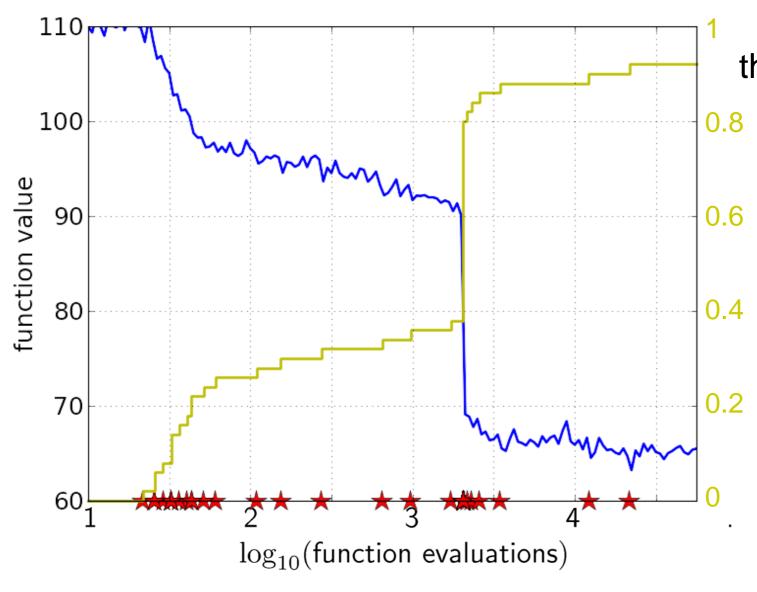


reconstructing a single run using different target values



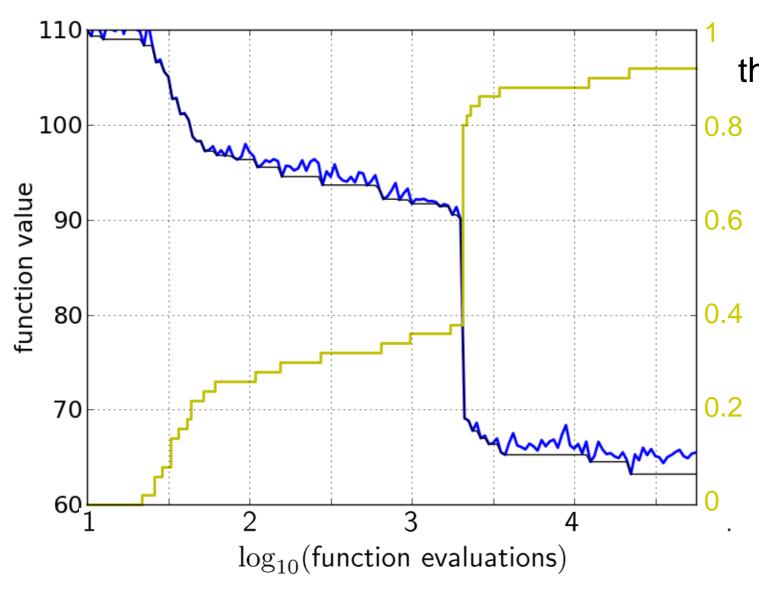




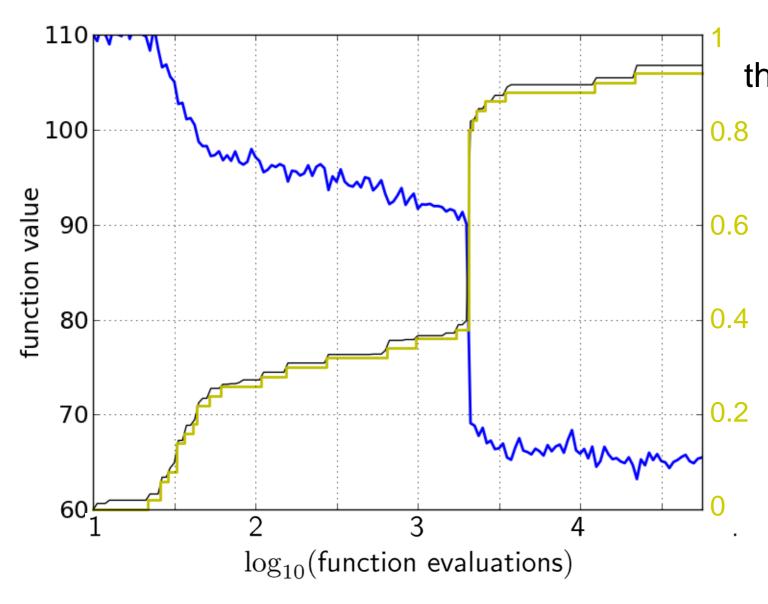


the empirical CDF

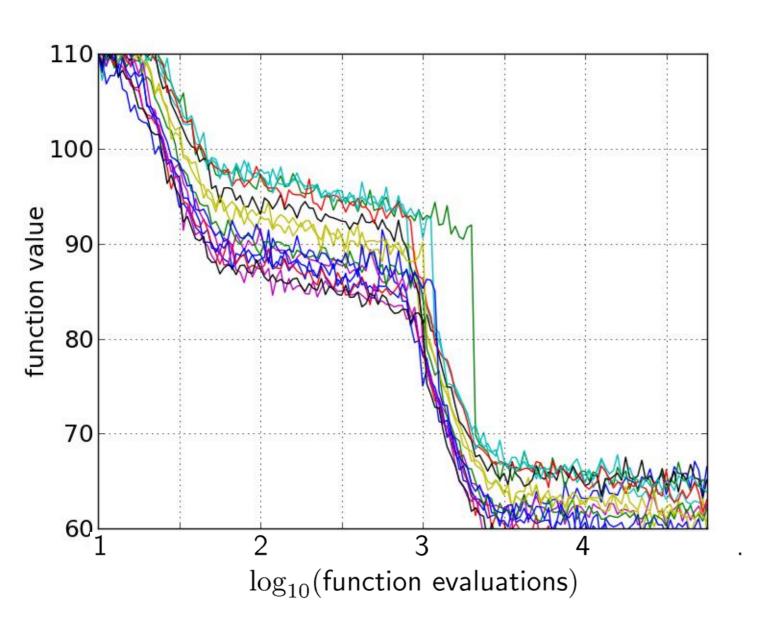
makes a step for each star, is monotonous and displays for each budget the fraction of targets achieved within the budget



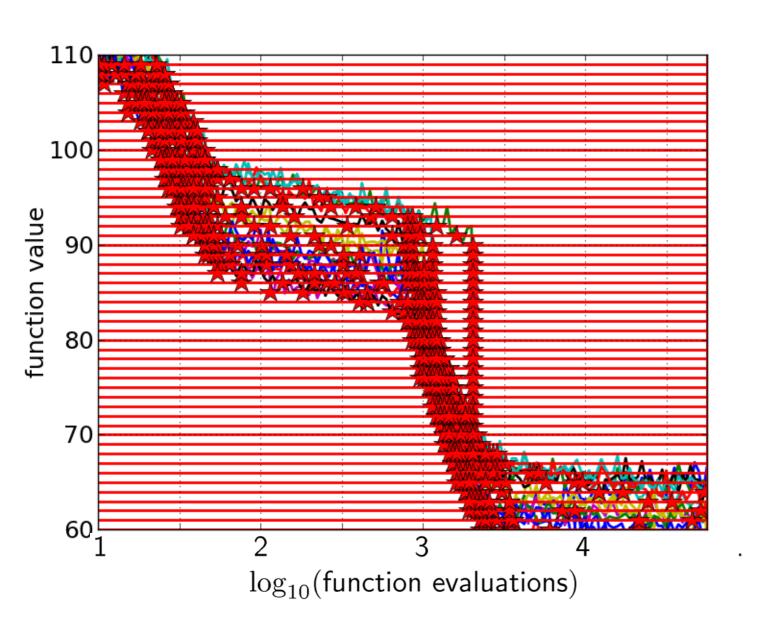
the ECDF recovers
the monotonous
graph,
discretised and
flipped



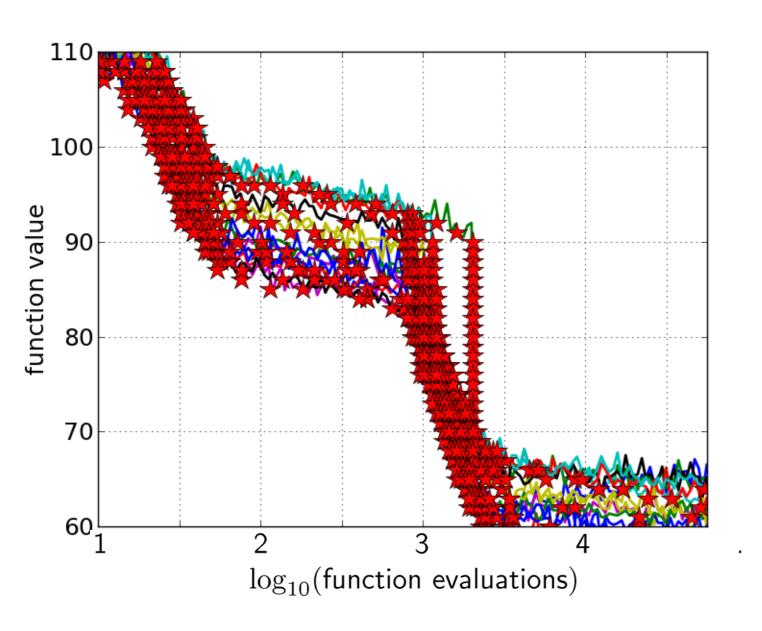
the ECDF recovers
the monotonous
graph,
discretised and
flipped



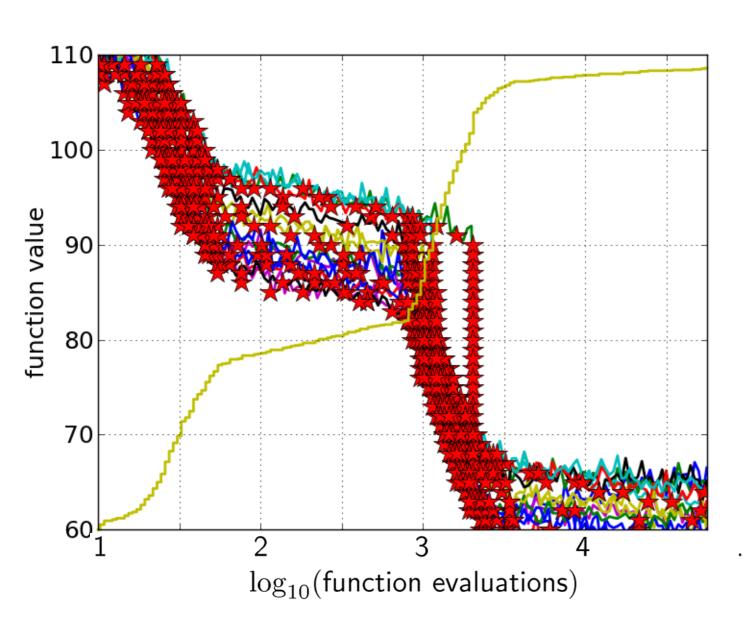
15 runs



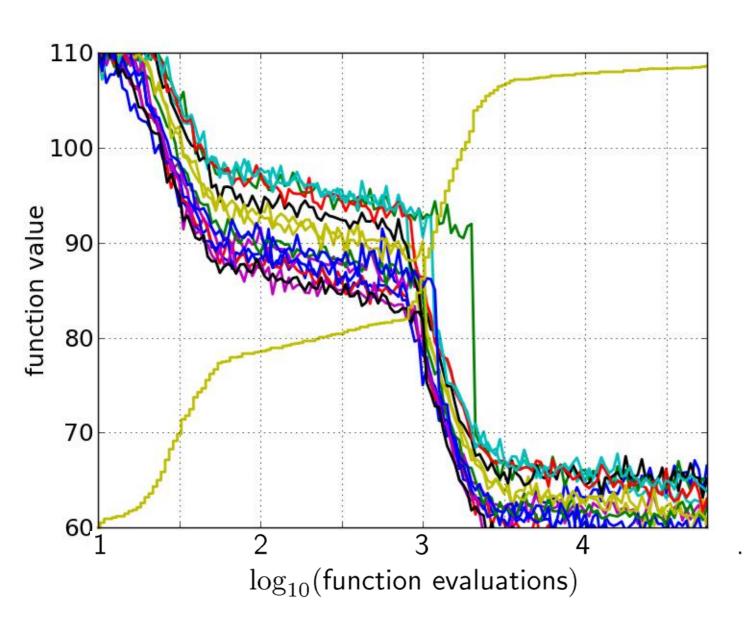
15 runs50 targets



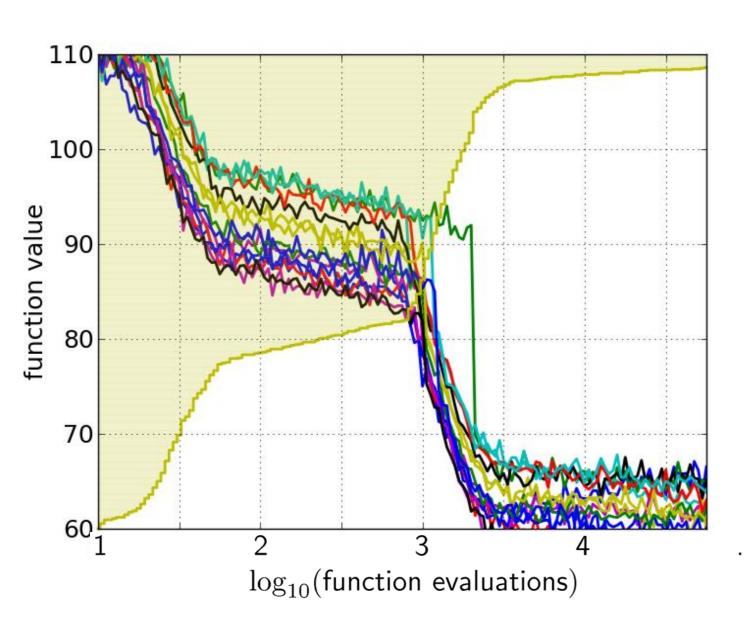
15 runs50 targets



15 runs50 targetsECDF with 750 steps

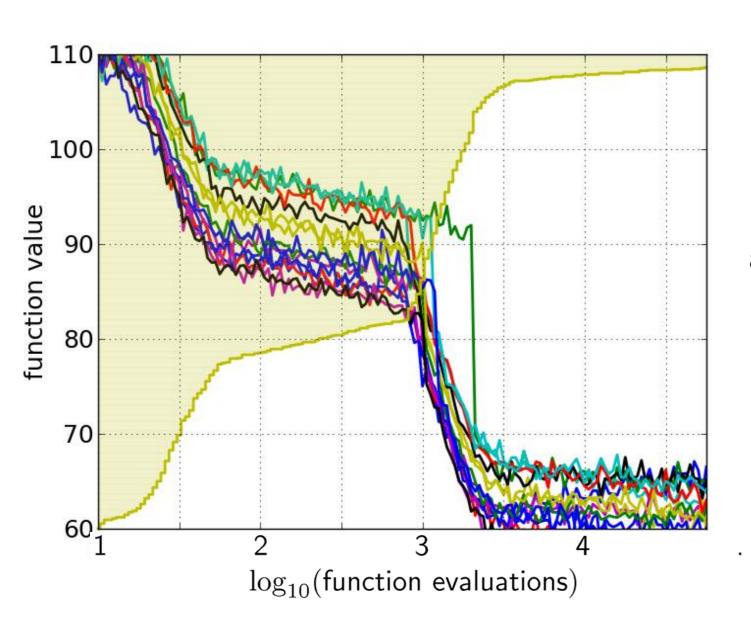


50 targets from 15 runs integrated in a single graph



50 targets from 15 runs integrated in a single graph

the area over the **ECDF** curve is the average log runtime (or geometric average runtime) over all targets (difficult and easy) and all runs

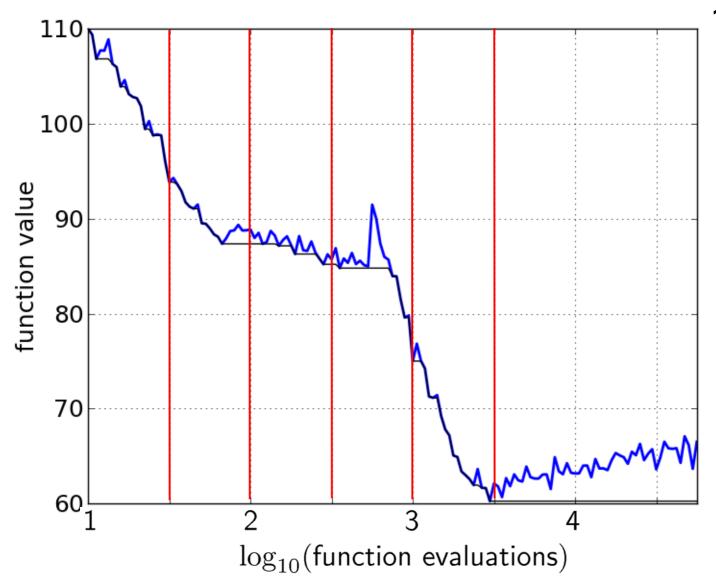


50 targets from 15 runs integrated in a single graph

a shift of the empirical CDF to the left means a speed up by a certain factor

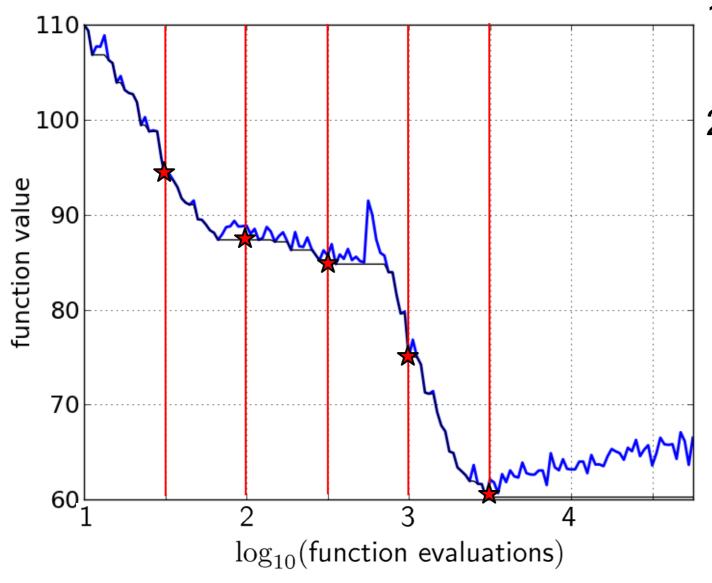
Runlength-based Target Values

Target budgets



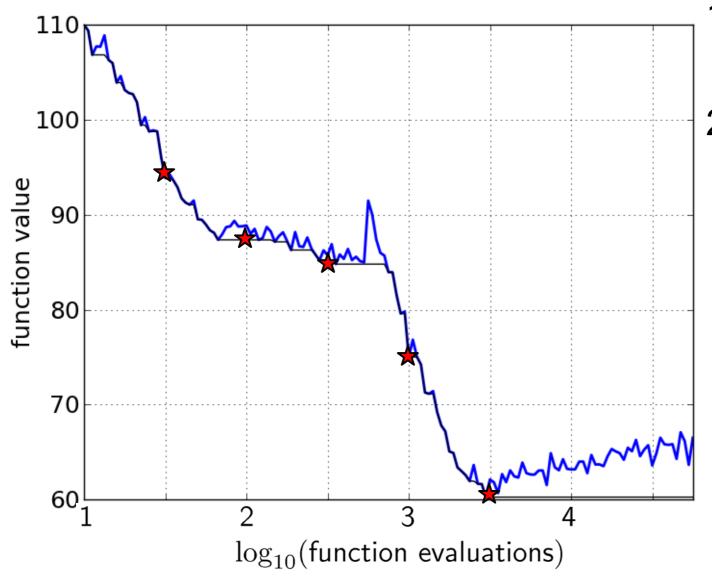
 define reference target budgets

Target budgets on the reference algorithm



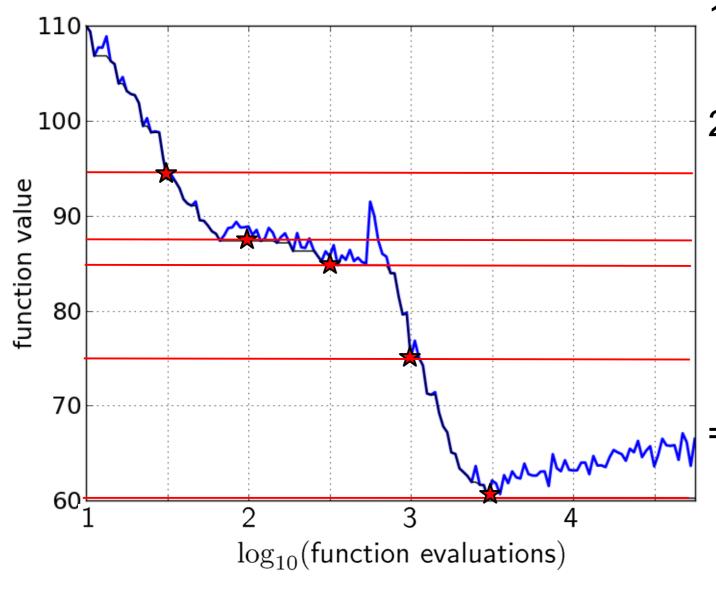
- define reference target budgets
- 2) compute best function value achieved by a reference algorithm

Target budgets on the reference algorithm



- define reference target budgets
- 2) compute best function value achieved by a reference algorithm

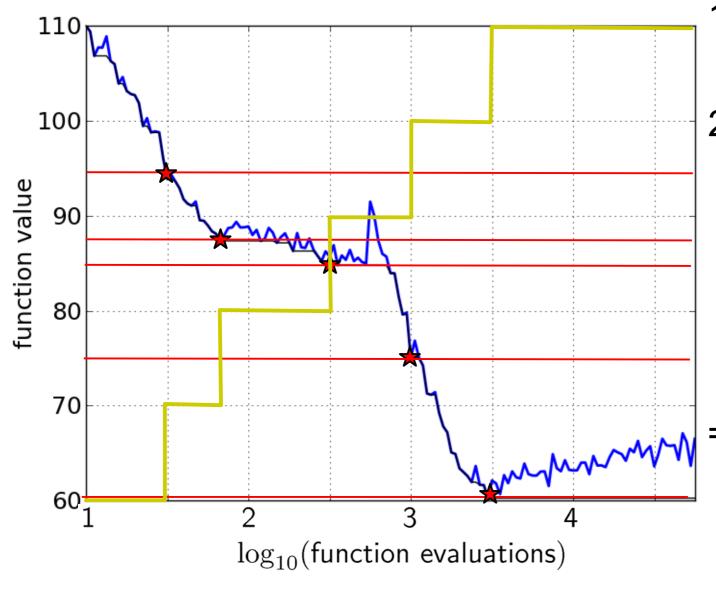
Run-length based target f-values



- define reference target budgets
- 2) compute best function value achieved by a reference algorithm

=> set of target
function values

Run-length based target f-values



- define reference target budgets
- 2) compute best function value achieved by a reference algorithm

=> set of target
function values

"Expensive" setting

- Changed target values compared to the original setting
- Target values depend on the function and the dimension
- Computation is based on the run length of the GECCO BBOB 2009 best result ==> runlengthbased target values

ECDF: Summary

Empirical Cumulative Distribution Functions

- recover a single convergence graph (and generalize)
- can aggregate over any set of functions and target values

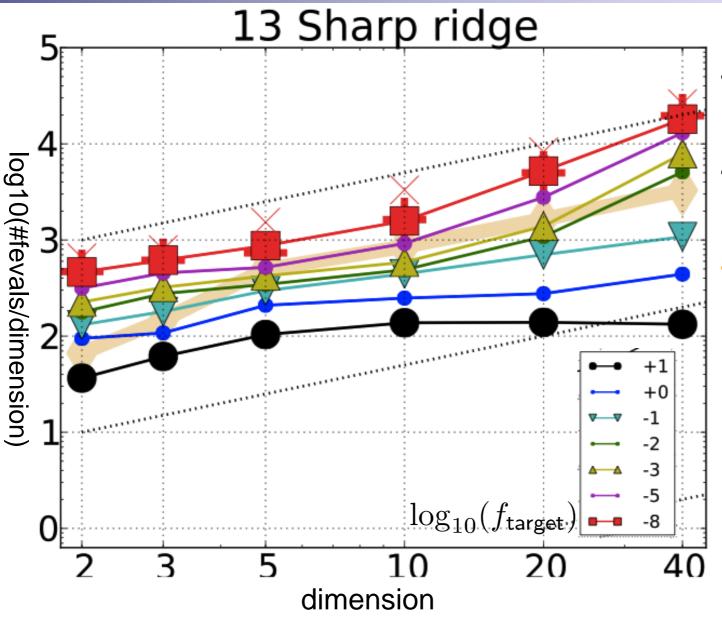
they display a set of run lengths or runtimes (RT)

- for RT on a single problem (function & target value)
 allow to estimate any statistics of interest from them,
 like median, expectation (ERT),... in a meaningful way
- AKA data profile [Moré&Wild 2009]
- Performance profile [Dolan&Moré 2002]: ECDFs of run lengths divided by the smallest observed run length

Questions?

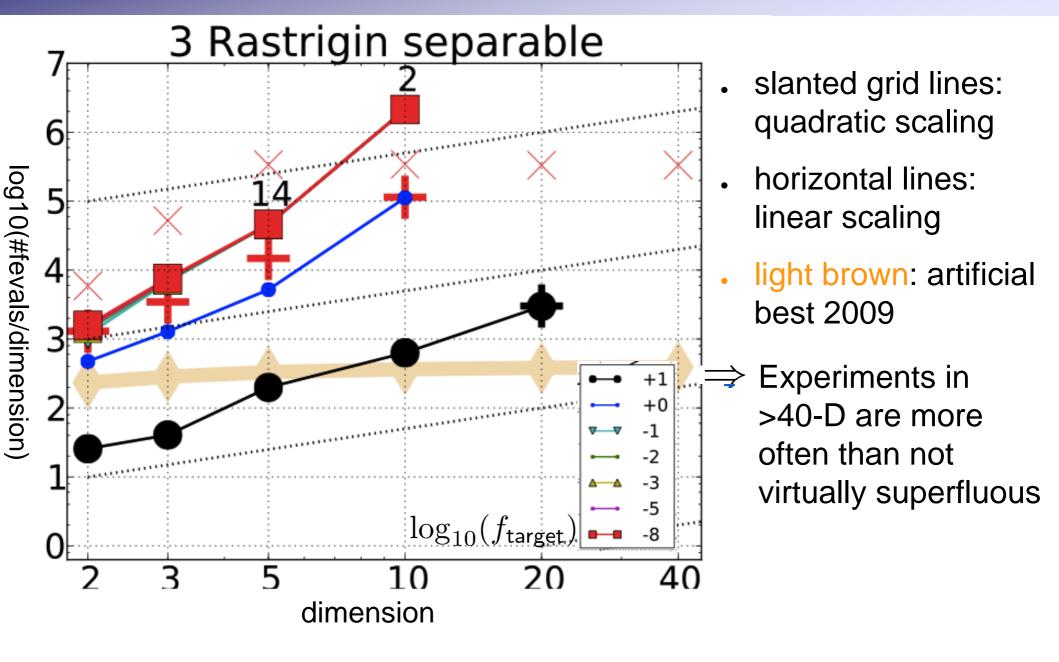
Different Displays of Runtimes

Scaling Behaviour with Dimension

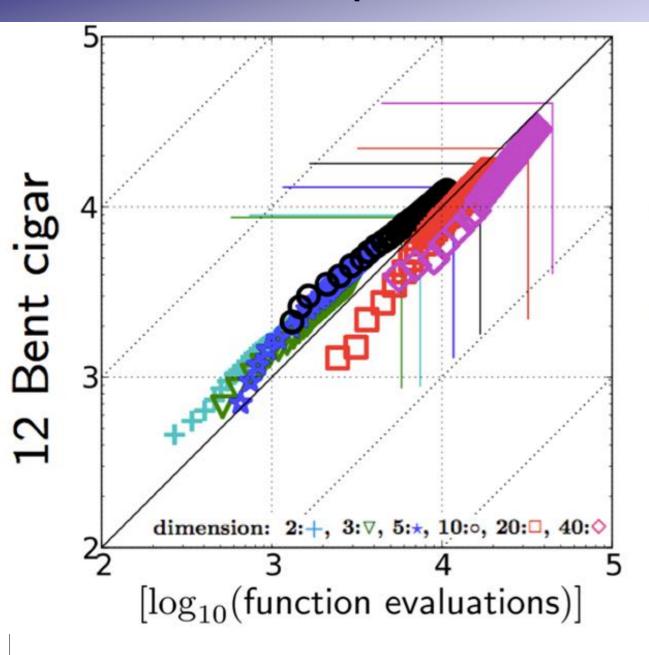


- slanted grid lines: quadratic scaling
 - horizontal lines: linear scaling
- light brown: artificial best 2009

Example: Scaling Behaviour

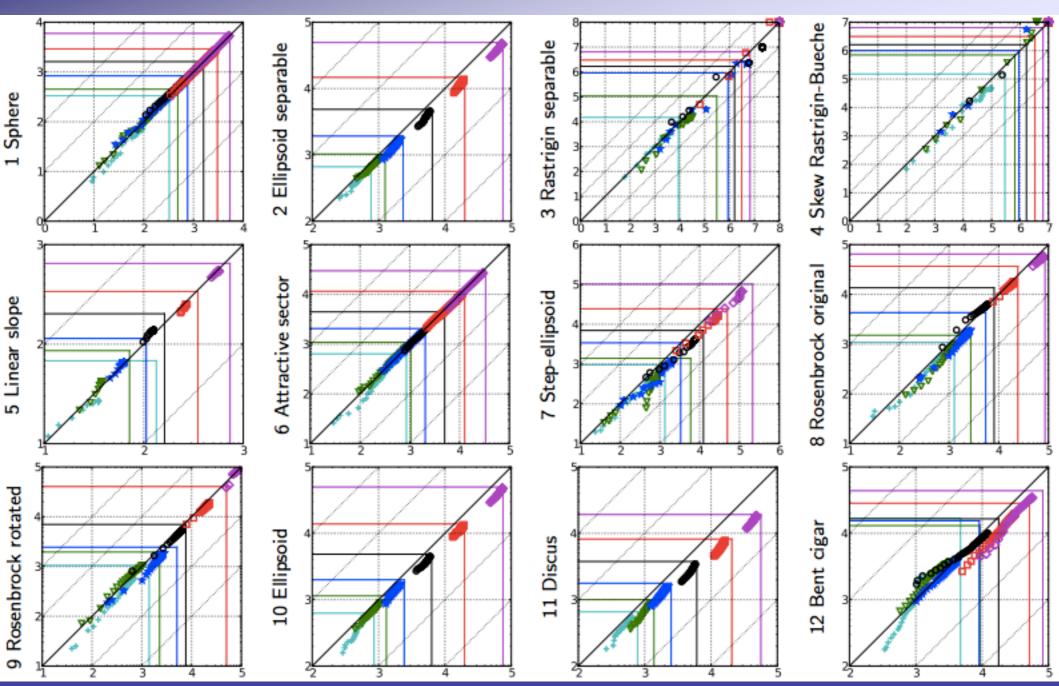


ERT scatter plots, all dimensions&targets



- estimated Expected Run Time (ERT), two algorithms
- 2-10 D: first algorithm "dominates"
- 20 & 40 D: second algorithm "dominates"

ERT scatter plots, all dimensions&targets



Single Function Table

Table 6: 20-D, running time excess ERT/ERT_{best} on f_6 , in italics is given the median final function value and the median number of function evaluations to reach this value divided by dimension

					6 Attrac	tive sector					
Δ ftarget	1e+03	1e+02	1e+01	1e + 00	1e-01	1e-02	1e-03	1e-04	1e-05	1e-07	Δftarget
ERT _{best} /D	4.03	26	64.7	87.2	123	152	184	219	248	309	ERT _{best} /D
ALPS	59	25	34	54	64	78	100	150	370	14c-7/2c5	ALPS [17]
AMaLGaM IDEA	26	22	19	22	21	22	22	21	22	22	AMalGaM IDÉA [4]
avg NEWUOA	2.3	1.1	1	1	1	1	1	1	1	1	avg NEWUOA [31]
BayEDAcG	46	41	60e+0/2e3								BayEDAcG [10]
BFGS	2.2	2.7	3.6	4.7	4.7	4.9	5	4.8	4.9	61	BFGS [30]
Cauchy EDA	6200	1500	1e3	1700	17e-1/5e4						Cauchy EDA [24]
BIPOP-CMA-ES	2.9	2.2	1.5	1.7	1.6	1.6	1.6	1.5	1.6	1.6	BIPOP-CMA-ES [15]
(1+1)-CMA-ES	1.9	4.5	13	180	1200	13c-1/1c4					(1+1)-CMA-ES [2]
DASA	12	6.8	9.9	19	25	33	49	58	63	74	DASA [19]
DEPSO	11	7.5	12	64	13e-1/2e3						DEPSO [12]
DIRECT	18	31	40e+0/5e3			2.5	- 4	-	2	- 1	DIRECT [25]
EDA-PSO	27	46	40	45	44	44	44	44	44	44	EDA-PSO [6]
full NEWUOA	5	1.9	1.5	1.4	1.4	1.4	1.4	1.4	1.4	1.4	full NEWUOA [31]
G3-PCX	4.1	1.4	1.4	2	2.1	2.1	2.2	2.2	2.3	2.4	G3-PCX [26]
simple GA	320	130	2e3	11e+0/1e5				7			simple GA [22]
GLOBAL	5	2.9	3.6	4.9	8.5	42c-3/2c3			74		GLOBAL [23]
iAMaLGaM IDEA	5.1	5.6	5.4	6.8	7.1	7.7	7.8	7.7	8	8.3	iAMaLGaM IDEÁ [4]
LSfminbnd	9	31	160	760	1100	960	72e-1/1e4		-		LSfminbnd [28]
LSstep	140	260	2300	59e+0/1e4							LSstep [28]
MA-LS-Chain	11	4.9	7.5	8.9	8	7.7	7.2	6.7	6.5	6	MA-LS-Chain [21]
MCS (Neum)	1.8	33	42e+0/4e3								MCS (Neum) [18]
NELDER (Han)	2.2	2.4	2.7	3.3	3.2	3.5	3.5	3.5	4	7.4	NELDER (Han) [16]
NELDER (Doe)	1.5	2.3	9.1	20	28	65	110	430	46e-5/2e4		NELDER (Doe) [5]
NEWUOA	1	1	1	1.3	1.4	1.5	1.6	1.6	1.7	1.7	NEWUOA [31]
(1+1)-ES	2	2.2	2.1	2.8	3.9	5.2	6.1	6.5	6.4	6.7	(1+1)-ES [1]
POEMS	89	26	31	37	36	36	36	35	36	37	POEMS [20]
PSO	6.4	280	1100	1400	980	820	710	620	570	790	PSO [7]
PSO_Bounds	9.5	45	120	150	140	140	140	130	160	220	PSO_Bounds [8]
Monte Carlo	2.4e5	48e+1/1e6				42	19		74		Monte Carlo [3]
Rosenbrock	2.1	3.9	31	76	210	230	810	21e-2/1e4			Rosenbrock [27]
IPOP-SEP-CMA-ES	3.2	2.1	1.7	1.9	1.9	1.9	1.9	1.9	2	2	IPOP-SEP-CMA-ES [29]
VNS (Garcia)	5	2.8	1.9	1.9	1.7	1.7	1.7	1.6	1.6	1.6	VNS (Garcia) [11]

Questions?