Parameter Optimization in the Nonlinear Stepsize Control Framework for Trust-Region Methods

EUROPT 2017

Abel Soares Siqueira
Federal University of Paraná - Curitiba/PR - Brazil

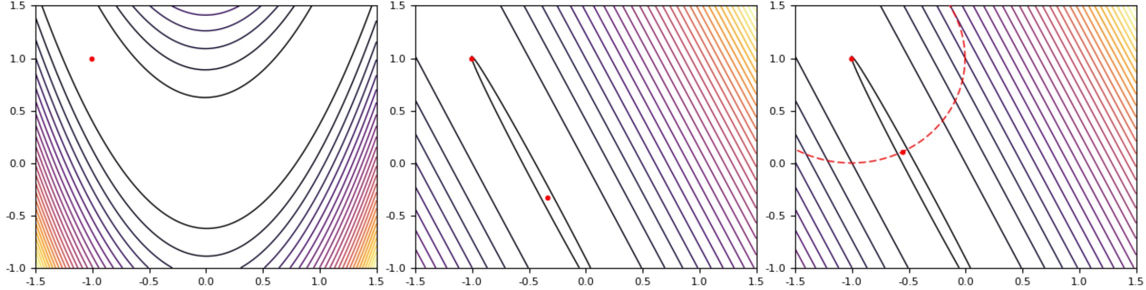
Geovani Nunes Grapiglia
Federal University of Paraná - Curitiba/PR - Brazil

July 12, 2017

Unconstrained Optimization

$$\min f(x),$$

$f : \mathbb{R}^n \to \mathbb{R}$ is twice continuously differentiable.



Classical Trust-Region Method (Powell [1])

1. $q_k(d) = f(x^k) + \nabla f(x^k)^T d + \frac{1}{2} d^T B_k d$
2. d^k such that $\|d^k\| \leq \delta_k$ and
$$q_k(0) - q_k(d^k) \geq \kappa \|\nabla f(x^k)\| \min\left\{ \frac{\|\nabla f(x^k)\|}{1 + \|B_k\|}, \delta_k \right\}.$$
3. $\rho_k = \dfrac{f(x^k) - f(x^k + d^k)}{q_k(0) - q_k(d^k)}$
4. If $\rho_k \geq \eta_1$, do $x^{k+1} = x^k + d^k$. Otherwise $x^{k+1} = x^k$.
5. Choose δ_{k+1}

Modified Trust-Region Method (Fan and Yuan [2])

1. $q_k(d) = f(x^k) + \nabla f(x^k)^T d + \frac{1}{2} d^T B_k d$
2. d^k such that $\|d^k\| \leq \delta_k \|\nabla f(x^k)\|$ and
$$q_k(0) - q_k(d^k) \geq \kappa \|\nabla f(x^k)\| \min \left\{ \frac{\|\nabla f(x^k)\|}{1 + \|B_k\|}, \delta_k \|\nabla f(x^k)\| \right\}.$$
3. $\rho_k = \dfrac{f(x^k) - f(x^k + d^k)}{q_k(0) - q_k(d^k)}$
4. If $\rho_k \geq \eta_1$, do $x^{k+1} = x^k + d^k$. Otherwise $x^{k+1} = x^k$.
5. Choose δ_{k+1}

ARC Method (Cartis, Gould, and Toint [3], [4])

1. $q_k(d) = f(x^k) + \nabla f(x^k)^T d + \frac{1}{2} d^T B_k d + \dfrac{1}{3\delta_k}\|d\|^3$
2. d^k such that $\|d^k\| \leq \delta_k^{\frac{1}{2}} \|\nabla f(x^k)\|^{\frac{1}{2}}$ and
$$q_k(0) - q_k(d^k) \geq \kappa \|\nabla f(x^k)\| \min\left\{ \frac{\|\nabla f(x^k)\|}{1 + \|B_k\|}, \delta_k^{\frac{1}{2}} \|\nabla f(x^k)\|^{\frac{1}{2}} \right\}.$$
3. $\rho_k = \dfrac{f(x^k) - f(x^k + d^k)}{q_k(0) - q_k(d^k)}$
4. If $\rho_k \geq \eta_1$, do $x^{k+1} = x^k + d^k$. Otherwise $x^{k+1} = x^k$.
5. Choose δ_{k+1}

NSC Method

- "Nonlinear stepsize control, trust regions and regularizations for unconstrained optimization", Toint (2013) [5]
- Generalizes trust-region and regularization methods;
- Provides unified convergence theory;
- Suggests new methods.

Let $\phi, \psi, \chi : \mathbb{R}^n \to \mathbb{R}$ be nonnegative functions such that

$$\min\{\phi(x), \psi(x), \chi(x)\} = 0 \Rightarrow x \text{ is a critical point}$$

NSC Method

1. $0 < \gamma_1 < \gamma_2 < 1$, $0 < \eta_1 \leq \eta_2 < 1$.
2. Find a model $q_k(d)$ such that $q_k(0) = f(x^k)$ and $f(x^k + d) - q_k(d) \leq \kappa_m \|d\|^2$
3. d^k such that $\|d^k\| \leq \Delta(\delta_k, \chi_k) = \delta_k^\alpha \chi_k^\beta$ and
$$q_k(0) - q_k(d^k) \geq \kappa \psi_k \min\left\{ \frac{\phi_k}{1 + \|B_k\|}, \Delta(\delta_k, \chi_k) \right\}.$$
4. $$\rho_k = \frac{f(x^k) - f(x^k + d^k)}{q_k(0) - q_k(d^k)}$$
5. If $\rho_k \geq \eta_1$, do $x^{k+1} = x^k + d^k$. Otherwise $x^{k+1} = x^k$.
6. $$\delta_{k+1} \in \begin{cases} [\gamma_1 \delta_k, \gamma_2 \delta_k] & \rho_k < \eta_1 \\ [\gamma_2 \delta_k, \delta_k] & \eta_1 \leq \rho_k < \eta_2 \\ [\delta_k, +\infty) & \rho_k \geq \eta_2 \end{cases}$$

NSC Method (Particular cases)

- Classical Trust-Region Method

$$\begin{cases} \alpha = 1 \text{ and } \beta = 0 \\ \phi_k = \psi_k = \chi_k = \|\nabla f(x^k)\| \end{cases} \Longrightarrow \Delta(\delta_k, \chi_k) = \delta_k$$

- Modified Trust-Region Method

$$\begin{cases} \alpha = \beta = 1 \\ \phi_k = \psi_k = \chi_k = \|\nabla f(x^k)\| \end{cases} \Longrightarrow \Delta(\delta_k, \chi_k) = \delta_k \|\nabla f(x^k)\|$$

- ARC Method

$$\begin{cases} \alpha = \beta = 1/2 \\ \phi_k = \psi_k = \chi_k = \|\nabla f(x^k)\| \end{cases} \Longrightarrow \Delta(\delta_k, \chi_k) = \delta_k^{\frac{1}{2}} \|\nabla f(x^k)\|^{\frac{1}{2}}$$

How α and β affect the method?

Theorem (Grapiglia, Yuan, and Yuan [6], 2016)

Under reasonable assumptions, the NSC method takes at most $\mathcal{O}(\epsilon^{-2})$ iterations to achieve $\chi_k \leq \epsilon$.

How α and β affect the method?

Theorem (Grapiglia, Yuan, and Yuan [6], 2016)

Under reasonable assumptions, the NSC method takes at most $\mathcal{O}(\epsilon^{-2})$ iterations to achieve $\chi_k \leq \epsilon$.

- How (α, β) affect the algorithm in practice;

How α and β affect the method?

Theorem (Grapiglia, Yuan, and Yuan [6], 2016)

Under reasonable assumptions, the NSC method takes at most $\mathcal{O}(\epsilon^{-2})$ iterations to achieve $\chi_k \leq \epsilon$.

- How (α, β) affect the algorithm in practice;
- Are the classical choices $(1, 0)$ and $(1, 1)$ among the best choices?

Numerical Experiments

- $q(d) = f(x^k) + \nabla f(x^k)^T d + \frac{1}{2} d^T \nabla^2 f(x^k) d$
- Find d^k by Steihaug-Toint
- $\|\nabla f(x^k)\| \leq 10^{-8} + 10^{-6} \|\nabla f(x^0)\|$
- Maximum f evaluations 5000, maximum time: 30 s;
- $\eta_1 = \frac{1}{4}$, $\eta_2 = \frac{3}{4}$, $\sigma_1 = \frac{1}{6}$, $\sigma_2 = 4$
- $\delta_{k+1} = \begin{cases} \sigma_1 \delta_k & \rho_k < \eta_1 \\ \delta_k & \eta_1 \leq \rho_k < \eta_2 \\ \sigma_2 \delta_k & \rho_k \geq \eta_2 \end{cases}$

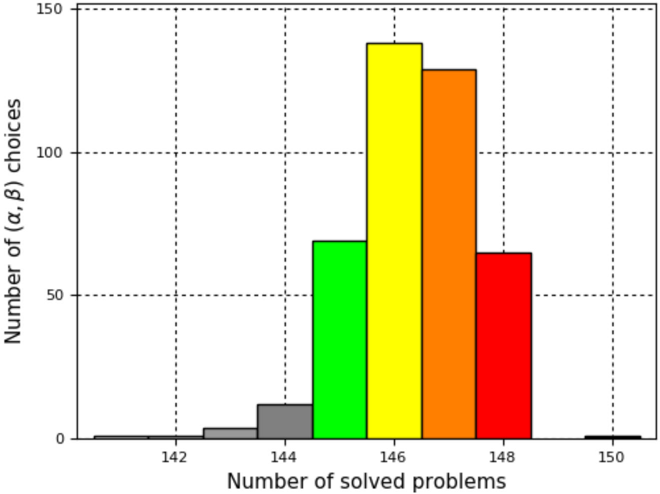
Numerical Experiments

- Similar to Gould, Orban, Sartenaer, *et al.* [7];
- $G = \left\{ \left(\frac{i}{20}, \frac{j}{20} \right) \mid \ i = 1, \dots, 20, \ j = 0, \dots, 20 \right\}$
- Define algorithm for each $(\alpha, \beta) \in G$;
- Run algorithm for 173 CUTEst problems (all unconstrained at the time);

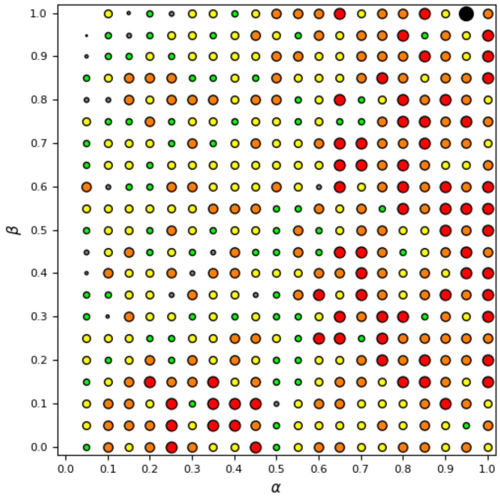
Robustness

- Robustness varies between 141 and 150 problems;
- Most: $(\alpha, \beta) = (0.95, 1.00)$;
- Least: $(\alpha, \beta) = (0.05, 1.00)$;
- All choices converged for 133 problems;
- All failed for 22 problems.

Robustness


150
100
50
0
142
144
146
148
150
Number of solved problems
Number of (α, β) choices

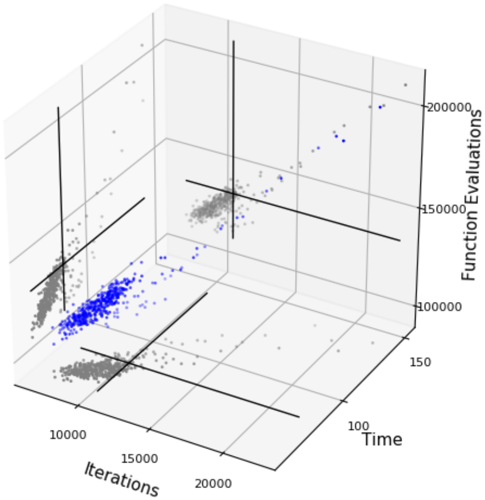
Robustness



Performance Metrics

- Elapsed time;
- Number of iterations;
- Number of functions evaluations;
- Consider the 133 converging problems;
- Comet plot similar do Gould, Orban, Sartenaer, *et al.* [7].

Performance Metrics



Performance Metrics - Correlations

- 0.878 between elapsed time and number of iterations;
- 0.784 between elapsed time and number of functions evaluations;
- 0.919 between number of iterations and number of functions evaluations.
- We chose functions evaluations as performance metrics.

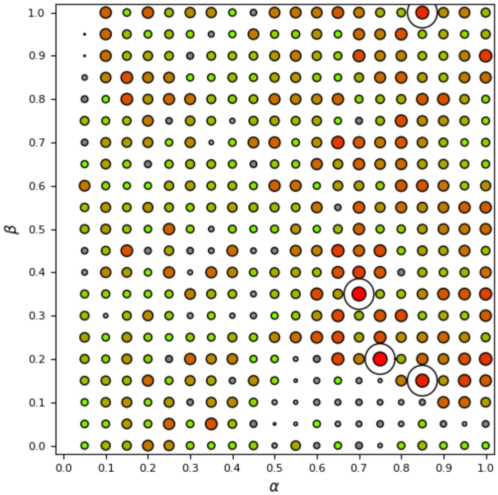
Best point for the comets

- Given $M = \{(x_i, y_i),\ i = 1, \dots, m\}$, (x_j, y_j) is dominated if $\exists\ (x_i, y_i) \neq (x_j, y_j)$ such that $x_i \leq x_j$ and $y_i \leq y_j$;
- The only point non-dominated for all three plots is $(0.45, 0.5)$;

Evaluations

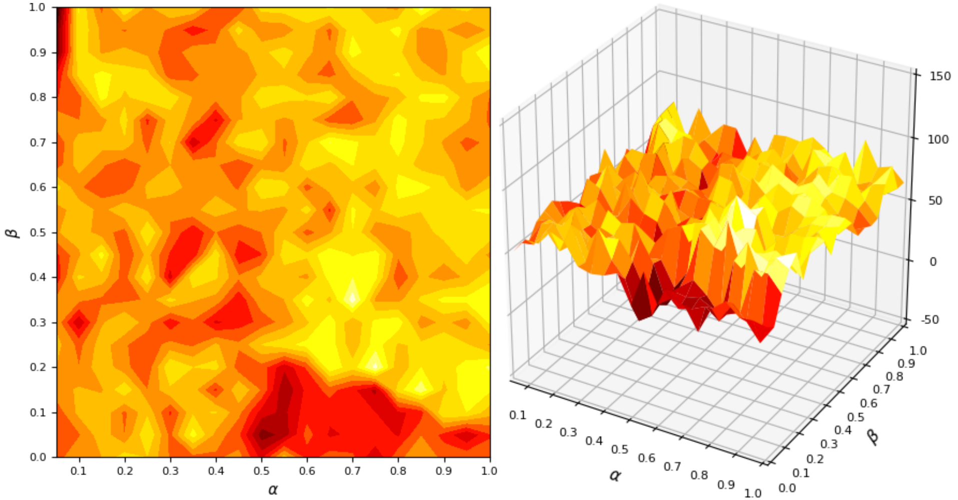
- $\#F_{p,\alpha,\beta}$ is the number of function evaluations declare convergence or failure for problem p and parameters (α, β).
- $\sigma_{p,\alpha,\beta} = \begin{cases} 1, & \text{if the problem converges,} \\ 0, & \text{otherwise.} \end{cases}$
- Declare the score $f(\alpha, \beta) = \sum_p \#F_{p,\alpha,\beta}\Big(\sigma_{p,\alpha,\beta} + 10(1 - \sigma_{p,\alpha,\beta})\Big)$

Score



Parameters	Position	Score
(0.70, 0.35)	1	100.0%
(0.75, 0.20)	2	99.49%
(0.85, 0.15)	3	94.05%
(0.85, 1.00)	4	91.94%
(0.95, 1.00)	99	75.86%
(1.00, 1.00)	207	64.37%
(0.45, 0.50)	345	51.42%
(1.00, 0.00)	352	50.86%

Score


1.0
0.9
0.8
0.7
0.6
0.5
0.4
0.3
0.2
0.1
0.0
β
0.1 0.2 0.3 0.4 0.5 0.6 0.7 0.8 0.9 1.0
α
150
100
50
0
-50
0.1 0.2 0.3 0.4 0.5 0.6 0.7 0.8 0.9 1.0
α
0.0 0.1 0.2 0.3 0.4 0.5 0.6 0.7 0.8 0.9 1.0
β

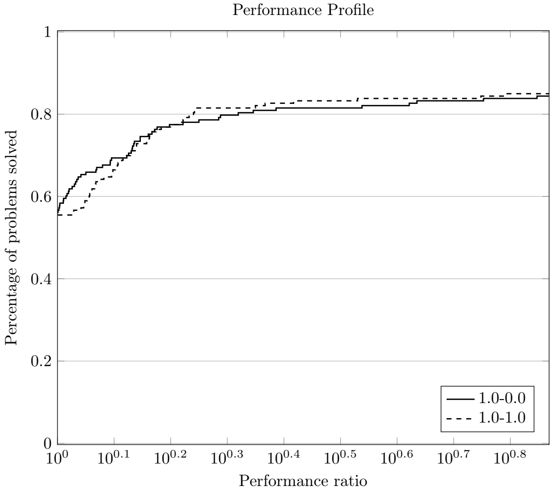
Performance Profiles

- Performance profile, for solvers $\mathcal{S}$ and problems $\mathcal{P}$
  - Cost $c_{s,p}$ ($+\infty$ if failed);
  - $r_{s,p} = \dfrac{c_{s,p}}{\min\{c_{s,p} \mid s \in \mathcal{S}\}} \quad s \in \mathcal{S}, p \in \mathcal{P}$;
  - $r_f = \max\{r_{s,p} \mid r_{s,p} < +\infty\}$.
  - $\rho_s(t) = \dfrac{\#\{r_{s,p} \leq t \mid p \in \mathcal{P}\}}{\#\mathcal{P}}$;
  - $\rho_s(1)$ is an efficiency measure;
  - $\rho_s(r_f)$ is the robustness (independent of $\mathcal{S}$);
- Used Perprof-py [8].
- Full set of 173 problems;

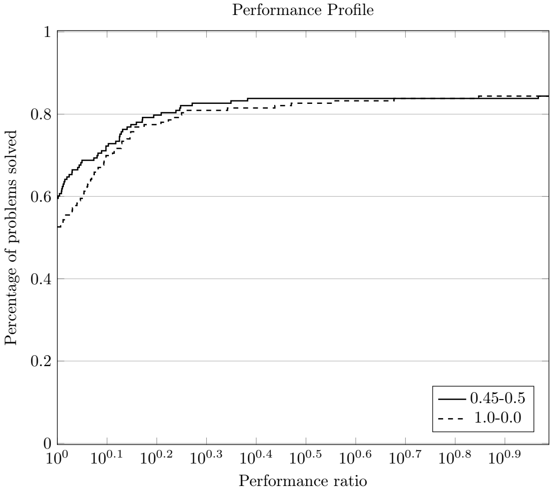
Performance Profiles

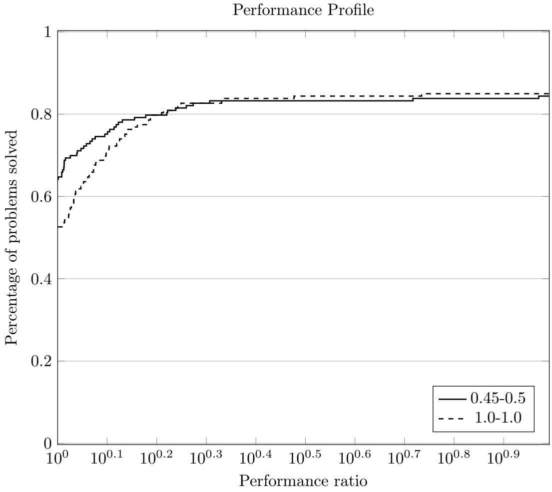
- Total number of functions evaluations;
- Classical: $(1, 0)$ and $(1, 1)$;
- Most robust: $(0.95, 1)$;
- Best for comet: $(0.45, 0.5)$;
- Best score: $(0.7, 0.35)$;

Performance Profile - (1,0) vs (1,1)

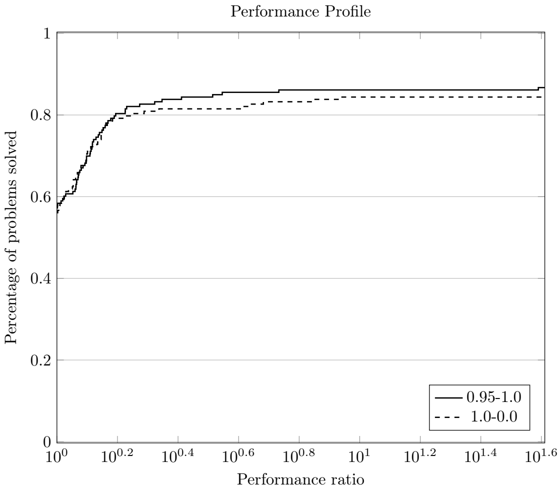


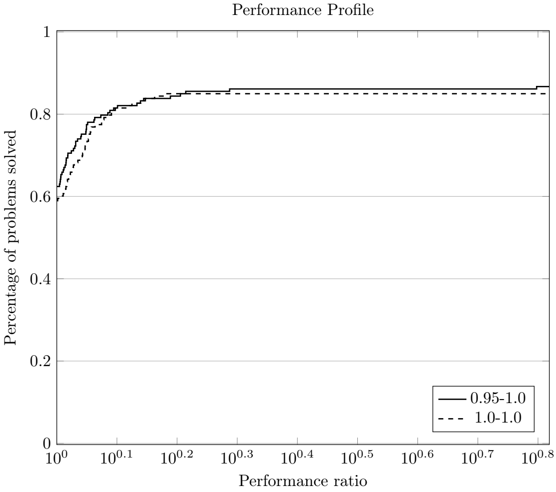
Performance Profile - (1,0) and (1,1) vs (0.45,0.5)



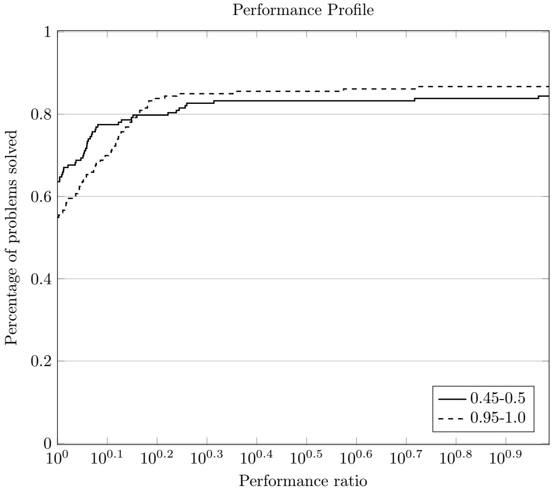


Performance Profile - (1,0) and (1,1) vs (0.95,1)

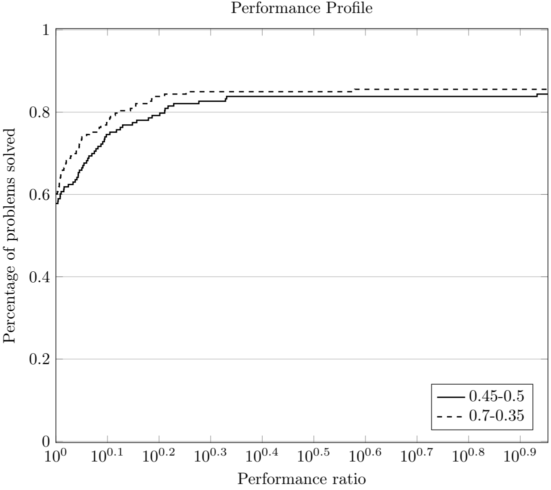


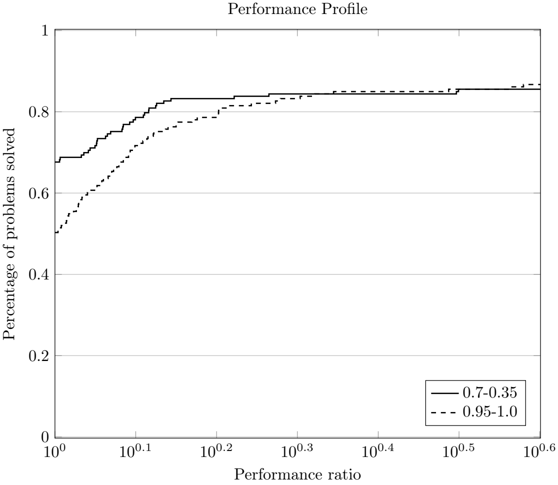


Performance Profile - (0.45,0.5) vs (0.95,1)



Performance Profile - (0.45,0.5) and (0.95,1) vs (0.7,0.35)





Parameter Optimization

- We've stabilished that the classical parameters are not the best;
- Furthermore, we've found a parameter choice that greatly increases the efficiency of the algorithm;
- Can we do optimize this choice?
- Following Audet and Orban [9], let's use Derivative-Free Optimization to find optimal parameters;

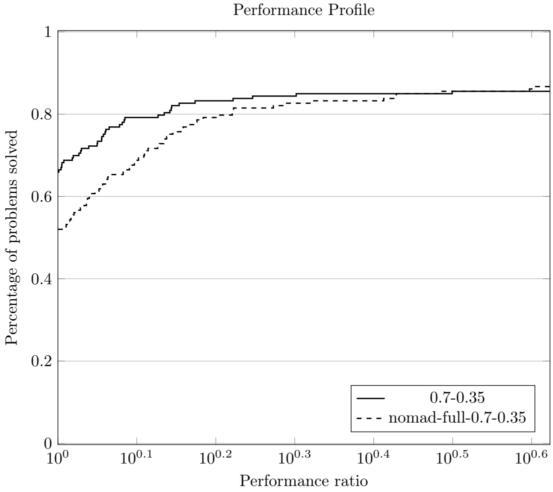
Parameter Optimization

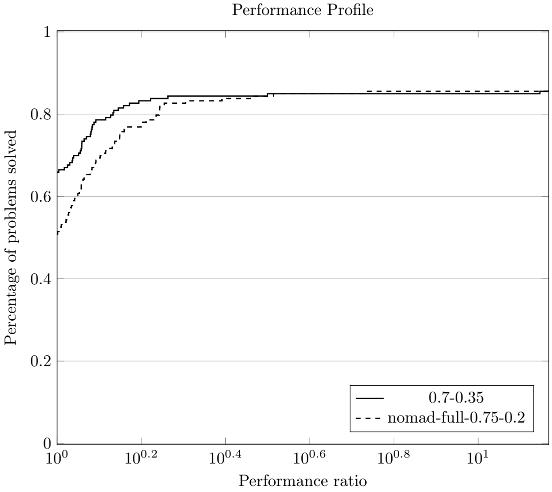
- We'll use NOMAD to minimize $f(\alpha, \beta)$ subject to $\begin{cases} 0 < \alpha \leq 1, \\ 0 \leq \beta \leq 1; \end{cases}$
- Surrogate function using fast problems didn't work well;
- From our results so far, we sense many local minima;
- Let's start NOMAD from different starting points from the grid;

Parameter Optimization

- From (0.7,0.35), we found (1, 0.9899494937), with 108.67% relative score, after 173 NOMAD evaluations;
- From (1,0) we found the same point after 302 NOMAD evaluations;
- From (0.75,0.2) we found (1, 0.8689539166), with 110.98% relative score, after 185 NOMAD evaluations;

(0.7, 0.35) against (1, 0.9899494937) and (1, 0.8689539166)

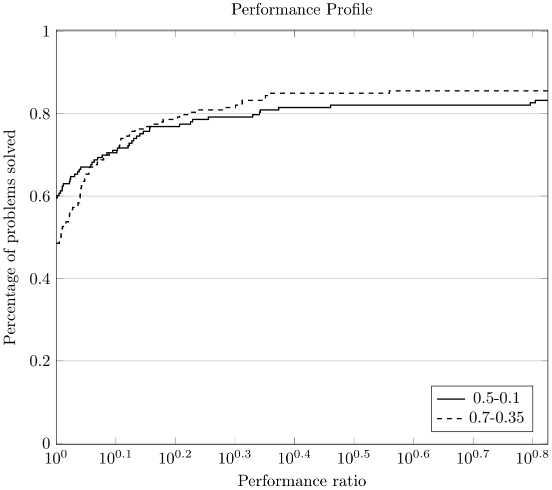


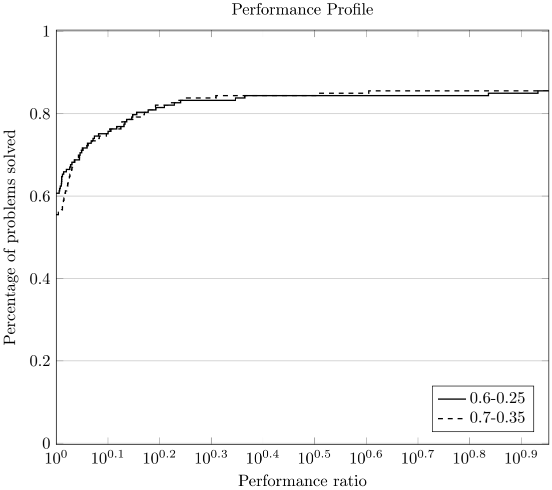


Searching the Performance Profiles

- Optimizing the score is not the same as optimizing the efficiency on the Performance Profile;
- Search among the efficiency of all performance profiles of (α, β) against $(0.7, 0.35)$;
- Restricting the robustness to the same as $(0.7, 0.35)$ or not.

Searching the Performance Profiles





Conclusions

- About 12 days of computer work;
- The NSC framework is actually very dependent on (α, β);
- There are many choices superior to $(1, 0)$ and $(1, 1)$;
- $\approx (1, 0.869)$ is the best found on the used metric;
- $(0.6, 0.25)$ and $(0.7, 0.35)$ are very good overall;
- Test on your specific problems.

Future work

- Optimize all parameters at the same time;
- ARC method;
- Modification that reduces sensitivity? (non-monotonicity?);

M. J. D. Powell, "Convergence properties of a class of minimization algorithms", in *Nonlinear Programming 2*, O. L. Mangasarian, R. R. Meyer, and S. M. Robinson, Eds., Academic Press, New York, 1975.

J. Fan and Y. Yuan, "A new trust region algorithm with trust region radius converging to zero", in *Proceedings of the 5th International Conference on Optimization: Techniques and Applications (ICOTA 2001, Hong Kong)*, D. Li, Ed., 2001, pp. 786–794.

C. Cartis, N. I. M. Gould, and P. L. Toint, "Adaptive cubic overestimation methods for unconstrained optimization. part I: Motivation, convergence and numerical results.", *Mathematical Programming, Series A*, vol. 127, no. 2, pp. 245–295, 2011. DOI: 10.1007/s10107-009-0286-5.

——,"Adaptive cubic overestimation methods for unconstrained optimization. part II: Worst-case function - and derivative - evaluation complexity", *Mathematical Programming*, vol. 130, no. 2, pp. 295–319, 2011. DOI: 10.1007/s10107-009-0337-y.

P. L. Toint, "Nonlinear stepsize control, trust regions and regularizations for unconstrained optimization", *Optimization Methods and Software*, vol. 28, no. 1, pp. 82–95, 2013. DOI: 10.1080/10556788.2011.610458.

G. N. Grapiglia, J. Yuan, and Y. Yuan, "Nonlinear stepsize control algorithms: Complexity bounds for first- and second-order optimality", *Journal of Optimization Theory and Applications*, 2016.

N. I. M. Gould, D. Orban, A. Sartenaer, and P. L. Toint, "Sensitivity of trust-region algorithms to their parameters", *4OR*, vol. 3, no. 3, pp. 227–241, 2005. DOI: 10.1007/s10288-005-0065-y.

A. S. Siqueira, R. G. C. da Silva, and L.-R. Santos, "Perprof-py: A python package for performance profile of mathematical optimization software", *Journal of Open Research Software*, vol. 4, no. 1, e12, 2016. DOI: 10.5334/jors.81.

C. Audet and D. Orban, "Finding optimal algorithmic parameters using derivative-free optimization", *SIAM Journal on Optimization*, vol. 17, no. 3, pp. 642–664, 2006. DOI: 10.1137/040620886.

Thanks