Cumulative Step-size Adaptation on Linear Functions

Alexandre Chotard, Anne Auger and Nikolaus Hansen

TAO, Inria Saclay, Université Paris-Sud, France

name@lri.fr

Introduction and motivation

- Objective: optimize $f : \mathbb{R}^n \to \mathbb{R}$, where f is the composite of a linear function by a strictly increasing function.
- Model for when the step-size is small compared to the distance to the optimum. This situation threatens premature convergence.
- W.l.o.g., as CSA-ES is invariant under change of orthonormal basis, we can assume that f is the projection on the first dimension of a point of $\mathbb{R}^n$, that is $f(\boldsymbol{x}) = [\boldsymbol{x}]_1$.
- Motivation: the linear functions case must be handled well by any search algorithm by increasing the step-size, which is of critical importance in converging independently of the starting point in more general functions. It is not handled well by the $(1,2)$-SA-ES.

$(1,\lambda)$-CSA-ES

While stopping criterion has not been met:

- Generate λ new samples from previous selected point $\boldsymbol{X}^{(g)}$ of generation g with i.i.d. sequence $(\boldsymbol{\xi}_i^{(g)})_{i\in[[1,\lambda]]}$ of random steps, distributed according to a standard normal law $\mathcal{N}(\boldsymbol{0}, Id_n)$:

$$\boldsymbol{Y}_i^{(g)} = \boldsymbol{X}^{(g)} + \sigma^{(g)}\boldsymbol{\xi}_i^{(g)}$$

- Select the sample minimizing f:

$$\boldsymbol{X}^{(g+1)} = \underset{\boldsymbol{Y}\in(\boldsymbol{Y}_i^{(g)})_{i\in[[1,\lambda]]}}{\operatorname{argmin}} f(\boldsymbol{Y}) = \boldsymbol{X}^{(g)} + \sigma^{(g)}\boldsymbol{\xi}_\star^{(g)}$$

- Adapt the cumulative path with the selected step:

$$\boldsymbol{p}^{(g+1)} = (1-c)\boldsymbol{p}^{(g)} + \sqrt{c(2-c)}\boldsymbol{\xi}_\star^{(g)} \quad (1)$$

Coefficients were chosen such that if $\boldsymbol{p}^{(g)} \sim \mathcal{N}(\boldsymbol{0}, Id_n)$ and $\boldsymbol{\xi}_\star^{(g)} \sim \mathcal{N}(\boldsymbol{0}, Id_n)$ (which is the case if f is 'random'), then $\boldsymbol{p}^{(g+1)} \sim \mathcal{N}(\boldsymbol{0}, Id_n)$.

- Adapt the step-size according to the cumulative path:

$$\sigma^{(g+1)} = \sigma^{(g)}\exp\left(\frac{c}{2d_\sigma}\left(\frac{\|\boldsymbol{p}^{(g+1)}\|^2}{n} - 1\right)\right) \quad (2)$$

- Increment g and loop over.

Simulations

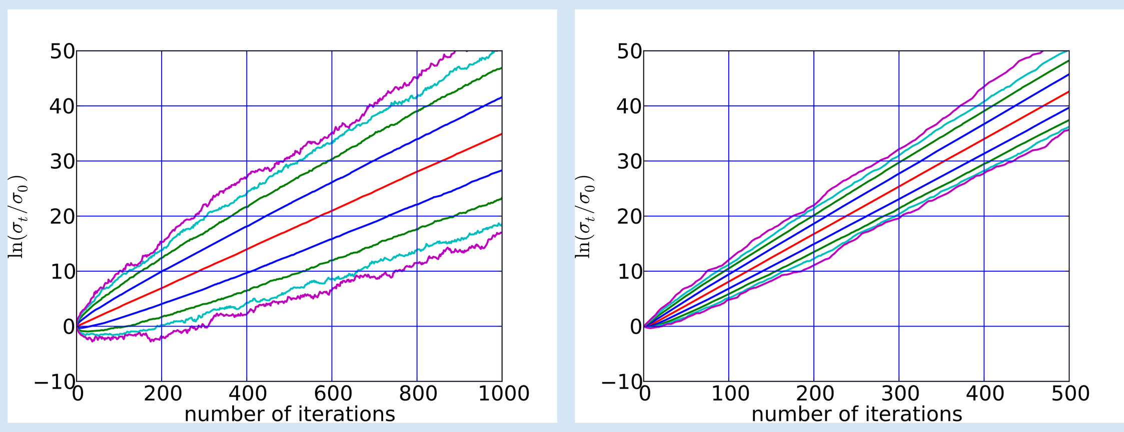

Fig. 1: Plot of the quantiles of 5001 simulations of $\ln(\sigma^{(g)}/\sigma^{(0)})$ against g with $\lambda = 8$ and $n = 20$. From the top to the bottom are the $1-10^{-i}$ quantiles, then the median, then the 10^{-i} quantiles (for $i = 1..4$). In the left plot, $c = 1$, and in the right plot $c = 1/\sqrt{20}$.
A lower c gives here a higher divergence speed, and decreases the standard deviation of $\ln(\sigma^{(g+1)}/\sigma^{(g)})$, relatively to its expected value. This decreases the probability of $\ln(\sigma^{(g)}/\sigma^{(0)})$ being negative (as it here happens with $c = 1$).

CSA-ES without cumulation

- Without cumulation $c = 1$ so Eq. (1) becomes $\boldsymbol{p}^{(g+1)} = \boldsymbol{\xi}_\star^{(g)}$.
- Applying the LLN with Eq. (2) we get geometric divergence of the step-size for $\lambda \geq 3$

$$\frac{1}{g}\ln\left(\frac{\sigma^{(g)}}{\sigma^{(0)}}\right) \xrightarrow[g\to\infty]{a.s} \frac{1}{2d_\sigma n}\left(\mathbf{E}\left(\mathcal{N}_{1:\lambda}^2\right) - 1\right) > 0 \quad (3)$$

With $\mathcal{N}_{i:\lambda}$ being the i^{th} order statistic of λ random variables, i.i.d. according to a standard normal distribution.

- With a LLN for Markov chains we find a similar result on $\boldsymbol{X}^{(g)}$ for $\lambda \geq 3$

$$\frac{1}{g}\ln\left|\frac{\left[\boldsymbol{X}^{(g)}\right]_1}{\left[\boldsymbol{X}^{(0)}\right]_1}\right| \xrightarrow[g\to\infty]{a.s} \frac{1}{2d_\sigma n}\left(\mathbf{E}\left(\mathcal{N}_{1:\lambda}^2\right) - 1\right) > 0 \quad (4)$$

CSA-ES with cumulation

Through Markov chain analysis we obtain geometric divergence of the step-size for $\lambda \geq 2$ and $c < 1$

$$\frac{1}{g}\ln\left(\frac{\sigma^{(g)}}{\sigma^{(0)}}\right) \xrightarrow[g\to\infty]{a.s} \frac{1}{2d_\sigma n}\left((2-2c)\,\mathbf{E}\left(\mathcal{N}_{1:\lambda}\right)^2 + c\left(\mathbf{E}\left(\mathcal{N}_{1:\lambda}^2\right) - 1\right)\right) > 0 \quad (5)$$

Noise to Signal ratio

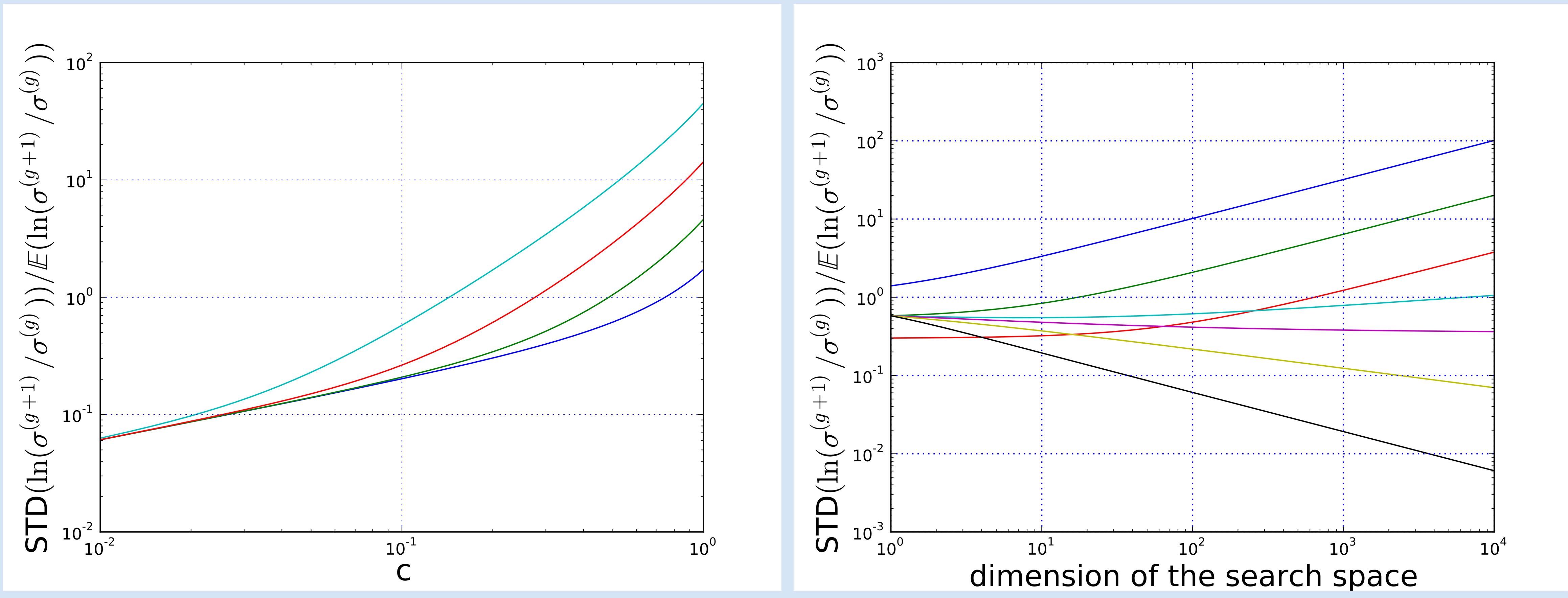

Fig. 2: Plot of the standard deviation of $\ln(\sigma^{(g+1)}/\sigma^{(g)})$ divided by its expected value, for $\lambda = 8$. On the left for different dimensions, from bottom to top $n = 2,\ 20,\ 200,\ 2000$. On the right different c, from top to bottom: $c = 1,\ 0.5,\ 0.2,\ 1/(1+n^{1/4}),\ 1/(1+n^{1/3}),\ 1/(1+n^{1/2}),\ 1/(1+n)$
As shown in Fig. 2 right, with a constant c the relative standard deviation increases with the dimension as $\sqrt{n}$. However, in Fig. 2 left, decreasing c decreases as well the relative standard deviation, as was shown in Fig. 1. Finally, we see in Fig. 2 right that for $c < 1/(1+n^{1/3})$ the relative standard deviation decreases with the dimension.

Divergence rate as a function of λ

The divergence rate in Eq. (5) increases with λ, as does the number of function evaluations per iterations. Dividing the right hand side of Eq. (5) by λ, we obtain a speed per evaluation (instead of a speed per iteration), shown in the following curves:

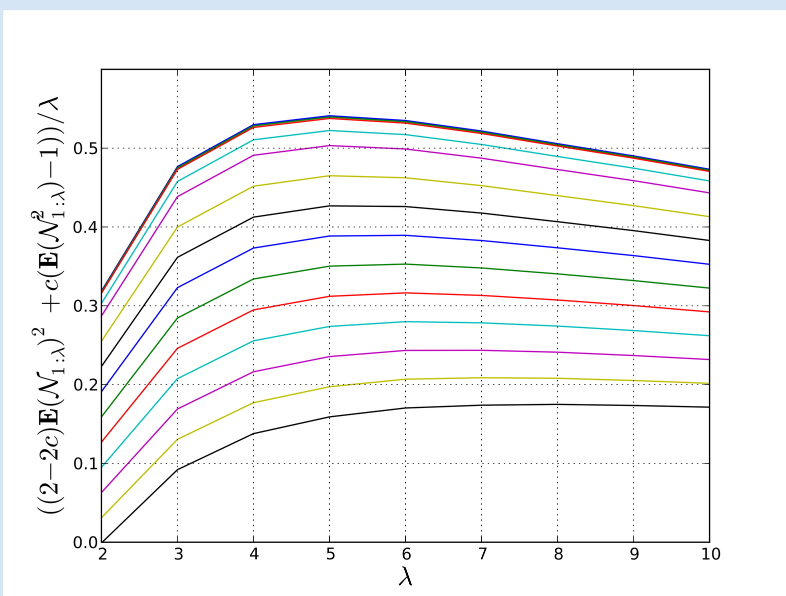

Fig. 3: Plot of $((2-2c)\mathbf{E}(\mathcal{N}_{1:\lambda})^2 + c(\mathbf{E}(\mathcal{N}_{1:\lambda}^2) - 1))/\lambda$ against λ for different values of c. The lowest curve is for $c = 1$, then $c = 0.9,\ \cdots,\ 0.1,\ 0.05,\ 0.01$.

The value of λ optimizing this divergence speed per evaluation depends here on the value of c, from $\lambda = 5$ when $c \leq 0.3$ to $\lambda = 8$ when $c = 1$.

Advantages of Cumulation

- For $\lambda = 2$,
  - Without cumulation ($c = 1$) the algorithm fails on linear functions (more precisely, $\ln(\sigma^{(g)})$ does an unbiased random walk), like the $(1,2)$-SA-ES, for the same symmetry reasons.
  - Cumulation ($c<1$) solves the problem, with the step-size geometrically diverging almost surely.
- Circumvent the problem of selection noise. For $c = 1/n^\alpha$
  - With $\alpha < 1/3$, the noise to signal ratio goes to infinity. More accurately, if c is constant then the standard deviation of $\ln(\sigma^{(g+1)}/\sigma^{(g)})$ divided by its expected value grows as $\sqrt{n}$ with the dimension.
  - With $\alpha > 1/3$, the noise to signal ration goes to 0, giving the algorithm strong stability in high dimensions.
- Decreasing c increases the divergence rate on linear functions. For $\lambda = 5$, it can be increased by more than 3 times compared to without cumulation.