CSG: A stochastic gradient method for a wide class of optimization problems appearing in a machine learning or data-driven context

Lukas Pflug*, Max Grieshammer†, Andrian Uihlein†, and Michael Stingl†

Abstract. In a recent article the so called *continuous stochastic gradient method* (CSG) for the efficient solution of a class of stochastic optimization problems was introduced. While the applicability of known stochastic gradient type methods is typically limited to so called expected risk functions, no such limitation exists for CSG. The key to this lies in the computation of design dependent integration weights, which allows for an optimal usage of available information leading to stronger convergence properties. However, due to the nature of the formula for these integration weights, the practical applicability was essentially limited to problems, in which stochasticity enters via a low-dimensional and suficiently simple probability distribution. In this paper the scope of the CSG method is significantly extended presenting new ways of calculating the integration weights. A full convergence analysis for this new variant of the CSG method is presented and its efficiency is demonstrated in comparison to more classical stochastic gradient methods by means of a number of problem classes, relevant in stochastic optimization and machine learning.

1. Introduction.

In the context of optimization problems in which the expected-value of a cost function j is minimized, i.e.,

$$\min_{\theta\in\Theta^{\mathrm{ad}}} \quad \mathbb{E}[j(\theta,X)] = \int_{\mathcal{X}} j(\theta,x)\mu(\mathrm{d}x) \tag{1.1}$$

with probability measure μ and the associated random variables X, a variety of different stochastic optimization schemes has been developed in the past, e.g., [7, 12, 13, 22, 24]. Among the most popular algorithms are the *stochastic gradient method* (SG) [17] and its modification the *stochastic average gradient method* (SAG) [18], both of which shine with their low iteration cost and have been analyzed extensively. a variety of different stochastic optimization schemes has been developed in the past. Among the most popular algorithms are the *stochastic gradient method* (SG) [17] and its modification the *stochastic average gradient method* (SAG) [18]. Both of these methods have been analyzed extensively in literature and are characterized by a low cost per iteration.

Nonetheless, SG and SAG face a number of known disadvantages, like the lack of efficient stopping criteria (cf. [15]) or optimal stepsize rules (cf. [14, 19]). To tackle these issues, a whole variety of modified SG methods can be found in the literature. For example, [7] uses a trust-region-type model to normalize the steplengths, whereas the iSARAH algorithm proposed in [13] combines an inner SG scheme with an outer (inexact) full gradient descent method.

Another disadvantage of SG is the quite restrictive setting of (1.1). [22] and [24] suggest inexact proximal stochastic second-order methods and stochastic primal-dual fixed-point methods to allow for a different type of objective function appearing in (1.1). In the case that the constraints include expected-valued functions, a level set method is analyzed in [12].

*Central Institute for Scientific Computing (ZISC), lukas.pflug@fau.de

†Department of Mathematics, Chair of Applied Mathematics (Continuous Optimization), Friedrich-Alexander University Erlangen-Nürnberg (FAU), max.grieshammer@fau.de, andrian.uihlein@fau.de, michael.stingl@fau.de

An even wider class of problems, can be solved by the *continuous stochastic gradient method* (CSG) proposed in [16]. The reason is that combining the information collected in previous iterations in an optimal way, CSG gains a significantly improved gradient approximation and is able to estimate the current objective function value during the optimization process. For a characterization of the class of problems CSG can solve, we refer to Remark 2.3. Here we just note that among them are objective finctions with nested expectation values (Section 5.2) and problems with chance constraints (Section 5.3).

While this is already known from the original version of CSG [16], there is also a serious drawback: in order to approximate function values and gradients in the above mentioned way, integration weights have to be computed by an analytical formula, which requires full knowledge about the probability measure μ. Moreover the evaluation is based on a Voronoi diagram, whose computation is not tractable, if the dimension of the parameter set $\mathcal{X}$ is larger than 2. As a consequence, in [16] only examples with a one-dimensional uniform distribution were presented.

In this contribution, we expand the setting of CSG even further by introducing new methods of calculating the weights used for the gradient and cost function value approximations. This enables us to apply the CSG algorithm to problems of higher dimension, to arbitrary measure μ and even to problems where the measure μ appearing in (1.1) might be unknown, e.g., in a data-driven context.

Depending on the concrete setting, i.e., depending on the dimensions of θ, x and on how time-consuming the evaluation of a gradient sample is, the different methods allow us to continuously trade weight-computation time and speed of convergence (w.r.t. number of gradient sample evaluations).

In this article we present of a full convergence analysis for the CSG method extended in this way. In particular, we show that the error in the gradient approximation as well as in the objective function value approximation vanish as the number of steps increases. As a consequence these values can be utilized, for instance, to apply stopping criteria based on first order optimality conditions. Moreover, this potentially allows, to combine the CSG method with slightly adapted step length strategies as they are known from the world of deterministic optimization methods, a topic we leave open for future research.

The remaining structure of the paper is as follows. In Section 2, the mathematical structure of the problems, we would like to solve by the CSG method is outlined in details. In Section 3, the CSG method with generalized weight computation is presented. Section 4 is devoted to the convergence analysis and in Section 5 we compare the generalized CSG method to more traditional SG-type algorithm using three different classes of test problems.

2. Problem setting and definitions.

Following the classic setup for expected-value objective functions, we introduce the set of admissible designs $\mathcal{P} \subset \mathbb{R}^{d_{\text{des}}}$ and the parameter set $\mathcal{X} \subset \mathbb{R}^{d_{\text{par}}}$, where $d_{\text{des}}, d_{\text{par}} \in \mathbb{N}$. In the optimization process, the drawn random samples $x_1, x_2, \ldots$ from the parameter set $\mathcal{X}$ are assumed to be realizations of independent uniformly random variables $X_i \sim \mu$ for all $i \in \mathbb{N}$, i.e., $X_1, X_2, \ldots$ are independent and follow an underlying probability distribution μ, which may be unknown.

To be precise, we define the following probability space setup:

Definition 2.1 (Probability space setup). *The probability space* $(\Omega, \mathcal{A}, \mathbb{P})$ *is given by*

$$\begin{aligned}\Omega &:= \mathcal{X}^{\mathbb{N}}, \mathbb{P} := \mu^{\otimes\mathbb{N}},\\ \mathcal{A} &:= \sigma(\{A_1 \times \ldots \times A_n : A_i \in \mathcal{B}(\mathcal{X}), \forall i, n \in \mathbb{N}\}),\end{aligned}$$

where $\mu^{\otimes\mathbb{N}}(A_1 \times \ldots \times A_n) = \prod_{i=1}^{n} \mu(A_i)$ *is the product measure,* μ *is a probability measure on* $\mathcal{X}$ *and* $\sigma(\cdot)$ *is the smallest* σ*-field that contains* $\cdot$*. We denote by* $\operatorname{supp}(\mu)$ *the support of the measure* μ*, i.e.,*

$$\operatorname{supp}(\mu) := \{x \in \mathcal{X} : \ \mu(B_\varepsilon(x)) > 0 \ \forall \varepsilon > 0\},$$

where $B_\varepsilon(x)$ *denotes an open ball of radius* $\varepsilon > 0$ *around* $x \in \mathcal{X}$*. We write* $X_n : \Omega \to \mathcal{X}$*,* $(\omega_k)_{k\in\mathbb{N}} \mapsto \omega_n$ *for the projection to the* $n = 1, 2, \ldots$ *coordinate and define* $X := X_1$*.*

With this setup, the objective function takes the following form:

Definition 2.2 (Objective function). *The objective function* $J : \mathcal{P} \to \mathbb{R}$ *is given by*

$$J(\theta) := \mathbb{E}\big[j(\theta, X)\big] = \int_{\mathcal{X}} j(\theta, x)\, \mu(\mathrm{d}x)$$

with a measurable function $j \in C^1(\mathcal{P} \times \mathcal{X}; \mathbb{R})$ *and random variable* X*.*

Remark 2.3 (Generalization of the setting). During the optimization process, we may also generate an approximation $\hat{J}_n$ to the exact objective function value $J(\theta_n)$ with almost no additional computational cost. We will show later that $\|\hat{J}_n - \nabla J(\theta_n)\|_{\mathcal{P}} \to 0$ (see Remark 4.8).

This enables us to solve a much broader class of optimization problems, where the objective function may depend non-linearly on the expression above, i.e.,

$$\tilde{J}(\theta) := f\big(\theta, \mathbb{E}[j(\theta, X)]\big),$$

with a Lipschitz continuously differentiable function $f : \mathcal{P} \times \mathbb{R} \to \mathbb{R}$. Included in the set of possible objective functions are for example tracking functionals

$$\tilde{J}(\theta) := \tfrac{1}{2}\big\|h(\theta, \cdot) - f(\theta, \mathbb{E}[j(\theta, \cdot, X)])\big\|_{L^2}^2$$

and nested expected values

$$\tilde{J}(\theta) := \mathbb{E}_{\mathcal{Y}}\big[f(Y, \mathbb{E}_{\mathcal{X}}[j(\theta, X)])\big].$$

Notice that such settings can not be solved by SG algorithms.

As we are aiming for a gradient based optimization scheme, we further state the derivative of the objective functional:

Lemma 2.4 (Derivative of objective function). *The gradient of the objective functional* J *is given by* $\nabla J(\theta) = \mathbb{E}\big[\boldsymbol{\delta}(\theta, X)\big]$*, where* $X \sim \mu$ *and* $\boldsymbol{\delta} : \mathcal{P} \times \mathcal{X} \to \mathbb{R}^{d_{\text{des}}}$ *denotes* $\nabla_1 j(\theta, x)$*.*

Proof. This is a direct consequence of the linearity of the expectation value and the finite-dimensional derivative of j. Integration and differentiation can be exchanged due to the Lipschitz continuity of the integrand w.r.t. the integration variable. ∎

In order to state and prove convergence results for the algorithm presented in this work, we define the norms on the used spaces as follows:

Definition 2.5 (Norms on $\mathcal{X}, \mathcal{P}$ and $\mathcal{P} \times \mathcal{X}$). *In this contribution, we will use for the norm on the underlying spaces of the parameter space $\mathcal{X}$ and the design space $\mathcal{P}$ the notation $\|\cdot\|_{\mathcal{X}}$ and $\|\cdot\|_{\mathcal{P}}$ respectively. Due to norm-equivalence in finite dimensional spaces, the norm used in the spaces $\mathcal{P}, \mathcal{X}$ does not have to be specified and can be chosen problem specific. In addition, we define on $\mathcal{P} \times \mathcal{X}$ the following metric:*

$$d\big((\theta, x), (\hat{\theta}, \hat{x})\big) := \big\|\big(\|\theta - \hat{\theta}\|_{\mathcal{P}}\,,\, \|x - \hat{x}\|_{\mathcal{X}}\big)\big\|_1 \;\forall (\theta, \hat{\theta}, x, \hat{x}) \in \mathcal{P}^2 \times \mathcal{X}^2.$$

Choosing the 1-norm in the three-dimensional space as "outer"-norm is arbitrary and could for instance - in the other extreme case - be the ∞-norm and of course could include positive weights for each individual component.

Assumption 2.6 (Regularity of the $\boldsymbol{\delta}$). *We assume $\boldsymbol{\delta} : \mathcal{P} \times \mathcal{X} \to \mathbb{R}^{d_{\text{des}}}$ to be bounded and Lipschitz continuous, i.e., there exist constants $\mathsf{C}_{\boldsymbol{\delta}}, \mathsf{L}_{\boldsymbol{\delta}} \in \mathbb{R}_{>0}$ s.t.*

$$\begin{aligned} \big\|\boldsymbol{\delta}(\theta, x)\big\|_{\mathcal{P}} &\leq \mathsf{C}_{\boldsymbol{\delta}} \\ \big\|\boldsymbol{\delta}(\theta, x) - \boldsymbol{\delta}(\hat{\theta}, \hat{x})\big\|_{\mathcal{P}} &\leq \mathsf{L}_{\boldsymbol{\delta}}\big(\|\theta - \hat{\theta}\|_{\mathcal{P}} + \|x - \hat{x}\|_{\mathcal{X}}\big) \end{aligned}$$

for all $\theta, \hat{\theta} \in \mathcal{P}$ and $x, \hat{x} \in \mathcal{X}$. A sufficient condition therefore is to assume ∇j to be Lipschitz continuous in both arguments.

For the convergence analysis of Algorithm 3.1, the following assumptions on the sets $\mathcal{P}, \mathcal{X}$ and the measure μ are an important ingredient.

Assumption 2.7 (Regularity of $\mathcal{P}$, $\mathcal{X}$ and the measure μ). *The set $\mathcal{P} \subset \mathbb{R}^{d_{\text{des}}}$ is compact and convex. $\operatorname{supp}(\mu) \subset \mathcal{X}$ with $\mathcal{X} \subset \mathbb{R}^{d_{\text{par}}}$ is open and bounded. In addition, there exists $\mathsf{M}_1, \mathsf{M}_2, \mathsf{M}_3 > 0$ s.t. $\forall \varepsilon \in (0, \mathsf{M}_3)$ there exists $\mathcal{X}_\varepsilon \subset \mathcal{X}$ satisfying $\mu(\mathcal{X}_\varepsilon) \geq 1 - \mathsf{M}_1\varepsilon$ and*

$$\inf_{x \in \mathcal{X}_\varepsilon} \mu\big(B_\varepsilon(x)\big) \geq \mathsf{M}_2 \varepsilon^{d_{\text{par}}},$$

where $B_\varepsilon(x) \subset \mathcal{X}$ is an open ball with radius ε centered in $x \in \mathcal{X}$.

Remark 2.8 (Examples for Assumption 2.7). In most cases, the choice $\mathcal{X}_\varepsilon = \mathcal{X}$ is suitable, for example when $\mathcal{X}$ satisfies the uniform cone condition (cf. [1, Definitioin 4.8]). However, there exist cases where the possibility of choosing $\mathcal{X}_\varepsilon \subset \mathcal{X}$ in the condition of Assumption 2.7 allows to consider even more general measures and sets.

As example therefore, let $\mathcal{X} := \{1/n : n \in \mathbb{N}\}$ and $\mu := \sum_{k=1}^{\infty} 2^{-k}\delta_{k^{-1}}(s)$. Then, for $M_n := [n^{-1}, 1] \cap \mathcal{X}$, it holds

$$\mu(M_n) = \sum_{k=1}^{n} 2^{-k} = 1 - 2^{-n}.$$

Thus for $\varepsilon = 2^{-n}$ and $c' = 1$ we obtain $\mu(M_n) \geq 1 - c'\varepsilon$ and $\inf_{x \in M_n} \mu(B_\varepsilon(x)) \geq 2^{-n} = \varepsilon$. Since $\inf_{x \in \mathcal{X}} \mu(B_\varepsilon(x)) = \mu(B_\varepsilon(0)) = 2^{1-2^n}$, there exists no $c > 0$ such that $2^{1-2^n} \geq c2^{-n}\ \forall n \in \mathbb{N}$.

For a uniform distribution and for all $0 < p \leq \infty$, the open $p-$Balls

$$\mathcal{X}^p := \big\{x \in \mathbb{R}^{d_{\text{par}}} : \|x\|_p < 1\big\}$$

satisfy Assumption 2.7 as well. While case $1 \leq p \leq \infty$ allows for $\mathcal{X}_\varepsilon^p = \mathcal{X}^p$, for $0 < p < 1$ we first have to obtain $\mathcal{X}_\varepsilon^p$ by trimming of the spikes of $\mathcal{X}$.

3. The algorithm. To state the algorithm, we first define the projection operator which ensures the sequence of generated designs $(\theta_n)_{n\in\mathbb{N}}$ to be in the set $\mathcal{P}$.

Definition 3.1 (Orthogonal projection). *We define the – in the sense of $\|\cdot\|_{\mathcal{P}}$ – orthogonal projection onto the set $\mathcal{P}$ as follows:*

$$\mathrm{Proj}_{\mathcal{P}}(\theta) := \operatorname*{arg\,min}_{\hat{\theta}\in\mathcal{P}} \left\|\theta - \hat{\theta}\right\|_{\mathcal{P}}.$$

Note that the existence and uniqueness of $\mathrm{Proj}_{\mathcal{P}}$ is guaranteed by the projection theorem (see e.g. [3]) building on the convexity of $\mathcal{P}$ as assumed in Assumption 2.7.

Lemma 3.2 (Properties of $\mathrm{Proj}_{\mathcal{P}}$). *Let $\mathcal{P} \subset \mathbb{R}^{d_{\mathrm{des}}}$ satisfy Assumpption 2.7. Then the following holds for all $x, y \in \mathbb{R}^{d_{\mathrm{des}}}$ and $z \in \mathcal{P}$:*

(a) $(\mathrm{Proj}_{\mathcal{P}}(x) - x)^T(\mathrm{Proj}_{\mathcal{P}}(x) - z) \leq 0$,

(b) $(\mathrm{Proj}_{\mathcal{P}}(y) - \mathrm{Proj}_{\mathcal{P}}(x))^T(y - x) \geq \|\mathrm{Proj}_{\mathcal{P}}(y) - \mathrm{Proj}_{\mathcal{P}}(x)\|_{\mathcal{P}}^2 \geq 0$,

(c) $\|\mathrm{Proj}_{\mathcal{P}}(y) - \mathrm{Proj}_{\mathcal{P}}(x)\|_{\mathcal{P}} \leq \|y - x\|_{\mathcal{P}}$.

Proof. A proof of (a) can be found in [3, Thm. 1.4.1 (ii)], (b) and (c) correspond to (iii) and (ii) in [3, Prop. 1.4.1] respectively. ∎

Given θ_1, $n = 1$ and a sequence $x_1, x_2, \ldots$ of inputs, where we assume that they are realizations of the independent random variables $X_1, X_2, \ldots$ introduced in Section 2, the CSG method for the (possibly unknown) measure μ is given in Algorithm 3.1.

Algorithm 3.1 CSG method

while Termination condition not met **do**
  Sample objective function (optional):
    $j_n := j(\theta_n, x_n)$
  Sample gradient:
    $g_n := \nabla_\theta j(\theta_n, x_n)$
  Calculate weights α_k
  Calculate search direction:
    $\hat{G}_n := \frac{1}{n}\sum_{k=1}^{n} \alpha_k g_k$
  Approximation to objective function value (optional):
    $\hat{J}_n := \frac{1}{n}\sum_{k=1}^{n} \alpha_k j_k$
  Choose stepsize τ_n
  Gradient step:
    $\theta_{n+1} := \mathrm{Proj}_{\mathcal{P}}\left(\theta_n - \tau_n \hat{G}_n\right)$
  Update index:
    $n \leftarrow n + 1$
end while

3.1. Calculating the weights. The quality of the weights α_k appearing in Algorithm 3.1 greatly impacts the accuracy of the gradient approximation $\hat{G}_n$ and therefore directly influences the overall performance of the CSG method. On the other hand, a more optimal computation of the weights might be time-consuming. Since the trade-off between the time spent calculating the weights and the time gained by performing fewer gradient evaluations is heavily problem-specific, we propose four different methods for the weight-calculation in the nth step:

Exact. Following an exact nearest neighbor approximation for the integral

$$\nabla J(\theta_n) = \int_{\mathcal{X}} \nabla_\theta j(\theta_n, x)\mu(\mathrm{d}x),$$

for each $k = 1, \dots, n$ we define the set

$$M_k := \left\{x \in \mathcal{X} : d\big((\theta_n, x), (\theta_k, x_k)\big) < d\big((\theta_n, x), (\theta_j, x_j)\big) \text{ for all } j \in \{1, \dots, n\} \setminus \{k\}\right\},$$

i.e., the set of points $x \in \mathcal{X}$ such that (θ_n, x) is closer to (θ_k, x_k) than to any other previous evaluation point. Assuming that the measure μ is known, we then set $\alpha_k := \mu(M_k)$. This method has been thoroughly analyzed in [16] and yields the best possible approximation to the exact gradient, but is computationally infeasible for problems of high dimensions.

Empirical. Utilizing the properties of the empirical measure μ_n (see Remark 4.6), we may replace the exact weights mentioned above by the empirical weights

$$\alpha_k := \frac{1}{n}\sum_{i=1}^{n} 1_{M_k}(x_i) = \mu_n(M_k) \approx \mu(M_k),$$

where 1_{M_k} denotes the indicator function of the set M_k. Note that the computation of the empirical weights requires no knowledge of μ and is also feasible for high-dimensional problems, but needs many samples x_i to approximate the exact gradient with a high accuracy.

Exact hybrid. Assuming that the dimension of $\mathcal{X}$ is much smaller than the dimension of $\mathcal{P}$, we might treat the designs and parameters separately. Instead of M_k, we now consider the sets

$$\widetilde{M}_i = \left\{x \in \mathcal{X} : \|x - x_i\|_{\mathcal{X}} \le \|x - x_j\|_{\mathcal{X}} \text{ for all } j = 1, \dots, n\right\}, \quad i = 1, \dots, n.$$

The α_k are now calculated as a combination of the empirical and exact method

$$\alpha_k := \sum_{i=1}^{n} 1_{M_k}(x_i)\mu(\widetilde{M}_i). \tag{3.1}$$

Inexact hybrid. As for the exact weights, the calculation of the exact hybrid weights requires knowledge of μ. If μ is unknown, we may replace the factor $\mu(\widetilde{M}_i)$ in (3.1) by an empirical approximation. Since this only requires samples of X, which we assume to have a plenitude of, we can control the quality of this approximation through the number of samples we draw. The inexact hybrid weights are therefore calculated as follows:

$$\alpha_k := \frac{1}{\lfloor n^\beta \rfloor}\sum_{i=1}^{n} 1_{M_k}(x_{j_i}) \sum_{m=1}^{\lfloor n^\beta \rfloor} 1_{\widetilde{M}_{j_i}}(x_m),$$

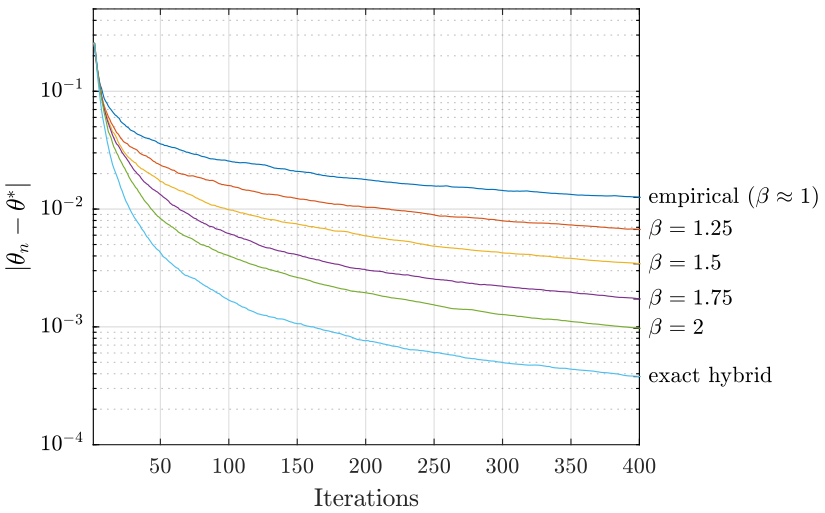

Figure 1. *Absolute error $|\theta_n - \theta^*|$ in iteration n for the setting $\mathcal{P} = [-\frac{1}{2}, \frac{1}{2}] = \mathcal{X}$, $j(\theta, x) = \frac{1}{2}(\theta - x)^2$ and $X \sim \mathcal{U}_{\mathcal{X}}$. The curves correspond to the median of 1000 runs with constant stepsizes $\tau_n = 1$ and randomized starting points in $\mathcal{P}$.*

where $\beta \geq 1$, $\lfloor n^\beta \rfloor$ is the total number of samples we have drawn until step n and x_{j_i} denote the samples where $\nabla_\theta j(\theta, x)$ has been evaluated at.

Figure 1 shows that the inexact hybrid method allows us to interpolate between the purely empirical method and the exact hybrid variant by choosing β appropriately.

Remark 3.3. In general, the nearest neighbor approximation, which is used in all methods mentioned above, worsens as the dimension of $\mathcal{P} \times \mathcal{X}$ increases (cf. [4]). Especially for problems where $\dim(\mathcal{P}) \ll \dim(\mathcal{X})$, results from Monte Carlo integration ([23]) suggest that the performance boost gained by better weight calculation starts to become negligible. The proposed CSG methods are therefore best suited for optimization problems where $\mathcal{X}$ is of small dimension when compared to $\mathcal{P}$ and the evaluation of $j(\theta, x)$ is time-consuming.

Furthermore, the metric d should be chosen problem-specific to ensure the best possible performance.

Remark 3.4 (SAG and SG as two extreme cases of the algorithm).
As stated in Definition 2.5, our metric d can be chosen as

$$d\big((\theta, x), (\hat{\theta}, \hat{x})\big) = a_1 \|\theta - \hat{\theta}\|_{\mathcal{P}} + a_2 \|x - \hat{x}\|_{\mathcal{X}},$$

where $a_1, a_2 > 0$ are arbitrary. By choosing $a_1 \gg a_2$, the nearest neighbor to (θ_n, x) is almost exclusively determined by the distance in the design variable. Hence, for the weights α_k we get $\alpha_n \approx 1$ and $\alpha_1, \ldots, \alpha_{n-1} \approx 0$, i.e., the CSG algorithm will behave very similar to the usual SG algorithm.

Analogously, choosing $a_1 \ll a_2$ will lead to a performance similar to SAG.

4. Convergence analysis.

In this section we will study the convergence of the proposed algorithm. By the matter of the randomly chosen evaluation point within the algorithm, we

will have to study probabilistic convergence behaviour in terms of "almost sure convergence". Therefore, we first state first order optimality conditions, assumptions on the regularity of the involved functions as well as the steplength τ and a suitable probability space setting.

4.1. Optimality conditions and assumptions. For $h \in C^1(\mathcal{P})$ and $\mathcal{P}$ convex we have the following equivalent sufficient conditions for first order optimality:

Corollary 4.1 (Optimality conditions). *For all $\theta^* \in \mathcal{P}$ the following items are equivalent:*
(a) $-\nabla h(\theta^*)^T(\theta - \theta^*) \leq 0 \quad \forall \theta \in \mathcal{P}$
(b) $\mathcal{P}(\theta^* - t\nabla h(\theta^*)) = \theta^* \quad \forall t \geq 0.$
A point $\theta^* \in \mathcal{P}$ satisfying these conditions is called a stationary point.

Proof. The proof can be found in e.g. [16]. ∎

In order to guarantee that Algorithm 3.1 generates a convergent subsequence, the stepsizes have to be damped, i.e., $(\tau_n)_{n\in\mathbb{N}}$ has to be a null series with upper and lower bound as stated in the following Assumption. However, in contrast to the ordinary stochastic gradient decent method, if Algorithm 3.1 generates – with stepsizes satisfying $\tau_n \geq \tau > 0 \ \forall n \in \mathbb{N}$ – a convergent sequence, the limit point is a stationary point of the objective function too. This is shown in Theorem 4.11.

Assumption 4.2 (Steplength). *The steplength $(\tau_n)_{n\in\mathbb{N}}$ in Algorithm 3.1 satisfies the following: $\exists N \in \mathbb{N}$, $\underline{\mathsf{S}}, \overline{\mathsf{S}} \in \mathbb{R}_{>0}$ and $\mathsf{D} \in \left(0, \frac{1}{\max\{d_{\text{par}},2\}}\right)$ s.t.*

$$\underline{\mathsf{S}} n^{-1} \leq \tau_n \leq \overline{\mathsf{S}} n^{-1+\frac{1}{\max\{d_{\text{par}},2\}}-\mathsf{D}} \quad \forall n \in \mathbb{N}_{>N}.$$

These bounds on the steplength satisfy the conditions stated in [17, Eqns. (6) and (26)] as well as equivalently in [6, Eqn. (4.19)] in the one-dimensional case and can be seen as a higher dimensional equivalent.

In the following we assume that these assumptions are always satisfied without mentioning it explicitly.

4.2. Error in the search direction. In this subsection we analyse the error in the $n-$th iteration of the search direction $\hat{G}_n$ and the gradient of the objective functional ∇J_n. For this, we define the following random variables:

Definition 4.3 (Random variables). *For $x \in \mathcal{X}$ and $\omega \in \Omega$ the sequence of random variables $(Z_n)_{n\in\mathbb{N}}$ with $Z_n : \Omega \times \mathcal{X} \to \mathbb{R}_{\geq 0}$ is defined by*

$$Z_n(\omega, x) := \min_{k=1,\dots,n} d\big((\Theta_k(\omega), X_k(\omega)), (\Theta_n(\omega), x)\big),$$

where the designs $\Theta_k \in \mathcal{P}$ for $k > 1$ depend by their construction on the initial design Θ_1 and all "previous" random variables $X_1(\omega), \dots, X_{k-1}(\omega)$, i.e.,

$$\Theta_k(\Theta_1, X_1(\omega), \dots, X_{k-1}(\omega))$$

and thus is also a random variable. We shorten this dependency by the notation $\Theta_k(\omega)$.

This random variable fulfills the following property:

Lemma 4.4. *For μ almost all $x \in \operatorname{supp}(\mu)$*

$$\sum_{n=1}^{\infty} \mathbb{P}\left(Z_n(\cdot, x) > \varepsilon_n\right) < \infty \quad \textit{and} \quad \sum_{n=1}^{\infty} \sup_{x \in \mathcal{X}_{\varepsilon_n}} \mathbb{P}\left(Z_n(\cdot, x) > \varepsilon_n\right) < \infty,$$

with

$$\varepsilon_n := \frac{\mathsf{C}_{\boldsymbol{\delta}} \overline{\mathsf{S}}}{1 - 2^{-\frac{1}{2\max\{d_{\mathrm{par}}, 2\}}}} \cdot n^{-\frac{\mathsf{D}}{2}} + \tilde{\varepsilon}_n \quad \textit{and} \quad \tilde{\varepsilon}_n := n^{\frac{\mathsf{D}}{2} - \frac{1}{2\max\{d_{\mathrm{par}}, 2\}}}. \tag{4.1}$$

Therein, $\mathsf{C}_{\boldsymbol{\delta}}$ is defined in Assumption 2.6, $\mathcal{X}_{\varepsilon_n}$ in Assumption 2.7 and $\overline{\mathsf{S}}$,D in Assumption 4.2.

Proof. We first define $i_0 \in \mathbb{N}$ as an auxiliary index as follows:

$$i_0 := \lceil n - a_n + 1 \rceil \quad \text{with} \quad a_n := n^{1 + \frac{\mathsf{D}}{2} - \frac{1}{\max\{d_{\mathrm{par}}, 2\}}}.$$

By construction, we have

$$\begin{aligned}
\mathbb{P}(Z_n(\cdot, x) \geq \varepsilon_n) &\leq \mathbb{P}\left(\min_{k = i_0, \dots, n} d((\Theta_k, X_k), (\Theta_n, x))) \geq \varepsilon_n\right) \\
&\leq \mathbb{P}\left(\sum_{i=i_0}^{n-1} \|\tau_i \hat{G}_i\|_{\mathcal{P}} + \min_{k = i_0, \dots, n} \|X_k - x\|_{\mathcal{X}} \geq \varepsilon_n\right) \\
&\leq \mathbb{P}\left(\mathsf{C}_{\boldsymbol{\delta}} \sum_{i=i_0}^{n-1} \tau_i + \min_{k = i_0, \dots, n} \|X_k - x\|_{\mathcal{X}} \geq \varepsilon_n\right).
\end{aligned}$$

Observe that for $n > 2$ we obtain for all $\kappa \in (0, 1)$

$$\begin{aligned}
\sum_{i=i_0}^{n-1} \frac{1}{i^{\kappa}} &\leq \int_{i_0 - 1}^{n} \tfrac{1}{s^{\kappa}} \, \mathrm{d}s = \frac{1}{1-\kappa} \cdot \left(n^{1-\kappa} - (\lceil n - a_n \rceil)^{1-\kappa}\right) \leq \frac{1}{1-\kappa} \cdot \left(n^{1-\kappa} - (n - a_n)^{1-\kappa}\right) \\
&= \frac{1}{1-\kappa} \cdot \left(\frac{n}{n^{\kappa}} - \frac{n - a_n}{(n - a_n)^{\kappa}}\right) = \frac{n}{1-\kappa} \cdot \left(\frac{(n - a_n)^{\kappa} - n^{\kappa}}{n^{\kappa} \cdot (n - a_n)^{\kappa}}\right) + \frac{a_n}{(1-\kappa)(n - a_n)^{\kappa}} \\
&= \frac{n}{1-\kappa} \cdot \left(\frac{n^{\kappa}(1 - a_n/n)^{\kappa} - n^{\kappa}}{n^{\kappa} \cdot (n - a_n)^{\kappa}}\right) + \frac{a_n}{(1-\kappa)(n - a_n)^{\kappa}}.
\end{aligned}$$

Applying Bernoulli's inequality in the first term, we conclude

$$\begin{aligned}
\sum_{i=i_0}^{n-1} \frac{1}{i^{\kappa}} &\leq \frac{n}{1-\kappa} \cdot \left(\frac{n^{\kappa}\left(1 - \kappa \cdot \frac{a_n}{n}\right) - n^{\kappa}}{n^{\kappa} \cdot (n - a_n)^{\kappa}}\right) + \frac{a_n}{(1-\kappa)(n - a_n)^{\kappa}} \\
&= \frac{1}{1-\kappa} \cdot \left(\frac{-\kappa a_n}{(n - a_n)^{\kappa}}\right) + \frac{a_n}{(1-\kappa)(n - a_n)^{\kappa}} \\
&= \frac{a_n}{(n - a_n)^{\kappa}} = \frac{a_n}{n^{\kappa}} (1 - \tfrac{a_n}{n})^{-\kappa}.
\end{aligned} \tag{4.2}$$

Combining Assumption 4.2 and (4.2) yields

$$\sum_{i=i_0}^{n-1} \tau_i \le \overline{\mathsf{S}} \sum_{i=i_0}^{n-1} \frac{1}{i^{1+\mathsf{D}-\frac{1}{\max\{d_{\mathrm{par}},2\}}}} \le \overline{\mathsf{S}} \frac{a_n}{n^{\kappa}} (1-\tfrac{a_n}{n})^{-\kappa},$$

with $\mathsf{D} \in \left(0, \frac{1}{\max\{d_{\mathrm{par}},2\}}\right)$ and $\kappa := 1 + \mathsf{D} - \frac{1}{\max\{d_{\mathrm{par}},2\}} \in (0,1)$. Hence, for $n \ge 2$ we obtain

$$\left(1-\frac{a_n}{n}\right)^{-\kappa} = \left(1 - n^{\frac{\mathsf{D}}{2} - \frac{1}{\max\{d_{\mathrm{par}},2\}}}\right)^{-\kappa} \le \left(1 - n^{-\frac{1}{2\max\{d_{\mathrm{par}},2\}}}\right)^{-\kappa} \le \left(1 - 2^{-\frac{1}{2\max\{d_{\mathrm{par}},2\}}}\right)^{-1}.$$

Collecting these results, we see

$$\sum_{i=i_0}^{n-1} \tau_i \le \frac{\overline{\mathsf{S}}}{1 - 2^{-\frac{1}{2\max\{d_{\mathrm{par}},2\}}}} \frac{a_n}{n^{1+\mathsf{D}-\frac{1}{\max\{d_{\mathrm{par}},2\}}}} \le \frac{\overline{\mathsf{S}}}{1 - 2^{-\frac{1}{2\max\{d_{\mathrm{par}},2\}}}} n^{-\frac{\mathsf{D}}{2}}.$$

Consequently,

$$\sum_{i=i_0}^{n-1} \|\tau_i \hat{G}_i\|_{\mathcal{P}} \le \frac{\mathsf{C}_{\delta}\overline{\mathsf{S}}}{1 - 2^{-\frac{1}{2\max\{d_{\mathrm{par}},2\}}}} n^{-\frac{\mathsf{D}}{2}} = \varepsilon_n - \tilde{\varepsilon}_n.$$

By Assumption 2.7, $\mu(\mathcal{X} \setminus \mathcal{X}_{\varepsilon_n}) \to 0$. Hence, for μ almost all $x \in \mathrm{supp}(\mu)$, there exists $n \in \mathbb{N}$ large enough, such that $x \in \mathcal{X}_{\varepsilon_n}$. Therefore,

$$\begin{aligned}
&\mathbb{P}(Z_n(\cdot,x) \ge \varepsilon_n) \\
&\quad \le \mathbb{P}\left(\min_{k=i_0,\dots,n-1} \|X_k - x\|_{\mathcal{X}} \ge \tilde{\varepsilon}_n\right) \\
&\quad \le \mathbb{P}\Big(\|X_k - x\|_{\mathcal{X}} \ge \tilde{\varepsilon}_n \ \forall k \in \{i_0,\dots,n-1\}\Big) \\
&\quad = \prod_{k=i_0}^{n-1} \mathbb{P}\Big(\|X_k - x\|_{\mathcal{X}} \ge \tilde{\varepsilon}_n\Big) = \prod_{k=i_0}^{n-1} \big(1 - \mu\big(B_{\tilde{\varepsilon}_n}(x)\big)\big) \\
&\quad \le \Big(1 - \min\left\{\mathsf{M}_2 (\tilde{\varepsilon}_n)^{d_{\mathrm{par}}}, 1\right\}\Big)^{a_n}.
\end{aligned}$$

As $\tilde{\varepsilon}_n \to 0$, there exists $N \in \mathbb{N}$ s.t. for $n \ge N$ we obtain

$$\mathbb{P}(Z_n(\cdot,x) \ge \varepsilon_n) \le \left(1 - \mathsf{M}_2 n^{\frac{d_{\mathrm{par}}\,\mathsf{D}}{2} - \frac{d_{\mathrm{par}}}{2\max\{2,d_{\mathrm{par}}\}}}\right)^{a_n}.$$

For simplicity, we define

$$c_1 := \frac{d_{\mathrm{par}}\,\mathsf{D}}{2} - \frac{d_{\mathrm{par}}}{2\max\{2,d_{\mathrm{par}}\}}$$

and recall that $\log(1-x) \le -x$ for all $x < 1$. Since $c_1 < 0$, for n large enough it holds

$$\left(1 - \mathsf{M}_2 n^{\frac{d_{\text{par}}\,\mathsf{D}}{2} - \frac{d_{\text{par}}}{2\max\{2,d_{\text{par}}\}}}\right)^{a_n} = (1 - \mathsf{M}_2 n^{c_1})^{a_n} = \exp\left(a_n \log\left(1 - \mathsf{M}_2 n^{c_1}\right)\right)$$

$$\le \exp\left(-a_n \mathsf{M}_2 n^{c_1}\right) = \exp\left(-\mathsf{M}_2 n^{1+\frac{\mathsf{D}}{2} - \frac{1}{\max\{2,d_{\text{par}}\}} + \frac{d_{\text{par}}\,\mathsf{D}}{2} - \frac{d_{\text{par}}}{2\max\{2,d_{\text{par}}\}}}\right)$$

$$= \begin{cases} \exp\left(-\mathsf{M}_2 n^{1+\mathsf{D}-\frac{1}{2}-\frac{1}{4}}\right) & d_{\text{par}} = 1 \\ \exp\left(-\mathsf{M}_2 n^{1+\frac{\mathsf{D}}{2} - \frac{1}{d_{\text{par}}} + \frac{d_{\text{par}}\,\mathsf{D}}{2} - \frac{1}{2}}\right) & d_{\text{par}} \ge 2 \end{cases} \le \exp\left(-\mathsf{M}_2 n^{\mathsf{D}}\right).$$

Recall that there is $N \in \mathbb{N}$ such that $\exp(-x) \le x^{-\frac{2}{\mathsf{D}}}$ for all $x \ge N$. It follows that for all n large enough: $\exp(-\mathsf{M}_2 n^{\mathsf{D}}) \le \mathsf{M}_2^{-\frac{2}{\mathsf{D}}} n^{-2}$. Hence,

$$\sum_{n=N}^{\infty} \left(1 - \mathsf{M}_2 n^{\frac{d_{\text{par}}\,\mathsf{D}}{2} - \frac{d_{\text{par}}}{2\max\{2,d_{\text{par}}\}}}\right)^{a_n} \le \sum_{n=N}^{\infty} \exp\left(-\mathsf{M}_2 n^{\mathsf{D}}\right) \le \sum_{n=N}^{\infty} \mathsf{M}_2^{-\frac{2}{\mathsf{D}}} n^{-2}$$

and thus

$$\sum_{n=1}^{\infty} \mathbb{P}\left(Z_n(\cdot, x) > \varepsilon_n\right) < \infty.$$

Finally, note that Assumption 2.7 gives

$$\sup_{x \in \mathcal{X}^{\varepsilon_n}} \mathbb{P}(Z_n(\cdot, x) \ge \varepsilon_n) \le \left(1 - c \cdot \frac{n^{\frac{\mathsf{D}}{2}}}{a_n}\right)^{a_n} = \left(1 - c \cdot n^{\frac{1}{\max\{d_{\text{par}},2\}} - 1}\right)^{a_n}$$

with $c > 0$. By the same steps as above, we obtain

$$\sum_{n=1}^{\infty} \sup_{x \in \mathcal{X}_{\varepsilon_n}} \mathbb{P}\left(Z_n(\cdot, x) > \varepsilon_n\right) < \infty.$$

As a direct consequence of the latter result we get almost sure convergence.

Corollary 4.5. *For μ almost all $x \in \operatorname{supp}(\mu)$*

$$Z_n(\cdot, x) \xrightarrow{a.s.} 0 \quad \textit{for} \quad n \to \infty.$$

Proof. The result follows by Lemma 4.4 and the Borel-Cantelli Lemma (see for example Theorem 2.7 in [11]). ∎

Remark 4.6 (Empirical distribution). The empirical measure defined as

$$\mu_n := \frac{1}{n} \sum_{i=1}^{n} \delta_{X_i} \tag{4.3}$$

satisfies $\mu_n \Rightarrow \mu$ as $n \to \infty$ almost surely, see [20, Theorem 3]. Here $\Rightarrow$ denotes the weak convergence of measures which is the weak-* convergence in dual space theory, i.e.,

$$\mu_n \Rightarrow \mu \quad \text{iff} \quad \int_{\mathcal{X}} f(x)\mu_n(dx) \to \int_{\mathcal{X}} f(x)\mu(dx) \quad \forall f \in C_b(\mathcal{X}, \mathbb{R}).$$

See for instance [5] for the empirical distribution and [8, Section 7.3] for a functional analytical perspective on weak-* convergence in the discussed function space setting.

Since this property of μ_n is all we need in the following proofs and since the measures

$$\mu_n^{\mathrm{eh}} := \sum_{i=1}^{n} \delta_{X_i} \mu(\widetilde{M}_i) \quad \text{and} \quad \mu_n^{\mathrm{ih}} := \sum_{i=1}^{n} \delta_{X_{j_i}} \mu_{\lfloor n^\beta \rfloor}(\widetilde{M}_{j_i}) \ ,$$

corresponding to exact hybrid weights and inexact hybrid weights respectively, satisfy $\mu_n^{\mathrm{eh}} \Rightarrow \mu$ and $\mu_n^{\mathrm{ih}} \Rightarrow \mu$ as well, we will w.l.o.g. work with empirical weights only.

Thus, due to the Lipschitz continuity of $\boldsymbol{\delta}$ as defined in Theorem 2.6, the expected value $\nabla J(\theta) = \mathbb{E}[\boldsymbol{\delta}(\theta, X)]$ is for $n \to \infty$ better and better approximated by $\hat{G}_n$:

Theorem 4.7 (Error in gradient approximation).
The norm of the difference between the search direction $\hat{G}_n$ and the gradient of the objective functional $\mathbb{E}[\boldsymbol{\delta}(\Theta_n, X)]$ vanishes for $n \to \infty$, i.e.,

$$\|\hat{G}_n - \mathbb{E}[\boldsymbol{\delta}(\Theta_n, X)]\|_{\mathcal{P}} \xrightarrow{a.s.} 0 \quad \textit{and} \quad \lim_{n\to\infty} \mathbb{E}\left[\|\hat{G}_n - \mathbb{E}[\boldsymbol{\delta}(\Theta_n, X)]\|_{\mathcal{P}}\right] = 0.$$

Proof. For $x \in \mathrm{supp}(\mu)$ define

$$k^n(\omega; x) := \operatorname*{arg\,min}_{k=1,\dots,n} d\big((\Theta_k(\omega), X_k(\omega)), (\Theta_n(\omega), x)\big).$$

For $\hat{G}_n$ as generated by Algorithm 3.1 with $n \in \mathbb{N}$ arbitrary but fixed the following holds:

$$\begin{aligned}
&\|\hat{G}_n - \mathbb{E}[\boldsymbol{\delta}(\Theta_n, X)]\|_{\mathcal{Y}} \\
&\quad = \left\| \sum_{i=1}^{n} \int_{\mathcal{X}} \delta_{k^n(\omega;x)}(i) \boldsymbol{\delta}(\Theta_i(\omega), x_i) \mu_n(\mathrm{d}x) - \int_{\mathcal{X}} \boldsymbol{\delta}(\Theta_n(\omega), x) \mu(\mathrm{d}x) \right\|_{\mathcal{Y}} \\
&\quad \le \left\| \int_{\mathcal{X}} \sum_{i=1}^{n} \delta_{k^n(\omega;x)}(i) \boldsymbol{\delta}(\Theta_i(\omega), x_i) - \boldsymbol{\delta}(\Theta_n(\omega), x) \mu_n(\mathrm{d}x) \right\|_{\mathcal{Y}} \\
&\qquad + \left\| \int_{\mathcal{X}} \boldsymbol{\delta}(\Theta_n(\omega), x) \mu_n(\mathrm{d}x) - \int_{\mathcal{X}} \boldsymbol{\delta}(\Theta_n(\omega), x) \mu(\mathrm{d}x) \right\|_{\mathcal{Y}} \\
&\quad \le \mathsf{L}_{\boldsymbol{\delta}} \int_{\mathcal{X}} Z_n(\omega, x) \mu_n(\mathrm{d}x) + \left\| \int_{\mathcal{X}} \boldsymbol{\delta}(\Theta_n(\omega), x) \mu_n(\mathrm{d}x) - \int_{\mathcal{X}} \boldsymbol{\delta}(\Theta_n(\omega), x) \mu(\mathrm{d}x) \right\|_{\mathcal{Y}} ,
\end{aligned} \tag{4.4}$$

where μ_n is the empirical measure given in Remark (4.3) and $\mathsf{L}_{\boldsymbol{\delta}}$ the Lipschitz constant defined in Assumption 2.6. We need to prove that both terms in (4.4) vanish for $n \to \infty$.

For the first term, the uniform (in n) Lipschitz continuity of $Z_n(\omega, \cdot)$ yields

$$\begin{aligned}
\int_{\mathcal{X}} Z_n(\omega, x) \mu_n(\mathrm{d}x) &= \int_{\mathcal{X}} Z_n(\omega, x) \mu(\mathrm{d}x) + \int_{\mathcal{X}} Z_n(\omega, x) \mu_n(\mathrm{d}x) - \int_{\mathcal{X}} Z_n(\omega, x) \mu(\mathrm{d}x) \\
&\le \int_{\mathcal{X}} Z_n(\omega, x) \mu(\mathrm{d}x) + \mathsf{L}_Z d_W(\mu_n, \mu),
\end{aligned}$$

where d_W denotes the Wasserstein distance of the measure μ_n and μ (see [10]). Since $\mathcal{X}$ is bounded, [10, Theorem 6] gives that the Wasserstein distance metrizices the weak topology on the set of probability measures on $\mathcal{X}$. Since $\mu_n \Rightarrow \mu$ almost surely, this gives $d_W(\mu_n, \mu) \to 0$ almost surely. Furthermore, by Assumption 2.6, there exists $C > 0$ s.t. $0 \leq Z_n \leq C$. Using Corollary 4.5, we obtain $Z_n(\omega, x) \to 0$ for almost all $\omega \in \Omega$. Therefore, Lebesgue's dominated convergence theorem yields

$$\int_{\mathcal{X}} Z_n(\omega, x)\mu(\mathrm{d}x) \to 0 \quad \text{for almost all } \omega \in \Omega.$$

In order to show that the second part of (4.4) vanishes, observe that

$$\begin{aligned}
&\left\| \int_{\mathcal{X}} \boldsymbol{\delta}(\Theta_n(\omega), x)\mu_n(\mathrm{d}x) - \int_{\mathcal{X}} \boldsymbol{\delta}(\Theta_n(\omega), x')\mu(\mathrm{d}x') \right\|_{\mathcal{P}} \\
&\quad = \left\| \int_{\mathcal{X}\times\mathcal{X}} \boldsymbol{\delta}(\Theta_n(\omega), x) - \boldsymbol{\delta}(\Theta_n(\omega), x)Q_n(d(x, x')) \right\|_{\mathcal{P}} \\
&\quad \leq \mathsf{L}_{\boldsymbol{\delta}} \int_{\mathcal{X}\times\mathcal{X}} \|x - x'\|_{\mathcal{X}} Q_n(d(x, x')),
\end{aligned}$$

where $Q_n(\cdot \times \mathcal{X}) = \mu_n$ and $Q_n(\mathcal{X} \times \cdot) = \mu$ is an arbitrary but fixed coupling of μ_n and μ. By taking the infimum of all such couplings, we again obtain the Wasserstein distance d_W of the measure μ_n and μ, i.e.,

$$\left\| \int_{\mathcal{X}} \boldsymbol{\delta}(\Theta_n(\omega), x)\mu_n(\mathrm{d}x) - \int_{\mathcal{X}} \boldsymbol{\delta}(\Theta_n(\omega), x)\mu(\mathrm{d}x) \right\|_{\mathcal{P}} \leq \mathsf{L}_{\boldsymbol{\delta}} d_W(\mu_n, \mu). \tag{4.5}$$

By the same arguments as mentioned earlier, $d_W(\mu_n, \mu) \to 0$ almost surely. Combining all the above facts gives

$$\|\hat{G}_n - \mathbb{E}[\boldsymbol{\delta}(\Theta_n, X)]\|_{\mathcal{P}} \to 0$$

almost surely. Since the above quantities are bounded, the almost sure convergence also implies the convergence in expectation via Lebesgue's dominated convergence theorem. ∎

Remark 4.8. Due to the regularity of J, we can show

$$\|\hat{J}_n - J(\theta_n)\|_{\mathcal{P}} \to 0$$

analogously to the proof of Theorem 4.7.

Theorem 4.9 (Sum of error in gradient approximation). *The expectation value of the summed norm of the difference between the search direction $\hat{G}_n$ and the gradient of the reduced objective functional ∇J weighted by the respected stepsize τ_n vanishes for $n \to \infty$, i.e.,*

$$\sum_{n=1}^{\infty} \tau_n \mathbb{E}\left[\|\hat{G}_n - \mathbb{E}[\boldsymbol{\delta}(\Theta, X)]\|_{\mathcal{P}} \right] < \infty. \tag{4.6}$$

Proof. Recall from the proof of Theorem 4.7 that

$$\mathbb{E}\left[\|\hat{G}_n - \mathbb{E}[\boldsymbol{\delta}(\Theta_n, X)]\|_{\mathcal{P}}\right] \leq \mathsf{L}_{\boldsymbol{\delta}}\mathbb{E}\left[\int_{\mathcal{X}} Z_n(\omega, x)\mu_n(\mathrm{d}x)\right]$$

$$+ \mathbb{E}\left[\left\|\int_{\mathcal{X}} \boldsymbol{\delta}(\Theta_n(\omega), x)\mu_n(\mathrm{d}x) - \int_{\mathcal{X}} \boldsymbol{\delta}(\Theta_n(\omega), x)\mu(\mathrm{d}x)\right\|_{\mathcal{P}}\right]. \tag{4.7}$$

We start with deriving an upper bound for the first term on the right hand side of the latter inequality. Recall the definition of $\tilde{\varepsilon}_n$ in Lemma 4.4, i.e.,

$$\tilde{\varepsilon}_n := n^{-\frac{1}{\max\{2, d_{\mathrm{par}}\}} + \frac{\mathsf{D}}{2}}$$

with D as defined in Assumption 4.2. Then, analogue to Lemma 4.4 (cf. the proof and the notation there), together with

$$D := \sup_{(\tilde{\theta},\tilde{x}),(\hat{\theta},\hat{x})\in\mathcal{P}\times\mathcal{X}} d((\tilde{\theta},\tilde{x}),(\hat{\theta},\hat{x})),$$

we obtain the following estimate:

$$\begin{aligned}
\mathbb{E}\left[\int_{\mathcal{X}} Z_n(\cdot, x)\mu_n(\mathrm{d}x)\right] &= \mathbb{E}\left[\int_{\mathcal{X}} Z_n(\cdot, x)1_{Z_n(\cdot,x)\leq\tilde{\varepsilon}_n}(x) + Z_n(\cdot, x)1_{Z_n(\cdot,x)>\tilde{\varepsilon}_n}(x)\mu_n(\mathrm{d}x)\right] \\
&\leq \tilde{\varepsilon}_n + D\mathbb{E}\left[\int_{\mathcal{X}} 1_{Z_n(\cdot,x)>\tilde{\varepsilon}_n}(x)\mu_n(\mathrm{d}x)\right] \\
&\leq \tilde{\varepsilon}_n + D\mathbb{E}\left[\int_{\mathcal{X}} \prod_{k=1}^{n} 1_{d((\Theta_k(\cdot),X_k(\cdot)),(\Theta_n(\cdot),x))>\tilde{\varepsilon}_n}\mu_n(\mathrm{d}x)\right].
\end{aligned}$$

Setting $i_0 := \lceil n - a_n + 1\rceil$ as in the proof of Lemma 4.4 yields

$$\mathbb{E}\left[\int_{\mathcal{X}} Z_n(\cdot, x)\mu_n(\mathrm{d}x)\right] \leq \tilde{\varepsilon}_n + D\mathbb{E}\left[\int_{\mathcal{X}} \prod_{k=i_0}^{n} 1_{d((\Theta_k(\cdot),X_k(\cdot)),(\Theta_n(\cdot),x))>\tilde{\varepsilon}_n}\mu_n(\mathrm{d}x)\right].$$

Since μ_n is the empirical maesure as defined in (4.3) and due to the linearity of $\mathbb{E}$, we obtain

$$\begin{aligned}
\mathbb{E}\left[\int_{\mathcal{X}} Z_n(\cdot, x)\mu_n(\mathrm{d}x)\right] &= \tilde{\varepsilon}_n + \frac{D}{n}\sum_{i=1}^{n}\mathbb{E}\left[\prod_{\substack{k=i_0\\k\neq i}}^{n} 1_{d((\Theta_k(\cdot),X_k(\cdot)),(\Theta_n(\cdot),X_i(\cdot)))>\tilde{\varepsilon}_n}\right] \\
&= \tilde{\varepsilon}_n + \frac{D}{n}\sum_{i=1}^{n}\prod_{\substack{k=i_0\\k\neq i}}^{n}\mathbb{P}\big(d((\Theta_k(\cdot), X_k(\cdot)), (\Theta_n(\cdot), X_i(\cdot))) > \tilde{\varepsilon}_n\big),
\end{aligned}$$

Where we used the independency of all $(X_i)_{i\in\mathbb{N}}$. Finally, applying Fubini's theorem results in

$$\mathbb{E}\left[\int_{\mathcal{X}} Z_n(\cdot, x)\mu_n(\mathrm{d}x)\right] = \tilde{\varepsilon}_n + \frac{D}{n}\sum_{i=1}^{n}\prod_{\substack{k=i_0\\k\neq i}}^{n}\int_{\mathcal{X}}\mathbb{P}\big(d((\Theta_k(\cdot), X_k(\cdot)), (\Theta_n(\cdot), x)) > \tilde{\varepsilon}_n\big)\mu(dx). \tag{4.8}$$

Let $\mathcal{X}_{\tilde{\varepsilon}_n} \subset \mathcal{X}$ be the set given in Assumption 2.7. Following the same argumentation as in the proof of Lemma 4.4, we obtain

$$\begin{aligned}
&\int_{\mathcal{X}} \mathbb{P}\big(d((\Theta_k(\cdot), X_k(\cdot)), (\Theta_n(\cdot), x)) > \tilde{\varepsilon}_n\big)\mu(dx) \\
&\quad = \int_{\mathcal{X}\setminus\mathcal{X}_{\tilde{\varepsilon}_n}} \mathbb{P}\big(d((\Theta_k(\cdot), X_k(\cdot)), (\Theta_n(\cdot), x)) > \tilde{\varepsilon}_n\big)\mu(dx) \\
&\qquad + \int_{\mathcal{X}_{\tilde{\varepsilon}_n}} \mathbb{P}\big(d((\Theta_k(\cdot), X_k(\cdot)), (\Theta_n(\cdot), x)) > \tilde{\varepsilon}_n\big)\mu(dx) \\
&\quad \leq c'\tilde{\varepsilon}_n + \sup_{x\in\mathcal{X}_{\tilde{\varepsilon}_n}} \mathbb{P}\big(d((\Theta_k(\cdot), X_k(\cdot)), (\Theta_n(\cdot), x)) > \tilde{\varepsilon}_n\big) \leq c'\tilde{\varepsilon}_n + \left(1 - c\cdot\tfrac{n^{\frac{\mathsf{D}}{2}}}{a_n}\right)^{a_n},
\end{aligned}$$

with a_n defined as in Lemma 4.4 and $c > 0$. Utilizing $\log(1-x) \leq -x$ for all $x < 1$ shows

$$\begin{aligned}
\left(1 - c\cdot\tfrac{n^{\frac{\mathsf{D}}{2}}}{a_n}\right)^{a_n} &= \exp\left(a_n \log\left(1 - c\cdot n^{\frac{1}{\max\{d_{\mathrm{par}},2\}}-1}\right)\right) \leq \exp\left(-ca_n\cdot n^{\frac{\mathsf{D}}{2}}\right) \\
&= \exp\left(-c\cdot n^{1+\mathsf{D}-\frac{1}{\max\{d_{\mathrm{par}},2\}}}\right) \leq \exp\big(-c\cdot n^{\mathsf{D}}\big) \leq \tilde{\varepsilon}_n
\end{aligned}$$

for n large enough. Therefore, we have

$$\int_{\mathcal{X}} \mathbb{P}\big(d((\Theta_k(\cdot), X_k(\cdot)), (\Theta_n(\cdot), x)) > \tilde{\varepsilon}_n\big)\mu(\mathrm{d}x) \leq (c'+1)\tilde{\varepsilon}_n.$$

Inserting into (4.8) yields

$$\mathbb{E}\left[\int_{\mathcal{X}} Z_n(\cdot, x)\mu_n(\mathrm{d}x)\right] \leq \tilde{\varepsilon}_n + D\big((c'+1)\tilde{\varepsilon}_n\big)^{n-i_0+1} \leq \bar{c}\cdot\tilde{\varepsilon}_n \tag{4.9}$$

for n large enough and some $\bar{c} > 0$.

Now, in order to bound the second term in (4.7), recall from (4.5) that

$$\mathbb{E}\left[\left\|\int_{\mathcal{X}} \boldsymbol{\delta}(\theta_n, x)\mu_n(dx) - \boldsymbol{\delta}(\theta_n, x)\mu(dx)\right\|_{\mathcal{P}}\right] \leq \mathsf{L}_{\boldsymbol{\delta}}\cdot\mathbb{E}\left[d_W(\mu_n, \mu)\right],$$

where d_W is the Wasserstein distance. By [9, Thm. 1 for $q = 3$ and $p = 1$], for all $d_{\mathrm{par}} \geq 1$ there exists $\tilde{C}(d_{\mathrm{par}}) \in \mathbb{R}_{>0}$ s.t.:

$$\begin{aligned}
\mathbb{E}\left[d_W(\mu_n, \mu)\right] &\leq \hat{C}(d_{\mathrm{par}})\cdot M_3 \begin{cases} n^{-\frac{1}{2}}, & d_{\mathrm{par}} = 1, \\ n^{-\frac{1}{2}}\log(1+n), & d_{\mathrm{par}} = 2, \\ n^{-\frac{1}{d_{\mathrm{par}}}}, & d_{\mathrm{par}} \geq 3, \end{cases} \\
&\leq \hat{C}(d_{\mathrm{par}})\cdot M_3 \cdot n^{-\frac{1}{\max\{d_{\mathrm{par}},2\}}}\log(1+n), \qquad (4.10)
\end{aligned}$$

with $M_3 := \big(\int_{\mathcal{X}} \|x\|_{\mathcal{X}}^3 \mu(dx)\big)^{1/3}$.

Substituting (4.9) and (4.10) into (4.7) yields

(4.11)
$$\sum_{n=N}^{\infty} \tau_n \mathbb{E}\left[\|\hat{G}_n - \mathbb{E}[\boldsymbol{\delta}(\theta, X)]\|_{\mathcal{P}}\right] \leq \bar{c} \sum_{n=N}^{\infty} \tau_n \tilde{\varepsilon}_n + \hat{C}(d_{\text{par}}) M_3 \sum_{n=N}^{\infty} \tau_n n^{-\frac{1}{\max\{d_{\text{par}},2\}}} \log(1+n)$$

for $N \in \mathbb{N}$ large enough. By Assumption 4.2, we have

$$\tau_n \leq \overline{\mathsf{S}} n^{-1-\mathsf{D}+\frac{1}{\max\{d_{\text{par}},2\}}},$$

which, when inserted into (4.11), gives

$$\sum_{n=N}^{\infty} \tau_n \mathbb{E}\left[\|\hat{G}_n - \mathbb{E}[\boldsymbol{\delta}(\theta, X)]\|_{\mathcal{P}}\right] \leq \bar{c}\overline{\mathsf{S}} \sum_{n=N}^{\infty} n^{-1-\frac{\mathsf{D}}{2}} + \hat{C}(d_{\text{par}}) M_3 \overline{\mathsf{S}} \sum_{n=N}^{\infty} n^{-1-\mathsf{D}} \log(1+n),$$

showing that

$$\sum_{n=1}^{\infty} \tau_n \mathbb{E}\left[\|\hat{G}_n - \mathbb{E}[\boldsymbol{\delta}(\theta, X)]\|_{\mathcal{P}}\right] \leq \infty.$$

■

Before we can present our main result, we collect a few auxiliary results.

Lemma 4.10 (Collection of auxiliary results).

(a) *The objective functional value in iteration $n \in \mathbb{N}$ satisfies*

$$J_{n+1} - J_n \leq -\tfrac{1}{\tau_n}\|\theta_{n+1} - \theta_n\|_{\mathcal{P}}^2 + \phi_n,$$

where $\phi_n := \tau_n \|\nabla J_n - \hat{G}_n\|_{\mathcal{P}} \|\hat{G}_n\|_{\mathcal{P}} + \tau_n^2 C \|\hat{G}_n\|_{\mathcal{P}}^2$ and $C \geq 0$ denotes a constant depending only on the Lipschitz constants and suprema of the involved functions.

(b) *For ϕ_n as defined above, it holds $\sum_{n=1}^{\infty} \mathbb{E}[\phi_n] < \infty$.*

(c) *For all $t \geq 0$, we have*

$$\|\text{Proj}_{\mathcal{P}}(\theta_n - t\hat{G}_n) - \theta_n\|_{\mathcal{P}} \leq \tfrac{t}{\tau_n}\|\theta_{n+1} - \theta_n\|_{\mathcal{P}}.$$

Proof. Assertions (a), (b) and (c) correspond to Lemma 16, Corollary 17 and Lemma 18 in [16]. Note that, by Theorem 4.9, the proofs given therein can be carried over to our setting as well. ■

Theorem 4.11 (Main theorem). *Let $(\theta_n)_{n\in\mathbb{N}}$ be generated by Algorithm 3.1 with weights calculated by one of the methods mentioned in Section 3.1. Then there exists a sub-sequence $(\theta_{n_k})_{k\in\mathbb{N}}$ converging to a stationary point, i.e.,*

$$\liminf_{n\to\infty} \mathbb{E}\left[\|\text{Proj}_{\mathcal{P}}(\theta_n - t\nabla J_n) - \theta_n\|_{\mathcal{P}}^2\right] = 0 \quad \textit{for all } t \geq 0.$$

On the other hand, assume the time-step series $(\tau_n)_{n\in\mathbb{N}}$ satisfies $\tau_n \geq \tau$ for all $n \in \mathbb{N}$ and some $\tau > 0$. Let further $(x_n)_{n\in\mathbb{N}}$ be dense in $\mathcal{X}$ and assume $(\theta_n)_{n\in\mathbb{N}}$ converges to $\theta^* \in \mathcal{P}$. Then θ^* is a stationary point of J, i.e.

$$\|\text{Proj}_{\mathcal{P}}(\theta^* - t\nabla J(\theta^*)) - \theta^*\|_{\mathcal{P}}^2 = 0 \quad \textit{for all } t \geq 0.$$

Proof. To prove the first part, we show

$$\sum_{n=1}^{\infty} \tau_n \mathbb{E}\left[\|\mathrm{Proj}_{\mathcal{P}}(\theta_n - t\nabla J_n) - \theta_n\|_{\mathcal{P}}^2\right] < \infty \quad \text{for all } t \geq 0. \tag{4.12}$$

By the assumed compactness of $\mathcal{P}$ and regularity of J, we have

$$J_{\inf} := \inf_{\theta \in \mathcal{P}} J(\theta) > -\infty.$$

For arbitrary $N \in \mathbb{N}$, Lemma 4.10 (a) gives

$$J_{\inf} - J_1 \leq \mathbb{E}[J_{N+1}] - J_1 = \sum_{n=1}^{N} \mathbb{E}[J_{n+1} - J_n] \leq \sum_{n=1}^{N} \left(-\frac{1}{\tau_n} \mathbb{E}\left[\|\theta_{n+1} - \theta_n\|_{\mathcal{P}}^2\right] + \mathbb{E}[\phi_n] \right).$$

Rearranging terms and utilizing Lemma 4.10 (b) yields

$$\sum_{n=1}^{\infty} \frac{1}{\tau_n} \mathbb{E}\left[\|\theta_{n+1} - \theta_n\|_{\mathcal{P}}^2\right] \leq J_1 - J_{\inf} + \sum_{n=1}^{\infty} \mathbb{E}[\phi_n] < \infty. \tag{4.13}$$

By Lemma 3.2 (c) and Lemma 4.10 (c), we obtain

$$\begin{aligned}
&\|\mathrm{Proj}_{\mathcal{P}}(\theta_n - t\nabla J_n) - \theta_n\|_{\mathcal{P}}^2 \\
&\quad \leq \left(\|\mathrm{Proj}_{\mathcal{P}}(\theta_n - t\hat{G}_n) - \theta_n\|_{\mathcal{P}} + \|\mathrm{Proj}_{\mathcal{P}}(\theta_n - t\nabla J_n) - \mathrm{Proj}_{\mathcal{P}}(\theta_n - t\hat{G}_n)\|_{\mathcal{P}} \right)^2 \\
&\quad \leq \left(\frac{t}{\tau_n} \|\theta_{n+1} - \theta_n\|_{\mathcal{P}} + t\|\hat{G}_n - \nabla J_n\|_{\mathcal{P}} \right)^2 \leq \frac{2t^2}{\tau_n^2} \|\theta_{n+1} - \theta_n\|_{\mathcal{P}}^2 + 2t^2 \|\hat{G}_n - \nabla J_n\|_{\mathcal{P}}^2,
\end{aligned}$$

where we used Young's inequality in the last line. Therefore, it holds

$$\sum_{n=1}^{\infty} \tau_n \mathbb{E}\left[\|\mathrm{Proj}_{\mathcal{P}}(\theta_n - t\nabla J_n) - \theta_n\|_{\mathcal{P}}^2\right] \leq 2t^2 \sum_{n=1}^{\infty} \frac{1}{\tau_n} \mathbb{E}\left[\|\theta_{n+1} - \theta_n\|_{\mathcal{P}}^2\right] + 2t^2 \sum_{n=1}^{\infty} \tau_n \mathbb{E}\left[\|\hat{G}_n - \nabla J_n\|_{\mathcal{P}}^2\right].$$

(4.12) now follows from (4.13) and Theorem 4.9.

For the second part, observe that convergence of $(\theta_n)_{n\in\mathbb{N}}$ and density of $(x_n)_{n\in\mathbb{N}}$ in $\mathcal{X}$ yield

$$Z_n(x) \to 0 \quad \text{for all } x \in \mathcal{X}.$$

Therefore, by similar steps as performed in the proof of Theorem 4.7, it holds

$$\|\hat{G}_n - \nabla J_n\|_{\mathcal{P}} \to 0,$$

where ∇J_n denotes $\nabla J(\theta_n)$. Hence, for all $t \geq 0$ we obtain

$$\begin{aligned}
&\|\mathrm{Proj}_{\mathcal{P}}(\theta^* - t\nabla J(\theta^*)) - \theta^*\|_{\mathcal{P}}^2 \\
&\quad = \lim_{n\to\infty} \|\mathrm{Proj}_{\mathcal{P}}(\theta_n - t\nabla J(\theta_n)) - \theta_n\|_{\mathcal{P}} \\
&\quad \leq \lim_{n\to\infty} \left(\|\mathrm{Proj}_{\mathcal{P}}(\theta_n - t\hat{G}_n) - \theta_n\|_{\mathcal{P}} + \|\mathrm{Proj}_{\mathcal{P}}(\theta_n - t\hat{G}_n) - \mathrm{Proj}_{\mathcal{P}}(\theta_n - t\nabla J_n)\|_{\mathcal{P}} \right) \\
&\quad \leq \lim_{n\to\infty} \left(\frac{t}{\tau_n} \|\theta_{n+1} - \theta_n\|_{\mathcal{P}} + \|\mathrm{Proj}_{\mathcal{P}}(\theta_n - t\hat{G}_n) - \mathrm{Proj}_{\mathcal{P}}(\theta_n - t\nabla J_n)\|_{\mathcal{P}} \right) \\
&\quad \leq \lim_{n\to\infty} \frac{t}{\tau} \|\theta_{n+1} - \theta_n\|_{\mathcal{P}} + \lim_{n\to\infty} t\|\hat{G}_n - \nabla J_n\|_{\mathcal{P}} = 0,
\end{aligned}$$

where we used Lemma 4.10 (c) for the second inequality. ∎

5. Numerical Results. In this section, we consider three different settings in which we compare the CSG methods to suiting algorithms from the literature. The comparison is based on the number of gradient evaluations, since these represent the time-consuming computations in complex optimization tasks.

5.1. Comparison with SG. To start our numerical analysis, we consider the problem

$$\min_{\theta\in\mathcal{P}} \quad \frac{1}{2}\int_{\mathcal{X}} (x-\theta)^2 \mathrm{d}x,$$

where $\mathcal{P} = \mathcal{X} = [-\frac{1}{2}, \frac{1}{2}]$.

To study the behavior of the algorithms, we choose four different stepsizes (n^{-1}, $n^{-2/3}$, $n^{-1/3}$ and a constant stepsize of 1) and track the absolute error in each iteration $|\theta_n - \theta^*|$. In order to obtain meaningful results, the 10000 starting points were chosen randomly in $\mathcal{P}$. For a comparison, we do the same for the ordinary stochastic gradient descent method (SG), since it is one of the most commonly used techniques for problems like our example.

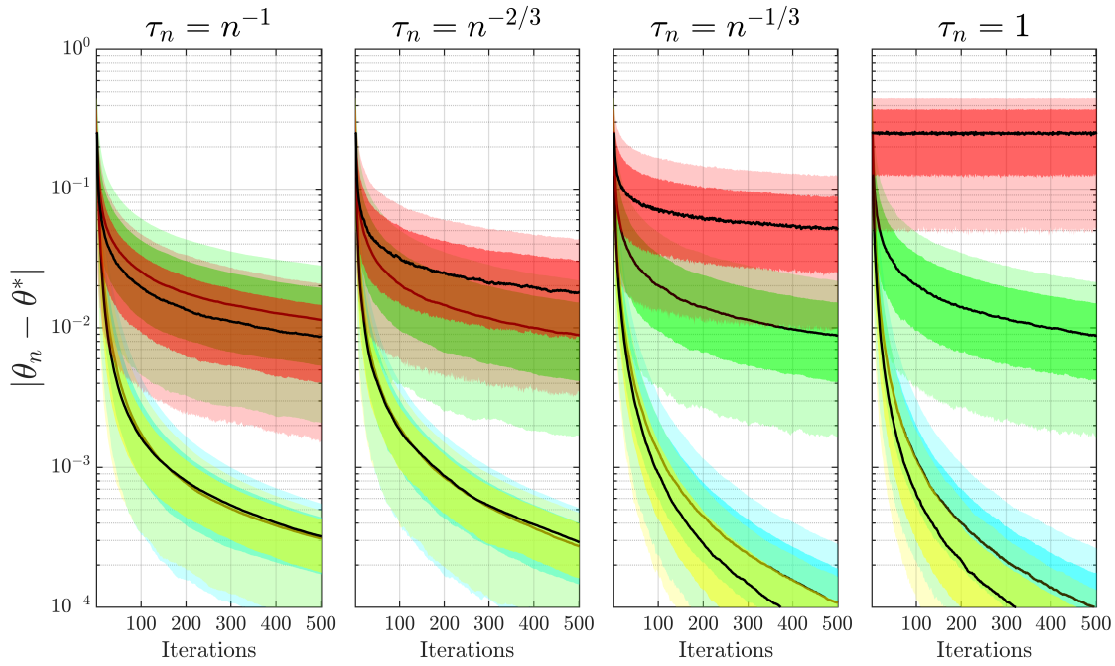

Figure 2. *Comparison of the absolute error $|\theta_n - \theta^*|$ for SG (red), CSG with empirical weights (green), exact hybrid CSG (cyan) and exact CSG (yellow).*

Notice that, in contrast to SG, a larger stepsize does not worsen the performance of the CSG algorithms for our example. Instead, a constant stepsize leads to a faster convergence for the hybrid and exact CSG method, whereas SG fails to solve the problem.

5.2. Comparison with SCGD. As mentioned in Remark 2.3, the vanishing error in inner function value approximations allows us to solve optimization problems in which the cost

function depends non-linearly on a suiting expectation value. For instance, we may solve the problem

$$\min_{\theta\in\mathcal{P}} \quad \frac{1}{20}\int_{\mathcal{Y}}\left(2y+5\int_{\mathcal{X}}\cos\Big(\frac{\theta-x}{\pi}\Big)\mathrm{d}x\right)^2\mathrm{d}y, \tag{5.1}$$

where $\mathcal{P}=[0,10]$, $\mathcal{X}=[-1,1]$ and $\mathcal{Y}=[-3,3]$. The optimal solution $\theta^*=\frac{\pi^2}{2}$ to this example can be found analytically. Setting

$$f_y(t) := \frac{3}{10}(2y+t)^2 \quad \text{and} \quad g_x(t) := 10\cos\Big(\frac{\theta-x}{\pi}\Big),$$

problem (5.1) can be reformulated as

$$\min_{\theta\in\mathcal{P}} \quad \mathbb{E}_{\mathcal{Y}}\big[f_y(\mathbb{E}_{\mathcal{X}}[g_x(\theta)])\big]. \tag{5.2}$$

Since f_y is non-linear, the SG algorithm can not be used to solve (5.1). Therefore, we compare our results with the so called stochastic compositional gradient descent (SCGD) method (see [21]), which is specifically designed for problems of the form (5.2).

Again, the 1000 starting points are randomly generated. This time however, we draw the starting points only from the interval $[\frac{11}{2},\frac{19}{2}]$ instead of $\mathcal{P}=[0,10]$. The reason for this is that the optimal solution $\frac{\pi^2}{2}\approx 4.935$ would otherwise be very close to the median starting point, resulting in artificially small absolute errors for all methods. Since the objective function in (5.1) is strongly convex in a neighborhood of the optimal solution, the accelerated SCGD method (see [21]) performed better than the standard version. Therefore, we compared our results to the aSCGD algorithm and chose the optimal stepsizes for aSCGD according to Theorem 7 in [21]. For the hybrid, inexact hybrid and empirical CSG algorithm, we chose a constant stepsize of $\frac{1}{30}$, which is a rough approximation to the inverse of the Lipschitz constant $\mathsf{L}_{\nabla J}$. The resulting graphs are shown in Figure 3.

From a practical viewpoint, one is mainly interested in how many iterations it takes the error to fall below a desired tolerance. For this purpose, we analyzed the number of steps after which the different methods achieved a given absolute error with 90% certainty. The results can be seen in Figure 4.

5.3. Chance constraint problems. As a prototype example for chance constraint problems, we consider

$$\begin{aligned}\max_{\theta\in[0,\frac{3}{4}]} \quad & \theta \\ \text{s.t.} \quad & \mathbb{P}(\theta - X^2 \le 0) \ge \frac{1}{2}, \quad X\sim\mathcal{U}_{[-1,1]}\end{aligned}$$

with optimal solution $\theta^*=\frac{1}{4}$. By introducing the characteristic function $\chi_{[0,\infty)}$ and transforming the constraint to a penalty term, we arrive at

$$\max_{\theta\in[0,\frac{3}{4}]} \quad \theta-\lambda\max\left\{0,\frac{1}{2}\int_{-1}^{1}\chi_{[0,\infty)}(\theta-x^2)\mathrm{d}x-\frac{1}{2}\right\}.$$

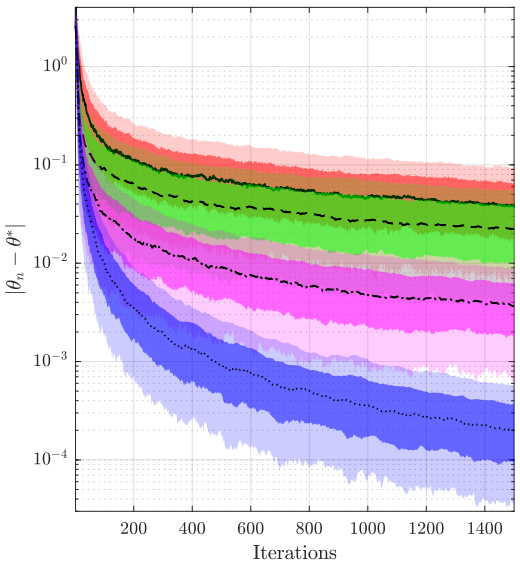

Figure 3. *Comparison of the absolute error $|\theta_n - \theta^*|$. From top to bottom: aSCGD (red/solid), CSG with empirical weights (green/dashed), inexact hybrid CSG with $\beta = 1.5$ (magenta/dash-dotted) and hybrid CSG (blue/dotted). The shaded areas indicate the quantiles $P_{0.1,0.9}$ (light) and $P_{0.25,0.75}$ (dark).*

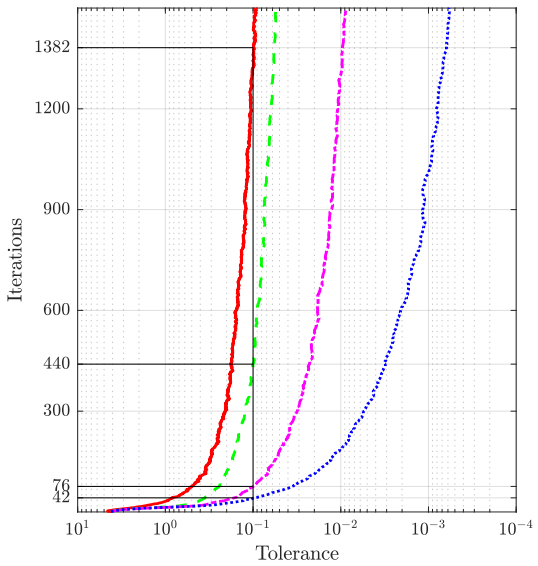

Figure 4. *Minimum number of steps needed for aSCGD (red/solid), CSG with empirical weights (green/dashed), inexact hybrid CSG with $\beta = 1.5$ (magenta/dash-dotted) and hybrid CSG (blue/dotted) such that at least 90% of the runs achieved an absolute error smaller than the given tolerance.*

Since the penalized objective function is no longer continuously differentiable, we can not guarantee the existence of a gradient and will have to work with subgradients instead, cf. [2]. Notice that the proofs provided above also hold true for a subgradient method, if the stepsize is chosen accordingly. While the computation of a (sub-)gradient of $\max\{0, \cdot\}$ is not an issue, $\chi_{[0,\infty)}$ needs to be regularized further. The final problem then reads as follows:

$$\max_{\theta \in [0, \frac{3}{4}]} \quad \theta - \lambda \max\left\{0, \frac{1}{4}\int_{-1}^{1} \left((\tanh\left(\alpha(\theta - x^2)\right) + 1\right) \mathrm{d}x - \frac{1}{2}\right\}. \tag{5.3}$$

Due to the non-linearity of $\max\{0, \cdot\}$, we again choose the SCGD method for comparison. This time, the objective function is not strongly convex in a neighborhood of θ_{opt}. Therefore, the stepsizes for the standard SCGD method are chosen according to Theorem 6 in [21], i.e., optimal for this setting. For the CSG algorithms, we choose $\tau_n = \frac{1}{n}$. Lastly, we fix $\lambda = 3$ and $\alpha = 25$. The optimal solution θ_{opt} to (5.3) then satisfies $|\theta^* - \theta_{opt}| < 1.5 \cdot 10^{-3}$. The results of 1000 runs with random starting points in $[0, \frac{3}{4}]$ are presented in Figure 5.

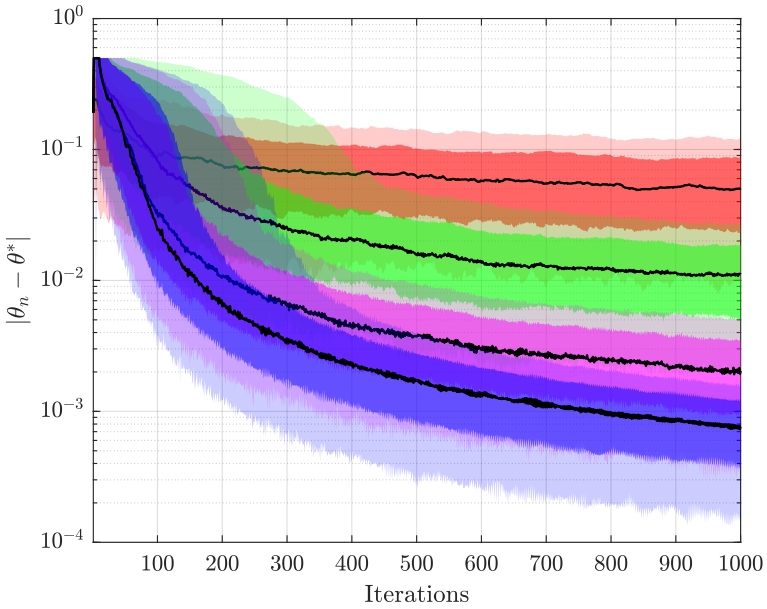

Figure 5. *Comparison of the absolute error $|\theta_n - \theta_{opt}|$. From top to bottom: SCGD (red), CSG with empirical weights (green), inexact hybrid CSG with $\beta = 1.5$ (magenta) and hybrid CSG (blue). The shaded areas indicate the quantiles $P_{0.1,0.9}$ (light) and $P_{0.25,0.75}$ (dark).*

6. Conclusion and Outlook.

In this article a more flexible way to compute design dependent integration weights for the efficient approximation of the full cost function and its gradient when applying the CSG method to a class of stochastic optimization problems was introduced. While this significantly widened the scope of the CSG method, there are still a number of research questions, which would be very interesting to be investigated in the future. First, as a consequence of the strong convergence properties shown in this paper, the CSG method – in the course of the optimization iterations – behaves more and more like a fully deterministic descent method. This calls for more elaborate techniques to calculate the step length such as linesearch or trust region strategies. Another interesting question is, if convergence of the iterates generated by the CSG method can be shown for a constant choice of the step size. Indeed the numerical examples we have presented in this paper suggest that this should be possible. And finally, exploiting specific structures of the given probability distributions, one could come up with even more efficient integration techniques allowing to solve problems with high dimensional distributions more efficiently than using the empirical weight strategy presented in this article.

Acknowledgements. Funded by the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation) through project D05 in the CRC 1411 (Project-ID 416229255) and subproject B06 in TRR 154 (Project-ID 239904186).

REFERENCES

[1] R. A. Adams and J. J. F. Fournier, *Sobolev spaces*, vol. 140 of Pure and Applied Mathematics (Amsterdam), Elsevier/Academic Press, Amsterdam, second ed., 2003.

[2] Y. I. Alber, A. N. Iusem, and M. V. Solodov, *On the projected subgradient method for nonsmooth convex optimization in a Hilbert space*, Math. Programming, 81 (1998), pp. 23–35.

[3] J.-P. Aubin, *Applied functional analysis*, Pure and Applied Mathematics (New York), Wiley-Interscience, New York, second ed., 2000. With exercises by Bernard Cornet and Jean-Michel Lasry, Translated from the French by Carole Labrousse.

[4] K. Beyer, J. Goldstein, R. Ramakrishnan, and U. Shaft, *When is "nearest neighbor" meaningful?*, ICDT 1999. LNCS, 1540 (1997).

[5] P. Billingsley, *Convergence of probability measures*, Wiley Series in Probability and Statistics: Probability and Statistics, John Wiley & Sons, Inc., New York, second ed., 1999. A Wiley-Interscience Publication.

[6] L. Bottou, F. E. Curtis, and J. Nocedal, *Optimization methods for large-scale machine learning*, SIAM Rev., 60 (2018), pp. 223–311.

[7] F. E. Curtis, K. Scheinberg, and R. Shi, *A stochastic trust region algorithm based on careful step normalization*, INFORMS J. Optim., 1 (2019), pp. 200–220.

[8] G. B. Folland, *A guide to advanced real analysis*, vol. 37 of The Dolciani Mathematical Expositions, Mathematical Association of America, Washington, DC, 2009. MAA Guides, 2.

[9] N. Fournier and A. Guillin, *On the rate of convergence in wasserstein distance of the empirical measure*, Probability Theory and Related Fields, 162 (2015), pp. 707–738.

[10] A. L. Gibbs and F. E. Su, *On choosing and bounding probability metrics*, International statistical review, 70 (2002), pp. 419–435.

[11] A. Klenke, *Probability theory*, Universitext, Springer-Verlag London, Ltd., London, 2008. A comprehensive course, Translated from the 2006 German original.

[12] Q. Lin, S. Nadarajah, N. Soheili, and T. Yang, *A data efficient and feasible level set method for stochastic convex optimization with expectation constraints*, J. Mach. Learn. Res., 21 (2020), pp. Paper No. 143, 45.

[13] L. M. Nguyen, K. Scheinberg, and M. Takávc, *Inexact SARAH algorithm for stochastic optimization*, Optim. Methods Softw., 36 (2021), pp. 237–258.

[14] C. Paquette and K. Scheinberg, *A stochastic line search method with expected complexity analysis*, SIAM J. Optim., 30 (2020), pp. 349–376.

[15] V. Patel, *Kalman-based stochastic gradient method with stop condition and insensitivity to conditioning*, SIAM J. Optim., 26 (2016), pp. 2620–2648.

[16] L. Pflug, N. Bernhardt, M. Grieshammer, and M. Stingl, *CSG: a new stochastic gradient method for the efficient solution of structural optimization problems with infinitely many states*, Struct. Multidiscip. Optim., 61 (2020), pp. 2595–2611.

[17] H. Robbins and S. Monro, *A stochastic approximation method*, Ann. Math. Statistics, 22 (1951), pp. 400–407.

[18] M. Schmidt, N. Le Roux, and F. Bach, *Minimizing finite sums with the stochastic average gradient*, Math. Program., 162 (2017), pp. 83–112.

[19] C. Tan, S. Ma, Y.-H. Dai, and Y. Qian, *Barzilai-borwein step size for stochastic gradient descent*, 2016, https://arxiv.org/abs/1605.04131.

[20] V. S. Varadarajan, *On the convergence of sample probability distributions*, Sankhyā, 19 (1958), pp. 23–26.

[21] M. Wang, E. X. Fang, and H. Liu, *Stochastic compositional gradient descent: algorithms for minimizing compositions of expected-value functions*, Math. Program., 161 (2017), pp. 419–449.

[22] X. Wang and H. Zhang, *Inexact proximal stochastic second-order methods for nonconvex composite optimization*, Optim. Methods Softw., 35 (2020), pp. 808–835.

[23] S. Yakowitz, J. E. Krimmel, and F. Szidarovszky, *Weighted Monte Carlo integration*, SIAM J. Numer. Anal., 15 (1978), pp. 1289–1300.

[24] Y.-N. Zhu and X. Zhang, *Stochastic primal dual fixed point method for composite optimization*, J. Sci. Comput., 84 (2020), pp. Paper No. 16, 25.