A dimensionality reduction technique for unconstrained global optimization of functions with low effective dimensionality*

Coralia Cartis ‡,§   Adilet Otemissov ‡,§

16th January 2020

Abstract

We investigate the unconstrained global optimization of functions with low effective dimensionality, that are constant along certain (unknown) linear subspaces. Extending the technique of random subspace embeddings in [Wang et al., Bayesian optimization in a billion dimensions via random embeddings. *JAIR*, 55(1): 361–387, 2016], we study a generic Random Embeddings for Global Optimization (REGO) framework that is compatible with any global minimization algorithm. Instead of the original, potentially large-scale optimization problem, within REGO, a Gaussian random, low-dimensional problem with bound constraints is formulated and solved in a reduced space. We provide novel probabilistic bounds for the success of REGO in solving the original, low effective-dimensionality problem, which show its independence of the (potentially large) ambient dimension and its precise dependence on the dimensions of the effective and randomly embedding subspaces. These results significantly improve existing theoretical analyses by providing the exact distribution of a reduced minimizer and its Euclidean norm and by the general assumptions required on the problem. We validate our theoretical findings by extensive numerical testing of REGO with three types of global optimization solvers, illustrating the improved scalability of REGO compared to the full-dimensional application of the respective solvers.

Keywords: global optimization, random matrix theory, dimensionality reduction techniques, functions with low effective dimensionality

1 Introduction

In this paper, we address the unconstrained global optimization problem

$$\min_{\boldsymbol{x}\in\mathbb{R}^D} \ f(\boldsymbol{x}), \tag{P}$$

where $f : \mathbb{R}^D \to \mathbb{R}$ is a real-valued continuous, possibly non-convex, deterministic, function defined on the whole $\mathbb{R}^D$. We assume that there exists $\boldsymbol{x}^* \in \mathbb{R}^D$ such that $\min_{\boldsymbol{x}\in\mathbb{R}^D} f(\boldsymbol{x}) = f(\boldsymbol{x}^*) = f^*$. This implies that f is bounded below, namely, $f^* > -\infty$, and that the minimum in (P) is attained (not all minimizers are at infinity).

To alleviate the curse of dimensionality, we further restrict ourselves to a particular class of functions whose true (intrinsic) dimension is much less than the ambient problem dimension. These functions are constant along certain linear subspaces, which may not necessarily
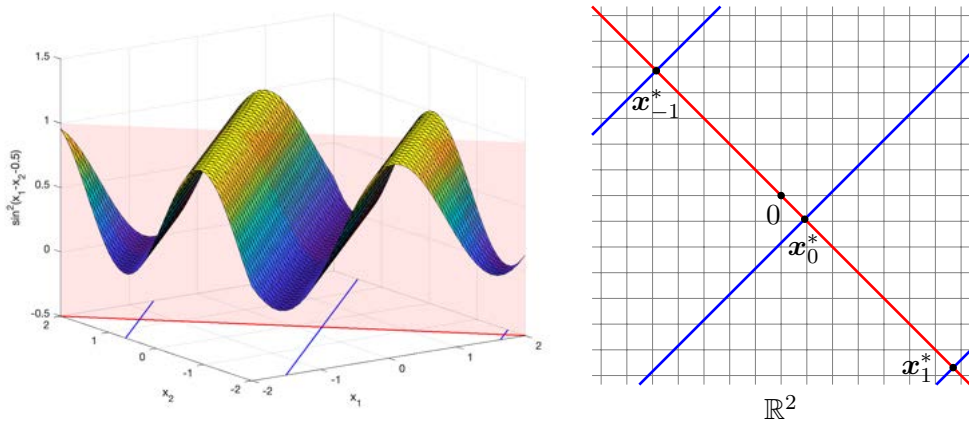
1

Figure 1: The function in (1.1) and its domain are plotted on the left- and right-side, respectively. The red line is the effective subspace $\boldsymbol{x} = (1 \quad -1)y$ and it intersects the blue lines of global minimizers at $\boldsymbol{x}_k^*$ defined in Example 1.1 for $k = -1, 0, 1$; these points also correspond to optimal solutions in the reduced space.

be aligned with the standard axes. In literature, these functions are known under different names: functions with '*low effective dimensionality*' [43], functions with '*active subspaces*' [8] and '*multi-ridge*' functions [17, 41]. They have been found in a number of applications mainly related to parameter studies. In hyper-parameter optimization for neural networks [2] and heuristic algorithms for combinatorial optimization problems [23], studies have shown that the respective objective functions are affected by only a few hyper-parameters while the many other input hyper-parameters are redundant. Similarly, in complex engineering and physical simulation problems [8], such as in climate modelling [26], systems are modelled by several input parameters with only a small number of the parameters or a combination of them having a true effect on the system's behaviour. To clarify this concept, we give a simple example of a function with lower effective dimensionality.

Example 1.1. Consider the following optimization problem:

$$\min_{\boldsymbol{x}\in\mathbb{R}^2} \ f(\boldsymbol{x}) = \sin^2(x_1 - x_2 - 0.5). \tag{1.1}$$

By solving $f(\boldsymbol{x}) = 0$, we find that the set of global minimizers is given by $\{(1 \ \ 1)^T t - (0 \ \ 0.5 + \pi k)^T : t \in \mathbb{R}, k \in \mathbb{Z}\}$. For each fixed value of k, the set corresponds to a distinct line of global minimizers along which the function is constant (see Figure 1). The effective subspace[1] of f is $(x_1 \ x_2) = (1 \ \ -1)^T y$ for $y \in \mathbb{R}$. We substitute this in (1.1) to obtain the reduced/lower-dimensional optimization problem $\min_{y\in\mathbb{R}} \sin^2(2y - 0.5)$, which has the same global minimum as (1.1), with global minimizers $y_k^* = \pi k/2 + 0.25$, $k \in \mathbb{Z}$. We recover the corresponding solutions to (1.1) by setting $\boldsymbol{x}_k^* = (1 \ \ -1)^T y_k^*$ for $k \in \mathbb{Z}$.

As Example 1.1 illustrates, it is possible to cast (P) into a lower-dimensional problem which has the same global minimum f^*. This is straightforward when the effective subspace is known, but far less so in applications where f is potentially black-box. When the effective subspace is unknown, it was proposed (in the context of Bayesian Optimization) in Wang et al. [43] to use random embeddings. The proposed technique solves the following lower-dimensional

---

[1]The effective subspace can be determined by considering the orthogonal complement of the constant subspace (along which f does not vary), in this example, spanned by the vector $(1 \ 1)^T$.

optimization problem instead of directly tackling (P):

$$\min_{\boldsymbol{y}} \ f(\boldsymbol{A}\boldsymbol{y})$$
$$\text{subject to} \ \ \boldsymbol{y} \in \mathcal{Y} = [-\delta, \delta]^d, \tag{RP}$$

where $\boldsymbol{A}$ is a $D \times d$ Gaussian random matrix (see Definition A.1) and $\mathcal{Y} = [-\delta, \delta]^d$ for some carefully chosen $\delta > 0$ and $d \ll D$. Note that, unlike (P), (RP) has (box) constraints, which are typically imposed to make the approach practical (i.e. to avoid unrealistic searches over infinite domains).

Definition 1.2. We say that (RP) is *successful* if there exists $\boldsymbol{y}^* \in \mathcal{Y}$ such that $f(\boldsymbol{A}\boldsymbol{y}^*) = f^*$.

Related work. The scalability challenges of Bayesian Optimization (BO) algorithms for generic black-box functions have prompted research into improving efficiency of this class of methods for functions with special structure. Different structural assumptions on the objective have been analysed for BO, such as additivity or (partial) separability, which assumes that the objective function can be represented as the sum of smaller-dimensional functions with non-overlapping variables [44, 25, 28] or with overlapping ones [35].

Another popular structural assumption is the above-mentioned low-effective dimensionality of the objective. In its simplest form, this considers the effective subspace to be aligned with the coordinate axes, which is equivalent to the presence of redundant variables [7, 1]. More generally, the optimization of functions that are constant along *arbitrary* linear subspaces – which, as mentioned above, is also the focus of this paper – has been addressed using BO methods in [11, 43, 19, 13], and extended to other problem and algorithm classes such as derivative-free optimization [34], multi-objective optimization [33] and evolutionary methods [38]. Some proposals learn the effective subspace of the function beforehand (using for example, a low rank matrix recovery approach) [41, 17] and then optimize in the reduced subspace [11, 13]. Alternating learning and optimization steps has also been proposed [19], as well as bypassing learning and directly optimizing in randomly-chosen low-dimensional subspaces (provided an estimate of the effective dimension is known) [43, 4, 5].

For the latter, Wang et al. [43] developed the so-called REMBO algorithm, which is a BO framework for problem (P) with box constraints $x \in \mathcal{X}$ that uses Gaussian random embeddings (namely, $\boldsymbol{A}$ is a Gaussian random matrix) to generate the reduced problem (RP). They find that the size of $\mathcal{Y}$ is the primary factor in determining the success (or failure) of the reduced problem, and quantify the probability of success of (RP) for the case when the embedded dimension d is equal to the effective one and the effective subspace is aligned with the coordinate axes (see [43, Theorem 3]). A challenge of (RP) for BO with box constraints is that, even when (RP) is successful, the high-dimensional image $\boldsymbol{A}\boldsymbol{y} \in \mathbb{R}^D$ of a point $\boldsymbol{y} \in \mathcal{Y}$ may be outside the feasible set $\mathcal{X}$. For this reason, REMBO is equipped with a map $p_{\mathcal{X}} : \mathbb{R}^D \to \mathbb{R}^D$ that projects the image of the reduced solutions that fall outside $\mathcal{X}$ to the closest point on the boundary of $\mathcal{X}$. To model a Gaussian Process for the reduced problem, [43] proposes two kernels: a high-dimensional $k_{\mathcal{X}}$ and a low-dimensional $k_{\mathcal{Y}}$. Kernel $k_{\mathcal{X}}$ suffers from high-dimensionality as it constructs a GP in a D-dimensional space. The benefit of $k_{\mathcal{Y}}$ is that it constructs a GP in a d-dimensional subspace, but this kernel over-explores regions in $\mathcal{Y}$ whose high-dimensional images outside $\mathcal{X}$ are mapped to the same points in $\mathcal{X}$ through a non-injective $p_{\mathcal{X}}$. To remedy these issues, Binois et al. [4] propose a new kernel k_{Ψ} which has the benefit of being low-dimensional while avoiding the over-exploratory tendency of $k_{\mathcal{Y}}$. In [5], Binois et al. also propose a new mapping γ (instead of $p_{\mathcal{X}}$) and define $\mathcal{Y}$ and new kernels based on this new mapping.

Sanyang and Kabán [38] develop REMEDA, which uses random embeddings within an Evolutionary algorithm EDA. Their theoretical results on quantifying the size of $\mathcal{Y}$/the success of

3

(RP) improve on those in [43] and are applicable for certain choices of d, greater than the effective subspace dimension; they also experiment with estimating the effective dimension numerically.

Qian et al. [34] extend the framework and some of the results in Wang et al. [43] to functions with approximate low effective subspaces, proposing the use of multiple random embeddings within any derivative-free solver. They contrast the use of a single versus multiple embeddings on three test problems of varying dimensions and using three different types of derivative-free solvers (evolutionary, Bayesian and model-based).

Recently, in the context of Bayesian optimization, Nayebi et al. [29] use a different random ensemble based on hashing matrices to represent the embedded subspaces and define $\mathcal{Y}$ as $[-1, 1]^d$; this formulation guarantees that the high-dimensional points are always inside $\mathcal{X}$ and, thus, their method avoids the feasibility corrections of REMBO.

Our contributions. We investigate a general random embeddings framework for unconstrained global optimization of functions with low effective dimensionality, where we allow the effective subspace of the objective function and its dimension (denoted by d_e) to be arbitrary (not necessarily aligned with coordinate axes and not limited in dimension by problem constants)[2]. This framework also allows the use of any global solver to solve the reduced problem.

We significantly extend and improve the theoretical analyses in [43, 38], providing an in-depth investigation of the reduced problem (RP) when Gaussian random embeddings are used. In particular, while [43, 38] estimate the Euclidean norm of a (random) reduced minimizer, we derive its exact distribution, using tools from random matrix theory. We show that this reduced minimizer, when appropriately scaled, follows the inverse chi-squared distribution with $d - d_e + 1$ degrees of freedom, where d is the dimension of the random embedding (Theorem 3.7). Moreover, we derive the probability density function of this reduced minimizer (Theorem 3.10) by first proving that it follows a spherical distribution. These results imply that, under certain assumptions, solving (RP) has no dependence on the ambient dimension D. Subsequently, Theorem 4.1 and Corollary 4.2 estimate the probability that (RP) is successful. The latter result extends both [43, Theorem 3] and [38, Theorem 2] to arbitrary effective subspaces and any $d \geq d_e$, and establishes a notable and more precise trade-off between the success of (RP), δ (the size of the reduced domain $\mathcal{Y}$) and the embedding dimension d; thus allowing us to choose appropriate values for these parameters in the algorithm. Furthermore, we describe how to extend the main results to affine random embeddings (which draw random subspaces at any chosen (reference) point in $\mathbb{R}^D$), which indicate that the probability of success of (RP) is higher if the point of reference is closer to the set of global minimizers.

Similarly to the algorithmic frameworks proposed in [43, 34], we propose REGO (Random Embeddings for Global Optimization) that solves a single randomly-embedded reduced problem (RP) instead of (P) and is compatible with any generic global optimization solver[3]. We use and validate our theoretical results by providing extensive numerical testing of REGO with three types of solvers for (RP): DIRECT (Lipschitz-optimization), BARON (branch and bound), and KNITRO (multi-start local optimization). We use 19 standard global optimization test problems to generate functions with effective dimensionality structure and of growing ambient dimension D. When comparing REGO with the direct optimization of the ensuing problems without embeddings, we find that REGO's performance is essentially independent of D for all three solvers and that it is successful in recovering the original global minimum in most cases

---

[2]Note that, as problem (P) has no (bound) constraints, we do not need to use the projection operator $p_{\mathcal{X}}$ in [43, 4, 5]. Of course, this comes at the cost of our approach being unable to guarantee feasibility for the original problem if (P) does have constraints.

[3]The REGO framework is defined for the unconstrained (P) but can also be helpful for constrained problems (for example, $\boldsymbol{x} \in \mathcal{X}$), where the constraints are imposed just to avoid searches over an infinite domain and where minimizers outside the feasible domain are acceptable.

with only one embedding[4]. We also test the robustness of REGO's performance to variations in algorithm parameters such as δ and d.

Paper outline. In Section 2, we formally define and describe functions with low effective dimensionality emphasizing their geometrical aspects. In Section 3, we characterize the reduced minimizers in the reduced space focusing on the minimal 2-norm minimizer. For this minimizer, we derive the distribution of its Euclidean norm and its probability density function. We use the former result in Section 4 to derive a probabilistic bound for the success of (RP). In Section 5, we conduct numerical experiments to test REGO algorithm on functions with low effective dimensionality using three optimization solvers, namely, DIRECT, BARON and KNITRO, while in Section 6 we draw our conclusions and future directions.

Notation. We use bold capital letters to denote matrices ($\boldsymbol{A}$) and bold lowercase letters ($\boldsymbol{a}$) to denote vectors. In particular, we use $\boldsymbol{I}_D$ to denote the $D \times D$ identity matrix and $\boldsymbol{0}_D$, $\boldsymbol{1}_D$ (or simply $\boldsymbol{0}$, $\boldsymbol{1}$) to denote the D-dimensional vectors of zeros and ones, respectively. For an $D \times d$ matrix $\boldsymbol{A}$, we write range($\boldsymbol{A}$) to denote the linear subspace spanned by the columns of $\boldsymbol{A}$ in $\mathbb{R}^D$.

We let $\langle \cdot, \cdot \rangle$, $\| \cdot \|$ and $\| \cdot \|_\infty$ to denote the usual Euclidean inner product, the Euclidean norm and the infinity norm, respectively. Where emphasis is needed, for the Euclidean norm we also use $\| \cdot \|_2$.

Given two random variables (vectors) x and y ($\boldsymbol{x}$ and $\boldsymbol{y}$), we write $x \stackrel{law}{=} y$ ($\boldsymbol{x} \stackrel{law}{=} \boldsymbol{y}$) to denote the fact that x and y ($\boldsymbol{x}$ and $\boldsymbol{y}$) have the same distribution. We reserve the letter $\boldsymbol{A}$ to refer to a $D \times d$ Gaussian random matrix (see Definition A.1) and write χ_n^2 to denote a chi-squared random variable with n degrees of freedom (see Appendix A.2).

2 Functions with low effective dimensionality

In this section, we formally define functions with low effective dimensionality and describe the geometry of (RP).

2.1 Definitions and assumptions

Functions with low effective dimensionality can be defined in at least two ways [17, 43]. We will work with a definition given in terms of linear subspaces, provided in [43].

Definition 2.1 (Functions with low effective dimensionality). A function $f : \mathbb{R}^D \to \mathbb{R}$ has effective dimensionality $d_e \leq D$ if there exists a linear subspace $\mathcal{T}$ of dimension d_e such that for all vectors $\boldsymbol{x}_\top$ in $\mathcal{T}$ and $\boldsymbol{x}_\perp$ in $\mathcal{T}^\perp$ (orthogonal complement of $\mathcal{T}$) we have

$$f(\boldsymbol{x}_\top + \boldsymbol{x}_\perp) = f(\boldsymbol{x}_\top). \tag{2.1}$$

and d_e is the smallest integer satisfying (2.1).

The linear subspace $\mathcal{T}$ is called the *effective* subspace of f and its orthogonal complement $\mathcal{T}^\perp$, the *constant* subspace of f. It is convenient to think of $\mathcal{T}^\perp$ as a subspace of no variation of largest dimension (along which the value of f does not change) and $\mathcal{T}$ as its orthogonal complement.

Every vector $\boldsymbol{x}$ can be decomposed as $\boldsymbol{x} = \boldsymbol{x}_\top + \boldsymbol{x}_\perp$, where $\boldsymbol{x}_\top$ and $\boldsymbol{x}_\perp$ are orthogonal projections of $\boldsymbol{x}$ onto $\mathcal{T}$ and $\mathcal{T}^\perp$, respectively. In particular, if $\boldsymbol{x}^*$ is a global minimizer and f^* is the global minimum of f in $\mathcal{X}$ then $\boldsymbol{x}^* = \boldsymbol{x}_\top^* + \boldsymbol{x}_\perp^*$ and

$$f^* = f(\boldsymbol{x}^*) = f(\boldsymbol{x}_\top^* + \boldsymbol{x}_\perp^*) = f(\boldsymbol{x}_\top^*). \tag{2.2}$$

---

[4]These numerical results assume that (an upper bound on) the true effective dimension d_e is known/available.

Moreover, we have

$$f^* = f(\boldsymbol{x}_\top^*) = f(\boldsymbol{x}_\top^* + \boldsymbol{x}_\perp)$$

for every vector $\boldsymbol{x}_\perp$ in $\mathcal{T}^\perp$. It is important to note that there can be multiple points $\boldsymbol{x}_\top^*$ in $\mathbb{R}^D$ satisfying the above definition such as, for instance, $\boldsymbol{x}_{-1}^*$, $\boldsymbol{x}_0^*$ and $\boldsymbol{x}_1^*$ in Example 1.1. By contrast, the function $f = (x_1 - x_2 - 0.5)^2$ admits a unique $\boldsymbol{x}_\top^*$ given by $(0.25 \ -0.25)^T$.

We summarize the above discussion in the following assumption.

Assumption 2.2. *The function $f : \mathbb{R}^D \to \mathbb{R}$ is continuous and has effective dimensionality $d_e \leq d$ with effective subspace[5] $\mathcal{T}$ and constant subspace $\mathcal{T}^\perp$ spanned by the columns of orthonormal matrices $\boldsymbol{U} \in \mathbb{R}^{D \times d_e}$ and $\boldsymbol{V} \in \mathbb{R}^{D \times (D-d_e)}$, respectively.*

Recalling the definition of problem (P) on page 1, let

$$\mathcal{G} = \{\boldsymbol{x} \in \mathbb{R}^D : f(\boldsymbol{x}) = f^*\}$$

be the set of global minimizers in $\mathbb{R}^D$. Under Assumption 2.2, the set $\mathcal{G}$ can be represented as a union of (possibly infinitely many) affine subspaces each containing one particular $\boldsymbol{x}_\top^*$ (see proof of Theorem 4.1). Each of these affine subspaces is a $(D - d_e)$-dimensional set $\{\boldsymbol{x} \in \mathbb{R}^D : \boldsymbol{x} \in \boldsymbol{x}_\top^* + \mathcal{T}^\perp\}$ — a translation of $\mathcal{T}^\perp$ by the vector $\boldsymbol{x}_\top^*$ that the corresponding affine subspace must contain. In particular, if there is a unique $\boldsymbol{x}_\top^*$ in $\mathbb{R}^D$ then $\mathcal{G} = \{\boldsymbol{x} \in \mathbb{R}^D : \boldsymbol{x} \in \boldsymbol{x}_\top^* + \mathcal{T}^\perp\}$. Note also that point(s) $\boldsymbol{x}_\top^*$ lie in $\mathcal{G} \cap \mathcal{T}$, and are the closest minimizers to the origin in Euclidean norm amongst all the minimizers lying in their respective affine subspaces.

Our analysis applies to any minimizer $\boldsymbol{x}^*$ with $\boldsymbol{x}_\top^* \neq \boldsymbol{0}$. If $\boldsymbol{x}_\top^* = \boldsymbol{0}$, then (RP) has a trivial solution. In that case, the origin is a global minimizer implying that every embedding is successful with a solution $\boldsymbol{y}^* = \boldsymbol{0}$. Hence, we focus our analysis for finding a(ny) minimizer $\boldsymbol{x}^* \in \mathcal{G}$ with $\boldsymbol{x}_\top^* \neq \boldsymbol{0}$.

Assumption 2.3. *Given Assumption 2.2, let $\boldsymbol{x}^* \in \mathcal{G}$ such that $\boldsymbol{x}_\top^* = \boldsymbol{U}\boldsymbol{U}^T\boldsymbol{x}^*$ — the unique Euclidean projection of $\boldsymbol{x}^*$ onto $\mathcal{T}$ — is non-zero. Let $\mathcal{G}^* := \boldsymbol{x}_\top^* + \mathcal{T}^\perp$ be the affine subspace of $\mathcal{G}$ that contains $\boldsymbol{x}_\top^*$.*

The set $\mathcal{G}$ contains infinitely many global minimizers — a particularly useful feature of the functions with low effective dimensionality; this fact suggests that targeting $\mathcal{G}$ numerically may potentially be easier than if $\mathcal{G}$ contained only one point.

2.2 Geometric description

We now provide a geometric description of (RP), which serves as a basis for our theoretical investigations.

In Figure 2, we illustrate schematically $\mathcal{T}$ (the effective subspace of f), $\mathcal{T}^\perp$ (the orthogonal component of $\mathcal{T}$), $\mathcal{G}^*$ (a connected component[6] of $\mathcal{G}$), $\boldsymbol{x}_\top^*$ (the orthogonal projection of the global minimizers on $\mathcal{G}^*$ onto $\mathcal{T}$). Since the orientation and position of these geometric objects are solely determined by the (deterministic) objective function, they are fixed, non-random.

By applying the 'random embedding' (RP), we switch from optimizing over $\mathbb{R}^D$ to optimizing over $\mathcal{Y}$. The linear mapping $\boldsymbol{y} \to \boldsymbol{A}\boldsymbol{y}$ maps points of the hypercube $\mathcal{Y}$ to points along the subspace range($\boldsymbol{A}$) in $\mathbb{R}^D$, which means that searching over $\mathcal{Y}$ is equivalent to searching over the corresponding feasible set along range($\boldsymbol{A}$) in $\mathbb{R}^D$. An example of this mapping is illustrated in Figure 2 with two red line segments: the segment (from $-\delta\boldsymbol{1}$ to $\delta\boldsymbol{1}$) representing $\mathcal{Y}$ is being mapped to the right segment, which lies in range($\boldsymbol{A}$). It is important to note that the centre of

---

[5]Note that $\mathcal{T}$ in Assumption 2.2 may not be aligned with the standard axes.

[6]Recall that $\mathcal{G}$ is a union of affine subspaces; $\mathcal{G}^*$ is one of them.
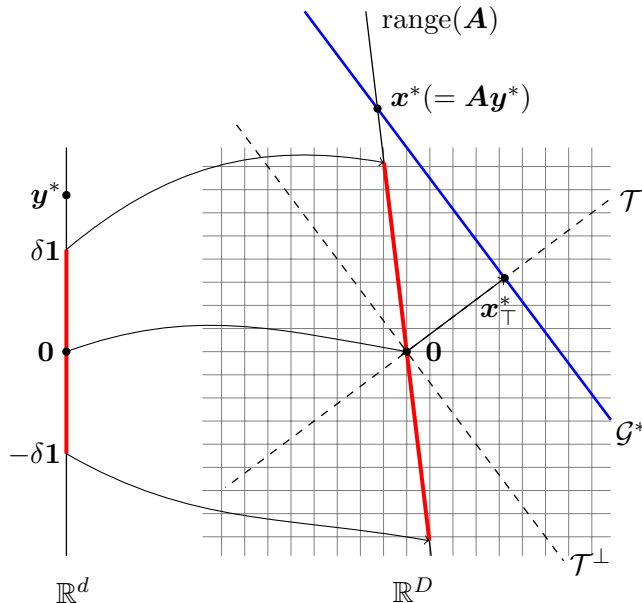
Figure 2: The figure shows an abstract illustration of the embedding of a d-dimensional linear subspace into $\mathbb{R}^D$. The line range($\boldsymbol{A}$) corresponds to the embedded subspace. The red line in $\mathbb{R}^d$ represents the hypercube $\mathcal{Y} = \{\boldsymbol{y} \in \mathbb{R}^d : -\delta\mathbf{1} \leq \boldsymbol{y} \leq \delta\mathbf{1}\}$, which, after application of $\boldsymbol{A}$, maps to the red line along range($\boldsymbol{A}$) in $\mathbb{R}^D$. In this configuration, condition (2.4) is satisfied but (2.3) is not: range($\boldsymbol{A}$) intersects $\mathcal{G}$ at $\boldsymbol{x}^* = \boldsymbol{A}\boldsymbol{y}^*$, but $\boldsymbol{y}^*$ lies outside $\mathcal{Y}$.

$\mathcal{Y}$ maps to the origin in $\mathbb{R}^D$ and, hence, the corresponding search in the original space is also centred at the origin.

The most valuable information that we want to retain while performing dimensionality reduction is the value of f^*. We would like $\min_{\boldsymbol{y} \in \mathcal{Y}} f(\boldsymbol{A}\boldsymbol{y}) = f^*$, which holds only if there is at least one $\boldsymbol{y}^*$ in $\mathcal{Y}$ such that $f(\boldsymbol{A}\boldsymbol{y}^*) = f^*$. This condition has a geometric interpretation and, following from the definition of $\mathcal{G}$, it can equivalently be stated as:

$$\text{there exists a } \boldsymbol{y}^* \in \mathcal{Y} \text{ such that } \boldsymbol{A}\boldsymbol{y}^* \in \mathcal{G}. \tag{2.3}$$

For (2.3) to hold, we must first ensure that

$$\text{there exists a } \boldsymbol{y}^* \text{ in } \mathbb{R}^d \text{ such that } \boldsymbol{A}\boldsymbol{y}^* \in \mathcal{G}. \tag{2.4}$$

In this regard, Wang et al. [43] proved the following theorem.

Theorem 2.4. *(see [43, Theorem 2]) Let Assumption 2.2 hold and let $\boldsymbol{A}$ be a $D \times d$ Gaussian matrix with $d \geq d_e$. Then, with probability one, for any fixed $\boldsymbol{x} \in \mathbb{R}^D$, there exists a $\boldsymbol{y} \in \mathbb{R}^d$ such that $f(\boldsymbol{x}) = f(\boldsymbol{A}\boldsymbol{y})$. In particular, for a global minimizer $\boldsymbol{x}^*$, with probability one, there exists a $\boldsymbol{y}^* \in \mathbb{R}^d$ such that $f(\boldsymbol{A}\boldsymbol{y}^*) = f(\boldsymbol{x}^*) = f^*$.*

While satisfaction of (2.4) only depends on $d \geq d_e$, that of (2.3) is determined by the values of both d and δ. For larger values of d and/or δ the probability that (2.3) is satisfied is higher. One must, on the other hand, be cognisant of the fact that larger values of d — the dimension of (RP) — and/or δ — the half-length of the domain — demand more computational resources. Therefore, a careful calibration of these two parameters is needed to ensure that (RP) is successful for most embeddings, at the same time being capable to converge to the solution within the computational budget. In this regard, our analysis will attempt to answer the following question: What are optimal values of d and δ such that (2.3) is satisfied with 'high' probability?

7

3 Characterizing minimizers in the reduced space

The analysis of this section focuses on determining the distribution of the random minimizer $\boldsymbol{y}^*$ of $f(\boldsymbol{Ay})$, which satisfies $f(\boldsymbol{Ay}^*) = f^*$. These results will inform us on the effects of the different parameters on the success of (RP) allowing us to estimate the values of δ and d that are likely to increase the chances of successful recovery of f^*.

The following theorem provides a useful characterization of $\boldsymbol{y}^*$. The theorem and its proof were inspired by the proofs of Theorems 2 and 3 in [43].

Theorem 3.1. *Let Assumption 2.2 hold and let $\boldsymbol{x}^*_\top$ and $\mathcal{G}^*$ be defined as in Assumption 2.3. Let $\boldsymbol{A}$ be a $D \times d$ Gaussian matrix. Then, $\boldsymbol{y}^* \in \mathbb{R}^d$ satisfies $\boldsymbol{Ay}^* \in \mathcal{G}^*$ if and only if*

$$\boldsymbol{By}^* = \boldsymbol{z}^* \tag{3.1}$$

where the $d_e \times d$ random matrix $\boldsymbol{B}$ satisfies $\boldsymbol{B} = \boldsymbol{U}^T \boldsymbol{A}$ and where $\boldsymbol{z}^* \in \mathbb{R}^{d_e}$ is uniquely defined by $\boldsymbol{Uz}^* = \boldsymbol{x}^*_\top$.

Proof. Let $\boldsymbol{y}^* \in \mathbb{R}^d$ be such that $\boldsymbol{Ay}^* \in \mathcal{G}^*$. First, we establish that

$$\boldsymbol{Ay}^* \in \mathcal{G}^* \text{ if and only if } \boldsymbol{x}^*_\top = \boldsymbol{UU}^T \boldsymbol{Ay}^*. \tag{3.2}$$

Suppose that $\boldsymbol{Ay}^* \in \mathcal{G}^*$. Then, using the definition of $\mathcal{G}^*$ in Assumption 2.3 we can write $\boldsymbol{Ay}^* = \boldsymbol{x}^*_\top + \boldsymbol{x}_\perp$ for some $\boldsymbol{x}_\perp \in \mathcal{T}^\perp$. The orthogonal projection of $\boldsymbol{Ay}^*$ onto $\mathcal{T}$ is given by

$$\boldsymbol{UU}^T \boldsymbol{Ay}^* = \boldsymbol{UU}^T(\boldsymbol{x}^*_\top + \boldsymbol{x}_\perp) = \boldsymbol{x}^*_\top,$$

where we have used $\boldsymbol{UU}^T \boldsymbol{x}^*_\top = \boldsymbol{x}^*_\top$ and $\boldsymbol{UU}^T \boldsymbol{x}_\perp = \boldsymbol{0}$.

Conversely, assume that $\boldsymbol{y}^*$ satisfies

$$\boldsymbol{x}^*_\top = \boldsymbol{UU}^T \boldsymbol{Ay}^*. \tag{3.3}$$

Denote by $\boldsymbol{S}$ the $D \times D$ orthogonal matrix $(\boldsymbol{U} \ \boldsymbol{V})$, where $\boldsymbol{V}$ is defined in Assumption 2.2. Using (3.3) and the identity $\boldsymbol{UU}^T + \boldsymbol{VV}^T = \boldsymbol{SS}^T = \boldsymbol{I}_D$, we obtain

$$\boldsymbol{Ay}^* = (\boldsymbol{UU}^T + \boldsymbol{VV}^T)\boldsymbol{Ay}^* = \boldsymbol{x}^*_\top + \boldsymbol{VV}^T \boldsymbol{Ay}^*.$$

Note that $\boldsymbol{VV}^T \boldsymbol{Ay}^*$ lies on $\mathcal{T}^\perp$ as it is the orthogonal projection of $\boldsymbol{Ay}^*$ onto $\mathcal{T}^\perp$, which implies that $\boldsymbol{Ay}^* \in \mathcal{G}^*$. This completes the proof of (3.2).

Now we show that (3.1) and (3.3) are equivalent. We multiply both sides of $\boldsymbol{x}^*_\top = \boldsymbol{UU}^T \boldsymbol{Ay}^*$ by $\boldsymbol{S}^T$, and obtain

$$\begin{pmatrix} \boldsymbol{U}^T \\ \boldsymbol{V}^T \end{pmatrix} \boldsymbol{x}^*_\top = \begin{pmatrix} \boldsymbol{U}^T \\ \boldsymbol{V}^T \end{pmatrix} \boldsymbol{UU}^T \boldsymbol{Ay}^*. \tag{3.4}$$

Since $\boldsymbol{x}^*_\top$ is in the column span of $\boldsymbol{U}$, it can be written as $\boldsymbol{x}^*_\top = \boldsymbol{Uz}^*$ for some (unique) vector $\boldsymbol{z}^* \in \mathbb{R}^{d_e}$. By substituting the above into (3.4) we obtain

$$\begin{pmatrix} \boldsymbol{U}^T \boldsymbol{Uz}^* \\ \boldsymbol{V}^T \boldsymbol{Uz}^* \end{pmatrix} = \begin{pmatrix} \boldsymbol{U}^T \boldsymbol{UU}^T \boldsymbol{Ay}^* \\ \boldsymbol{V}^T \boldsymbol{UU}^T \boldsymbol{Ay}^* \end{pmatrix}.$$

This reduces to

$$\begin{pmatrix} \boldsymbol{z}^* \\ \boldsymbol{0} \end{pmatrix} = \begin{pmatrix} \boldsymbol{U}^T \boldsymbol{Ay}^* \\ \boldsymbol{0} \end{pmatrix},$$

where we have used the identities $\boldsymbol{U}^T \boldsymbol{U} = \boldsymbol{I}$ and $\boldsymbol{V}^T \boldsymbol{U} = \boldsymbol{0}$, which follow from Assumption 2.2. To obtain (3.3) from (3.1), multiply (3.1) by $\boldsymbol{U}$. $\qquad\square$

Remark 3.2. Thereafter, we write $\boldsymbol{B}$ to refer to the $d_e \times d$ Gaussian matrix[7] $\boldsymbol{U}^T \boldsymbol{A}$, where $\boldsymbol{A}$ is a $D \times d$ Gaussian matrix and where $\boldsymbol{U}$ is defined in Assumption 2.2. Furthermore, we write $\boldsymbol{z}^*$ to refer to the $d_e \times 1$ vector that satisfies $\boldsymbol{U}\boldsymbol{z}^* = \boldsymbol{x}_\top^*$, where $\boldsymbol{x}_\top^*$ is defined in Assumption 2.3. Observe that $\|\boldsymbol{z}^*\| = \|\boldsymbol{x}_\top^*\|$ since $\boldsymbol{U}\boldsymbol{z}^* = \boldsymbol{x}_\top^*$ and $\boldsymbol{U}$ is orthogonal.

Corollary 3.3. *Let Assumption 2.2 hold. Let* $S^* = \{\boldsymbol{y}^* : \boldsymbol{A}\boldsymbol{y}^* \in \mathcal{G}^*\}$, *where* $\boldsymbol{A}$ *is a* $D \times d$ *Gaussian matrix and where* $\mathcal{G}^*$ *is defined as in Assumption 2.3. Then, the following holds*

- *If* $d = d_e$, *then* S^* *has exactly one element with probability 1.*

- *If* $d > d_e$, *then* S^* *has infinitely many elements with probability 1.*

Proof. It follows from Theorem 3.1 that the set S^* and the set of solutions to $\boldsymbol{B}\boldsymbol{y} = \boldsymbol{z}^*$ coincide. According to Theorem A.3, $\boldsymbol{B}\boldsymbol{B}^T$ is positive definite with probability 1, which implies that $\text{rank}(\boldsymbol{B}\boldsymbol{B}^T) = d_e$ with probability 1. Since $\text{rank}(\boldsymbol{B}) = \text{rank}(\boldsymbol{B}\boldsymbol{B}^T)$, $\text{rank}(\boldsymbol{B}) = d_e$ with probability 1. Hence, the linear system $\boldsymbol{B}\boldsymbol{y} = \boldsymbol{z}^*$ almost surely has a solution. If $d = d_e$ the linear system has only one solution. If $d > d_e$ the system is underdetermined and, therefore, has infinitely many solutions. $\qquad\square$

3.1 Choosing a suitable minimizer

While S^* contains infinitely many solutions if $d > d_e$, it is sufficient that one of these solutions is contained in $\mathcal{Y}$ for (RP) to be successful. We proceed further by choosing one particular solution $\boldsymbol{y}^*$ that is easy to analyse and based on the analysis will adjust parameters δ and d appropriately to ensure that the chosen $\boldsymbol{y}^*$ falls inside the feasible set $\mathcal{Y}$ with high probability. The solutions that are likely to fall inside the feasible domain must be close to the origin. In this regard, we propose two candidates:

$$\boldsymbol{y}_2^* = \underset{\boldsymbol{y} \in \mathbb{R}^d}{\text{argmin}} \ \|\boldsymbol{y}\|_2 \qquad (3.5) \qquad \boldsymbol{y}_\infty^* = \underset{\boldsymbol{y} \in \mathbb{R}^d}{\text{argmin}} \ \|\boldsymbol{y}\|_\infty \qquad (3.6)$$
$$\text{s.t.} \ \ \boldsymbol{y} \in S^*, \qquad\qquad\qquad \text{s.t.} \ \ \boldsymbol{y} \in S^*.$$

Due to the definition of $\mathcal{Y}$ as a box, the minimizer (3.6) with the minimal infinity norm is particularly of interest. Since $\boldsymbol{y}_\infty^*$ has the smallest infinity norm among all solutions in S^*, knowledge of $\boldsymbol{y}_\infty^*$ would allow us to choose the smallest possible $\mathcal{Y}$ while ensuring that (RP) is successful. Despite this convenient fact, we found that it is more difficult to study $\boldsymbol{y}_\infty^*$ and have decided to investigate $\boldsymbol{y}_2^*$ instead.

Remark 3.4. For $d = d_e$, $\boldsymbol{y}_2^* = \boldsymbol{y}_\infty^*$ because S^* contains only one element.

Lemma 3.5. *Let Assumption 2.2 hold and let* $\mathcal{G}^*$ *and* $\boldsymbol{y}_2^*$ *be defined as in Assumption 2.3 and* (3.5), *respectively. Problem* (RP) *is successful in the sense of Definition 1.2 if* $\boldsymbol{y}_2^* \in \mathcal{Y}$.

Proof. Assume that $\boldsymbol{y}_2^* \in \mathcal{Y}$. Then, $\boldsymbol{y}_2^*$ is a feasible solution of (RP). By the definitions of $\boldsymbol{y}_2^*$ and S^*, we also have $\boldsymbol{A}\boldsymbol{y}_2^* \in \mathcal{G}^*$; this implies that $f(\boldsymbol{A}\boldsymbol{y}_2^*) = f^*$. Hence, (RP) is successful by Definition 1.2. $\qquad\square$

Corollary 3.6. *Let Assumption 2.2 hold. Problem* (3.5) *has a unique solution given by*

$$\boldsymbol{y}_2^* = \boldsymbol{B}^T(\boldsymbol{B}\boldsymbol{B}^T)^{-1}\boldsymbol{z}^*. \qquad (3.7)$$

---

[7]Since $\boldsymbol{U}$ is orthogonal, it follows from Theorem A.2 that $\boldsymbol{B} = \boldsymbol{U}^T\boldsymbol{A}$ is a Gaussian matrix.

Proof. It follows from Theorem 3.1 that the solution(s) of (3.5) must be equal to the solution(s) of the following problem

$$\min \ \|\boldsymbol{y}\|_2^2$$
$$\text{s.t. } \boldsymbol{B}\boldsymbol{y} = \boldsymbol{z}^*,$$

which has the solution (3.7). $\qquad\square$

3.2 Distribution of minimal Euclidean norm minimizer

The present section derives the distribution of $\boldsymbol{y}_2^*$ and its Euclidean norm.

The distribution of the Euclidean norm of $\boldsymbol{y}_2^*$

Theorem 3.7. *Let Assumption 2.2 hold and let $\boldsymbol{x}_\top^*$ and $\boldsymbol{y}_2^*$ be defined as in Assumption 2.3 and (3.5), respectively. Then, $\boldsymbol{y}_2^*$ satisfies*

$$\frac{\|\boldsymbol{x}_\top^*\|_2^2}{\|\boldsymbol{y}_2^*\|_2^2} \sim \chi_{d-d_e+1}^2.$$

Proof. The result almost immediately follows from Corollary 3.6 and Lemma A.15. These yield

$$\frac{\|\boldsymbol{z}^*\|^2}{\|\boldsymbol{y}_2^*\|^2} \sim \chi_{d-d_e+1}^2,$$

The result is implied by $\|\boldsymbol{z}^*\| = \|\boldsymbol{x}_\top^*\|$ (see Remark 3.2). $\qquad\square$

The above result is equivalent to saying that $\|\boldsymbol{y}_2^*\|^2/\|\boldsymbol{x}_\top^*\|^2$ follows the inverse chi-squared distribution with $d - d_e + 1$ degrees of freedom (see Definition A.7). The theorem reveals a linear dependence of $\|\boldsymbol{y}_2^*\|$ on $\|\boldsymbol{x}_\top^*\|$; larger values of $\|\boldsymbol{x}_\top^*\|$ contribute to the increase in the likelihood of $\boldsymbol{y}_2^*$ being further away from the origin. The theorem also suggests that $\|\boldsymbol{y}_2^*\|$ is independent of D as long as $\|\boldsymbol{x}_\top^*\|$ is fixed.

Corollary 3.8. *Let Assumption 2.2 hold. Let $\boldsymbol{x}_\top^*$ and $\boldsymbol{y}_2^*$ be defined as in Assumption 2.3 and (3.5), respectively. Then,*

$$\mathbb{P}[\|\boldsymbol{y}_2^*\|_2 \le \delta] = \mathbb{P}\left[\chi_{d-d_e+1}^2 \ge \frac{\|\boldsymbol{x}_\top^*\|_2^2}{\delta^2}\right]$$

for any $\delta > 0$.

Proof. For any $\epsilon > 0$, we have

$$\mathbb{P}\left[\|\boldsymbol{y}_2^*\|_2 \le \frac{\|\boldsymbol{x}_\top^*\|_2}{\epsilon}\right] = \mathbb{P}\left[\frac{\|\boldsymbol{x}_\top^*\|_2^2}{\|\boldsymbol{y}_2^*\|_2^2} \ge \epsilon^2\right] = \mathbb{P}[\chi_{d-d_e+1}^2 \ge \epsilon^2],$$

where the second equality follows from Theorem 3.7. By letting $\epsilon = \|\boldsymbol{x}_\top^*\|_2/\delta$, we obtain the result. $\qquad\square$

Corollary 3.9. *Let Assumption 2.2 hold and let $\boldsymbol{x}_\top^*$ and $\boldsymbol{y}_2^*$ be defined as in Assumption 2.3 and (3.5), respectively. Provided that $d - d_e > 1$ we have*

$$\mathbb{E}[\|\boldsymbol{y}_2^*\|^2] = \frac{\|\boldsymbol{x}_\top^*\|^2}{d - d_e - 1}. \tag{3.8}$$
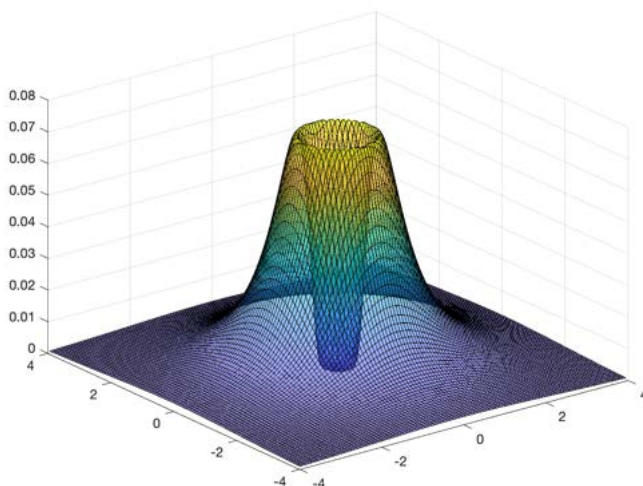
Figure 3: The illustration of the p.d.f. of $\boldsymbol{y}_2^*$ for $d = 2$, $k = 2$ and $\boldsymbol{x}_\top^* = [1\ 1]^T$.

Proof. Let W follow the inverse chi-squared distribution with $d - d_e + 1$ degrees of freedom. Then, $W \overset{law}{=} \|\boldsymbol{y}_2^*\|^2 / \|\boldsymbol{x}_\top^*\|^2$ by Theorem 3.7. By applying Lemma A.8, we obtain

$$\mathbb{E}[\|\boldsymbol{y}_2^*\|^2] = \mathbb{E}[\|\boldsymbol{x}_\top^*\|^2 W] = \frac{\|\boldsymbol{x}_\top^*\|^2}{d - d_e - 1}$$

for $d - d_e > 1$. $\qquad\qquad\square$

The expected value in (3.8) is inversely proportional to $d - d_e$. In other words, for a fixed d_e, larger values of the dimension of the embedding subspace bring $\boldsymbol{y}_2^*$ closer to the origin. This observation indicates that the increase in d allows us to decrease δ whilst the probability of $\boldsymbol{y}_2^* \in \mathcal{Y}$ is kept constant.

The probability density function

The following theorem derives the probability density function of $\boldsymbol{y}_2^*$.

Theorem 3.10. *Let Assumption 2.2 hold and let $\boldsymbol{x}_\top^*$ and $\boldsymbol{y}_2^*$ be defined as in Assumption 2.3 and (3.5), respectively. Then, the probability density function of $\boldsymbol{y}_2^*$ is given by*

$$g^*(\boldsymbol{y}) = \pi^{-d/2} \left(\frac{\Gamma(d/2)}{\Gamma(n/2)} \right) \left(\frac{\|\boldsymbol{x}_\top^*\|}{\sqrt{2}} \right)^n (\boldsymbol{y}^T \boldsymbol{y})^{-(n+d)/2} e^{-\|\boldsymbol{x}_\top^*\|^2/(2\boldsymbol{y}^T \boldsymbol{y})},$$

where $n = d - d_e + 1$.

Proof. Corollary 3.6 and Lemma A.17 imply that the p.d.f. of $\boldsymbol{y}_2^*$ is given by

$$g^*(\boldsymbol{y}) = \pi^{-d/2} \left(\frac{\Gamma(d/2)}{\Gamma(n/2)} \right) \left(\frac{\|\boldsymbol{z}^*\|}{\sqrt{2}} \right)^n (\boldsymbol{y}^T \boldsymbol{y})^{-(n+d)/2} e^{-\|\boldsymbol{z}^*\|^2/(2\boldsymbol{y}^T \boldsymbol{y})}.$$

By using the equation $\|\boldsymbol{z}^*\| = \|\boldsymbol{x}_\top^*\|$, we obtain the desired result. $\qquad\square$

Figure 3 illustrates the p.d.f. of two-dimensional $\boldsymbol{y}_2^*$. The shape of the p.d.f. resembles a volcano with the mass concentrated at a certain distance from the origin suggesting that $\boldsymbol{y}_2^*$ is unlikely to be neither too close to, nor too distant from the origin. We also note that the p.d.f. is independent of D.

11

4 Bounding the success of the reduced problem

This section is the culmination of this paper's analysis. Based on the results established earlier we derive a bound for the probability of success of (RP).

The following theorem presents a notable connection between the success of (RP) and the chi-squared distribution.

Theorem 4.1. *Let Assumption 2.2 hold. Then, for any $\delta > 0$, we have*

$$\mathbb{P}[(RP) \text{ is successful}] \geq \mathbb{P}\left[\chi^2_{d-d_e+1} \geq \frac{\min_{\boldsymbol{x}^* \in \mathcal{G}} \|\boldsymbol{x}^*\|_2^2}{\delta^2}\right]. \tag{4.1}$$

Proof. Note the following relationship between the probabilities:

$$\mathbb{P}[(RP) \text{ is successful}] \geq \mathbb{P}[\boldsymbol{y}_2^* \in \mathcal{Y}] = \mathbb{P}[\|\boldsymbol{y}_2^*\|_\infty \leq \delta] \geq \mathbb{P}[\|\boldsymbol{y}_2^*\|_2 \leq \delta], \tag{4.2}$$

where the first inequality follows from Lemma 3.5 and where the second inequality is implied by $\|\boldsymbol{y}_2^*\|_\infty \leq \|\boldsymbol{y}_2^*\|_2$. By applying Corollary 3.8 to the last probability in (4.2) and using the definition of $\boldsymbol{x}_\top^*$ given in Assumption 2.3, we obtain

$$\mathbb{P}[(RP) \text{ is successful}] \geq \mathbb{P}\left[\chi^2_{d-d_e+1} \geq \frac{\|\boldsymbol{U}\boldsymbol{U}^T\boldsymbol{x}^*\|_2^2}{\delta^2}\right] \tag{4.3}$$

for any $\delta > 0$ and any $\boldsymbol{x}^* \in \mathcal{G}$ such that $\|\boldsymbol{U}\boldsymbol{U}^T\boldsymbol{x}^*\|_2 \neq 0$. Note that (4.3) also holds for $\|\boldsymbol{U}\boldsymbol{U}^T\boldsymbol{x}^*\|_2 = 0$ since, in this case, (RP) is successful with probability 1 (see the discussion preceding Assumption 2.3). Hence, (4.3) holds for any $\boldsymbol{x}^* \in \mathcal{G}$, which then implies

$$\mathbb{P}[(RP) \text{ is successful}] \geq \max_{\boldsymbol{x}^* \in \mathcal{G}} \mathbb{P}\left[\chi^2_{d-d_e+1} \geq \frac{\|\boldsymbol{U}\boldsymbol{U}^T\boldsymbol{x}^*\|_2^2}{\delta^2}\right] = \mathbb{P}\left[\chi^2_{d-d_e+1} \geq \frac{\min_{\boldsymbol{x}^* \in \mathcal{G}} \|\boldsymbol{U}\boldsymbol{U}^T\boldsymbol{x}^*\|_2^2}{\delta^2}\right],$$

where the equality follows from the fact that the tail distribution $\mathbb{P}[X > x]$ of any random variable X is a monotonically decreasing function in x.

In what follows, we show that $\min_{\boldsymbol{x}^* \in \mathcal{G}} \|\boldsymbol{U}\boldsymbol{U}^T\boldsymbol{x}^*\|_2^2 = \min_{\boldsymbol{x}^* \in \mathcal{G}} \|\boldsymbol{x}^*\|_2^2$. Define sets $\mathcal{Z} = \{\boldsymbol{z} \in \mathbb{R}^d : \boldsymbol{U}\boldsymbol{z} = \boldsymbol{U}\boldsymbol{U}^T\boldsymbol{x}^*, \boldsymbol{x}^* \in \mathcal{G}\}$ and $\mathcal{S} = \{\boldsymbol{U}\boldsymbol{z} + \boldsymbol{V}\boldsymbol{c} : \boldsymbol{z} \in \mathcal{Z}, \boldsymbol{c} \in \mathbb{R}^{D-d_e}\}$, where $\boldsymbol{V}$ is defined in Assumption 2.3. First, we establish that $\mathcal{G} = \mathcal{S}$ by showing that $\mathcal{G} \subseteq \mathcal{S}$ and that $\mathcal{S} \subseteq \mathcal{G}$.

Let $\boldsymbol{x}^* \in \mathcal{G}$. We can write $\boldsymbol{x}^* = \boldsymbol{U}\boldsymbol{U}^T\boldsymbol{x}^* + \boldsymbol{V}\boldsymbol{V}^T\boldsymbol{x}^*$ since $\boldsymbol{U}\boldsymbol{U}^T + \boldsymbol{V}\boldsymbol{V}^T = \boldsymbol{I}$. Let $\boldsymbol{z} = \boldsymbol{U}^T\boldsymbol{x}^*$ and $\boldsymbol{c} = \boldsymbol{V}^T\boldsymbol{x}^*$ and note that $\boldsymbol{z} \in \mathcal{Z}$ and $\boldsymbol{c} \in \mathbb{R}^{D-d_e}$. Hence, $\boldsymbol{x}^* \in \mathcal{S}$, which proves that $\mathcal{G} \subseteq \mathcal{S}$.

Let $\boldsymbol{x}^* \in \mathcal{S}$. Then, $\boldsymbol{x}^* = \boldsymbol{U}\boldsymbol{z} + \boldsymbol{V}\boldsymbol{c}$ for some $\boldsymbol{z} \in \mathcal{Z}$ and $\boldsymbol{c} \in \mathbb{R}^{D-d_e}$. We have

$$f(\boldsymbol{x}^*) = f(\boldsymbol{U}\boldsymbol{z} + \boldsymbol{V}\boldsymbol{c}) = f(\boldsymbol{U}\boldsymbol{z}) = f(\boldsymbol{U}\boldsymbol{U}^T\boldsymbol{x}^*) = f(\boldsymbol{x}_\top^*) = f^*,$$

where the second equality follows from the assumption that f has low effective dimensionality and the fact that $\boldsymbol{V}\boldsymbol{c} \in \mathcal{T}^\perp$, the fourth equality follows from the definition of $\boldsymbol{x}_\top^*$ (given in Assumption 2.3) and the last equality follows from (2.2). Hence, by definition of $\mathcal{G}$, $\boldsymbol{x}^* \in \mathcal{G}$. This proves that $\mathcal{S} \subseteq \mathcal{G}$.

Finally, we have

$$\min_{\boldsymbol{x}^* \in \mathcal{G}} \|\boldsymbol{x}^*\|_2^2 = \min_{\boldsymbol{x}^* \in \mathcal{S}} \|\boldsymbol{x}^*\|_2^2 = \min_{\boldsymbol{z} \in \mathcal{Z}, \boldsymbol{c} \in \mathbb{R}^{D-d_e}} \|\boldsymbol{U}\boldsymbol{z} + \boldsymbol{V}\boldsymbol{c}\|_2^2 \qquad (\text{since } \mathcal{G} = \mathcal{S} \text{ and by definition of } \mathcal{S})$$

$$= \min_{\boldsymbol{z} \in \mathcal{Z}, \boldsymbol{c} \in \mathbb{R}^{D-d_e}} \|\boldsymbol{U}\boldsymbol{z}\|_2^2 + \|\boldsymbol{V}\boldsymbol{c}\|_2^2 \qquad (\text{since } \boldsymbol{U}^T\boldsymbol{V} = \boldsymbol{V}^T\boldsymbol{U} = \boldsymbol{0})$$

$$= \min_{\boldsymbol{z} \in \mathcal{Z}, \boldsymbol{c} \in \mathbb{R}^{D-d_e}} \|\boldsymbol{U}\boldsymbol{z}\|_2^2 + \|\boldsymbol{c}\|_2^2 \qquad (\text{since } \boldsymbol{V} \text{ is orthogonal})$$

$$= \min_{\boldsymbol{z} \in \mathcal{Z}} \|\boldsymbol{U}\boldsymbol{z}\|_2^2 + \min_{\boldsymbol{c} \in \mathbb{R}^{D-d_e}} \|\boldsymbol{c}\|_2^2$$

$$= \min_{\boldsymbol{z} \in \mathcal{Z}} \|\boldsymbol{U}\boldsymbol{z}\|_2^2 + 0$$

$$= \min_{\boldsymbol{x}^* \in \mathcal{G}} \|\boldsymbol{U}\boldsymbol{U}^T\boldsymbol{x}^*\|_2^2 \qquad (\text{by definition of } \mathcal{Z})$$

$\square$

Using Theorem 4.1, one can now bound the success of (RP) by applying any tail bound on the chi-squared distribution. We use the bound derived in Lemma A.6.

Corollary 4.2. *Let Assumption 2.2 hold and let* $\mu = \min_{\boldsymbol{x}^* \in \mathcal{G}} \|\boldsymbol{x}^*\|_2$. *Then, for any* $\delta > 0$, *we have*

$$\mathbb{P}[\text{(RP) is successful}] \geq 1 - C(n)\left(1 + \frac{n}{2}e^{-\mu^2/(2\delta^2)}\right)\left(\frac{\mu}{\sqrt{2}\delta}\right)^n, \tag{4.4}$$

where $n = d - d_e + 1$ *and*

$$C(n) = \frac{4}{n(n+2)\Gamma(n/2)}.$$

Proof. Lemma A.6 implies that

$$\mathbb{P}[\chi_n^2 \geq \epsilon^2] \geq 1 - C(n)\left(1 + \frac{n}{2}e^{-\epsilon^2/2}\right)(\epsilon^2/2)^{n/2} \tag{4.5}$$

for any $\epsilon > 0$. By letting $\epsilon = \mu/\delta$ and applying (4.5) to (4.1), we obtain the wished bound. $\square$

Let R^* denote the right hand side of (4.4). First, we note that R^* is a function of μ/δ and $d - d_e$. The bound reveals a linear relationship between μ and δ; scaling μ and δ by the same factor does not affect the value of R^*. Furthermore, observe that for smaller values of μ or larger values of δ, R^* is closer to 1. Numerical experiments show that for large values of n and/or μ/δ, the bound (4.4) is less tight; this is also signified by the asymptotic behaviour of R^*, $R^* \to -\infty$ monotonically as $\mu/\delta \to \infty$ making the bound useless for large enough μ/δ.

It is remarkable that R^* has no dependence on D, the dimension of the original optimization problem. This implies that larger D does not diminish the success of the reduced problem as long as μ and d_e are unchanged. Dependence of R^* on $d - d_e$ indicates that the success is determined by the value of d relative to d_e and not so much by the individual values of d and d_e. Larger (smaller) values of d with respect to d_e require smaller (larger) δ if R^* is kept constant; knowing this fact is crucial when initializing values of d and δ in practice. It displays a convenient interplay between d and δ allowing more flexibility in choosing one versus another.

Previous bounds. One can derive similar bounds for the success of (RP) by bounding $\mathbb{P}[\|\boldsymbol{y}_2^*\| \leq \delta]$ in (4.2) using the Cauchy-Schwarz inequality. Since $\boldsymbol{y}_2^* = \boldsymbol{B}^T(\boldsymbol{B}\boldsymbol{B}^T)^{-1}\boldsymbol{z}^*$, we have

$$\|\boldsymbol{y}_2^*\| \leq \|\boldsymbol{B}^T(\boldsymbol{B}\boldsymbol{B}^T)^{-1}\| \cdot \|\boldsymbol{z}^*\|.$$

By using the fact that $\|\boldsymbol{B}^T(\boldsymbol{B}\boldsymbol{B}^T)^{-1}\| = 1/s_{\min}(\boldsymbol{B}^T)$, where $s_{\min}(\boldsymbol{B}^T)$ denotes the smallest singular value of $\boldsymbol{B}^T$, we obtain

$$\mathbb{P}[\|\boldsymbol{y}_2^*\| \leq \delta] \geq \mathbb{P}\left[\frac{\|\boldsymbol{z}^*\|}{s_{\min}(\boldsymbol{B}^T)} \leq \delta\right].$$

We can now use any suitable tail bound for the smallest singular value of the Gaussian matrix to bound the latter probability.

Wang et al. [43], by applying the above technique and the result in [12] to bound the singular value, derived the following bound

$$\mathbb{P}[\text{(RP) is successful}] \geq 1 - \frac{\mu\sqrt{d_e}}{\delta}.$$

13

Their derivation is predicated on the assumptions that $d = d_e$ and that $\mathcal{T}$ is spanned by the standard basis vectors. In [38], Sanyang and Kabán extended Wang et al.'s bound to any δ satisfying $\delta > \|\boldsymbol{x}_\top^*\|/(\sqrt{d} - \sqrt{d_e})$. Using the bound in [9] for $s_{\min}(\boldsymbol{B}^T)$ they showed that

$$\mathbb{P}[(\text{RP}) \text{ is successful}] \geq 1 - e^{-(\sqrt{d} - \sqrt{d_e} - \mu/\delta)^2/2}$$

One can also use Rudelson and Vershynin's bound in [36, Theorem 1.1] to obtain

$$\mathbb{P}[(\text{RP}) \text{ is successful}] \geq 1 - \left(\frac{C\mu}{\delta(\sqrt{d} - \sqrt{d_e - 1})}\right)^{d - d_e + 1} - e^{-cd},$$

where $C, c > 0$ are absolute constants. This bound shows dependence of the probability on the difference $d - d_e$, which is also manifest in our bound. The Rudelson and Vershynin's bound cannot be used for practical purposes due to the unknown C and c; we require explicit bounds to define the size of $\mathcal{Y}$.

Unlike the bounds of Wang et al. [43] and Sanyang and Kaban [38], Corollary 3.8 is applicable to any $d \geq d_e$ and an arbitrary subspace $\mathcal{T}$. Moreover, using the exact distribution of $\|\boldsymbol{y}_2^*\|$ given in Corollary 3.8, we circumvent the application of the intermediate Cauchy-Schwarz and bound the distribution of $\|\boldsymbol{y}_2^*\|$ directly.

Affine random embeddings. It is not difficult to extend (RP) to affine random subspace embeddings. In the affine case, we replace $\boldsymbol{x}$ by $\boldsymbol{Ay} + \boldsymbol{p}$, where $\boldsymbol{p} \in \mathbb{R}^D$ is a fixed point. The reduced optimization problem is then given by

$$\min \; f(\boldsymbol{Ay} + \boldsymbol{p})$$
$$\text{subject to} \; \boldsymbol{y} \in \mathcal{Y},$$

The results that apply to the linear embeddings also apply to the affine embeddings after minor adjustments. Theorem 2.4, for example, can be easily extended to the affine case to show that the intersection between $\boldsymbol{p} + \text{range}(\boldsymbol{A})$ and $\mathcal{G}$ takes place with probability 1 if $d \geq d_e$. The affine version of the results are provable with the same assumptions except for a minor alteration in Assumption 2.3: the condition $\boldsymbol{x}_\top^* \neq \boldsymbol{0}$ changes to $\boldsymbol{x}_\top^* \neq \boldsymbol{p}$. To obtain the affine versions of Theorem 3.7 and Theorem 3.10, replace $\boldsymbol{x}_\top^*$ with $\boldsymbol{x}_\top^* - \boldsymbol{p}_\top$, where $\boldsymbol{p}_\top = \boldsymbol{UU}^T\boldsymbol{p}$ is the orthogonal projection of $\boldsymbol{p}$ onto $\mathcal{T}$. For the affine versions of Theorem 4.1 and Corollary 4.2, replace $\min_{\boldsymbol{x}^* \in \mathcal{G}} \|\boldsymbol{x}^*\|$ with $\min_{\boldsymbol{x}^* \in \mathcal{G}} \|\boldsymbol{x}^* - \boldsymbol{p}\|$.

5 Numerical experiments

5.1 Choices of (RP) parameters

The present section aims to test numerically the quality of the bound (4.4). We will also use the results of this section to select suitable pairs of parameters d and δ for (RP) in the numerical experiments later.

Suppose that we are given a function f satisfying Assumption 2.2 with the set of global minimizers $\mathcal{G}$ consisting of only one connected component. Let $\boldsymbol{x}_\top^*$ for f be defined as in Assumption 2.3 and $\boldsymbol{z}$ be defined by the equation $\boldsymbol{Uz} = \boldsymbol{x}_\top^*$. We also define $\mu := \min_{\boldsymbol{x}^* \in \mathcal{G}} \|\boldsymbol{x}^*\|$ and note that $\mu = \|\boldsymbol{x}_\top^*\|$.

We test (4.4) for f by contrasting the left-hand side of (4.4) (denoted by L^*) to its right-hand side (denoted by R^*). We compare L^* and R^* for four different values of $d - d_e$, namely, 0, 1, 2 and 3. For each value of $d - d_e$, we express R^* as a function of $\bar{\delta} := \delta/\mu$ and using its closed form we plot R^* for $\bar{\delta} \in [0.02, 10]$. We do not have a closed form expression for L^*, but we can
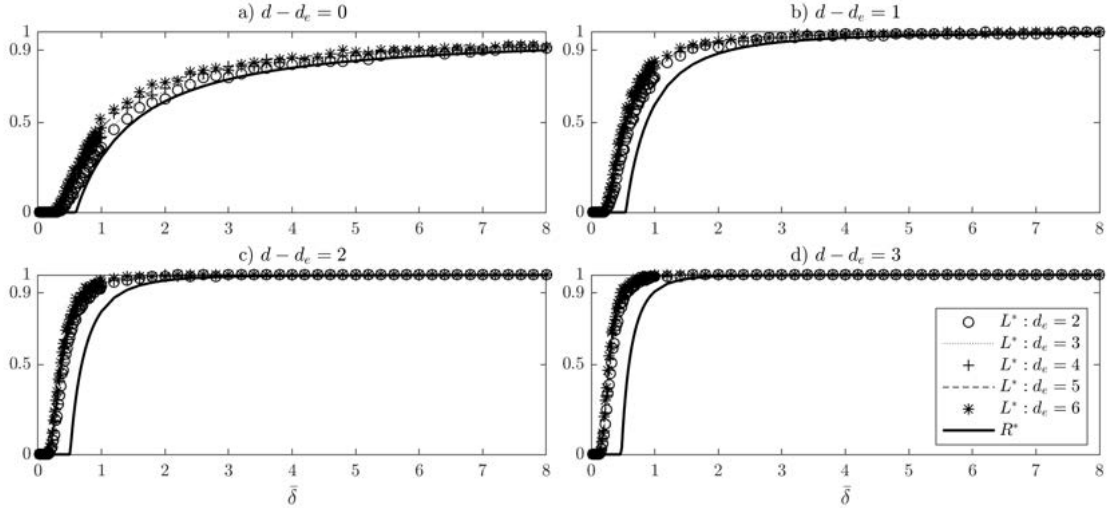
Figure 4: The four plots depict the function $R^*(\bar{\delta})$ and the estimates of $L^*(\bar{\delta})$; each plot corresponds to a particular value of $d - d_e \in \{0, 1, 2, 3\}$. Each plot contains estimates of $L^*(\bar{\delta})$ for $d_e = 2, 3, 4, 5, 6$.

approximate it numerically. In what follows, we describe how this could be done. We start by writing

$$L^* := \mathbb{P}[(\text{RP}) \text{ is successful}] = \mathbb{P}[\exists \boldsymbol{y} \in [-\delta, \delta]^d : \boldsymbol{A}\boldsymbol{y} \in \mathcal{G}] = \mathbb{P}[\exists \boldsymbol{y} \in [-\delta, \delta]^d : \bar{\boldsymbol{B}}\boldsymbol{y} = \boldsymbol{z}]$$
$$= \mathbb{P}[\exists \boldsymbol{y} \in [-\bar{\delta}, \bar{\delta}]^d : \bar{\boldsymbol{B}}\boldsymbol{y} = \bar{\boldsymbol{z}}],$$
(5.1)

where $\bar{\boldsymbol{B}}$ denotes a $d_e \times d$ Gaussian matrix and $\bar{\boldsymbol{z}} = \boldsymbol{z}/\mu$. Here, the second equality follows from Definition 1.2, and the third equality follows from Theorem 3.1 and the fact that $\mathcal{G}$ has only one connected component. Note that $\|\bar{\boldsymbol{z}}\| = 1$ since $\|\boldsymbol{z}\| = \|\boldsymbol{x}_\top^*\| = \mu$ (see Remark 3.2). We assign $\bar{\boldsymbol{z}}$ to a random vector with unit norm and keep $\bar{\boldsymbol{z}}$ fixed throughout the experiment[8]. For each $\bar{\delta} \in [0.02, 10]$, we generate a thousand Gaussian matrices $\bar{\boldsymbol{B}}$ and estimate the latter probability in (5.1) as the proportion of instances for which the statement under the probability is true. Unlike for R^*, L^* depends on individual values of d and d_e. We plot the estimates for L^* for the following values of $d - d_e$: 0, 1, 2, 3 and, in each plot, we repeat the experiment for $d_e = 2, 3, 4, 5, 6$. The plots are presented in Figure 4.

Numerical findings. The plots in Figure 4 — confirming the conclusions of Corollary 4.2 — suggest that the variation in success of (RP) is mainly determined by the value of $d - d_e$; the larger is the difference, the higher is the probability of success of (RP) for a given $\bar{\delta}$. These curves, being independent of μ, can be used to find suitable $\bar{\delta}$ for any problem for the corresponding values of $d - d_e$; the size of the $\mathcal{Y}$ box, δ, can then be set to $M\bar{\delta}$ if an upper bound M on μ is known.

Choosing d and δ in practice. When it comes to the numerical application of (RP) in practice, initialization of parameters d and δ might be problematic. From the theoretical discussions above we learned that the parameters d and δ must be defined based on d_e and μ, the values of

---

[8]Note that the results of the experiment are invariant of the choice of $\bar{\boldsymbol{z}}$ as long as its norm is fixed. Let $\boldsymbol{z}_1$ and $\boldsymbol{z}_2$ be two fixed vectors with unit norm. Consider two systems: $\boldsymbol{B}\boldsymbol{y} = \boldsymbol{z}_1$ and $\boldsymbol{B}\boldsymbol{y} = \boldsymbol{z}_2$. Note that $\boldsymbol{z}_2$ can be written as $\boldsymbol{Q}\boldsymbol{z}_1$ for some orthogonal $\boldsymbol{Q} \in \mathbb{R}^{d_e \times d_e}$. Then, the second system becomes $\boldsymbol{Q}^T \boldsymbol{B}\boldsymbol{y} = \boldsymbol{z}_1$ and this generates vectors $\boldsymbol{y}$ with the same distribution as the first system since $\boldsymbol{Q}^T \boldsymbol{B}$ is also Gaussian.

Algorithm 1 Random Embeddings for Global Optimization (REGO) applied to (P).

1: Initialise d and δ and define $\mathcal{Y} = [-\delta, \delta]^d$
2: Generate a $D \times d$ Gaussian matrix $\boldsymbol{A}$
3: Apply a global optimization solver (e.g. BARON, DIRECT, KNITRO) to (RP) until a termination criterion is satisfied, and define $\boldsymbol{y}_{min}$ to be the generated (approximate) solution of (RP).
4: Reconstruct $\boldsymbol{x}_{min} = \boldsymbol{A}\boldsymbol{y}_{min}$

---

which are typically unknown in practice, for example, for black-box functions. We circumvent this issue by estimating d_e and μ rather than trying to calculate their exact values; note that all we need is an upper bound d on d_e. The parameter d_e or an upper bound may be known from prior studies or can be found with active subspace identification methods (see, e.g., [8]); these use gradients of f to estimate d_e.

Estimating μ can be a harder task. A rough estimate for μ can be obtained if the search in the original space is restricted to a certain domain; a trivial upper bound in this case is given by the maximum distance between the origin and the boundary of the domain. The search domain that is commonly imposed to practically solve unconstrained optimization problems is box constraints, such as $\mathcal{X} = [-1, 1]^D$ for which $\mu \leq \sqrt{D}$. In Appendix C, we test REGO assuming that $\sqrt{D}$ is the best bound known for μ. To compensate for unknown μ, one could also try increasing δ or d gradually to explore larger regions in $\mathbb{R}^D$.

5.2 Testing REGO with state-of-the-art global solvers

Algorithms. The algorithm for the random embeddings method named REGO (Random Embeddings for Global Optimization) is outlined in Algorithm 1. Below, we give the descriptions of the three state-of-the-art solvers we use to test REGO.

DIRECT([18, 24, 16]) version 4.0 (DIviding RECTangles) is a deterministic[9] global optimization solver first introduced in [24] as an extension of Lipschitzian optimization. DIRECT does not require information about the gradient nor about the Lipschitz constant and, hence, can be used for black-box functions. DIRECT divides the search domain into rectangles and evaluates the function at the centre of each rectangle. Based on the previously sampled points, DIRECT carefully decides what rectangle to divide next balancing between local and global searches. Jones et al. [24] showed that DIRECT is guaranteed to converge to global minimum, but convergence may sometimes be slow.

BARON([37, 40]) version 17.10.10 (Branch-And-Reduce Optimization Navigator) is a branch-and-bound type global optimization solver for non-linear and mixed-integer non-linear programs. To provide lower and upper bounds for each branch, BARON utilizes algebraic structure of the objective function. It also includes a preprocessing step where it performs a multi-start local search to obtain a tight global upper bound. In comparison to other existing global solvers, BARON was demonstrated to be the most robust and fastest (see [30]). However, BARON accepts only a few (general) classes of functions[10] including polynomial, exponential, logarithmic, etc. and, unlike DIRECT, it is unable to optimize black-box functions.

KNITRO([6]) version 10.3.0 is a large-scale non-linear local optimization solver capable of handling problems with hundreds of thousands of variables. KNITRO allows to solve problems using one of the four algorithms: two interior point type methods (direct and conjugate gradient) and two active set type methods (active set and sequential quadratic programming). In contrast

---

[9]Here, we refer to the predictable behaviour of the solver given a fixed set of parameters.

[10]For instance, BARON cannot be applied to problems which include trigonometric functions.

to BARON and DIRECT, which specialize on finding global minima, KNITRO focuses on finding local solutions. Nonetheless, KNITRO has multi-start capabilities, i.e., it solves a problem locally multiple times every time starting from a different point in the feasible domain. It is this feature that we make use of in the experiments.

Generating the test set. Our test set of functions with low effective dimensionality will be derived from 19 global optimization problems (of dimensions 2–6) with known global minima [20, 14, 39], some of which are from the Dixon-Szego set [10]. The list of the problems is given in Table 3, Appendix B.

Below we describe the method adopted from Wang et al. [43] to generate high-dimensional functions with low effective dimensionality. Let $\bar{g}(\bar{\boldsymbol{x}})$ be any function from Table 3; let d_e be its dimension and let the given domain be scaled to $[-1, 1]^{d_e}$. We create a D-dimensional function $g(\boldsymbol{x})$ by adding $D - d_e$ fake dimensions to $\bar{g}(\bar{\boldsymbol{x}})$, $g(\boldsymbol{x}) = \bar{g}(\bar{\boldsymbol{x}}) + 0 \cdot x_{d_e+1} + 0 \cdot x_{d_e+2} + \cdots + 0 \cdot x_D$. We further rotate the function by applying a random orthogonal matrix $\boldsymbol{Q}$ to $\boldsymbol{x}$ to obtain a non-trivial constant subspace. The final form of the function we test is given as

$$f(\boldsymbol{x}) = g(\boldsymbol{Q}\boldsymbol{x}). \tag{5.2}$$

Note that the first d_e rows of $\boldsymbol{Q}$ now span the effective subspace $\mathcal{T}$ of $f(\boldsymbol{x})$. Furthermore,

$$\mu := \min_{\boldsymbol{x} \in \mathcal{G}_1} f(\boldsymbol{x}) = \min_{\bar{\boldsymbol{x}} \in \mathcal{G}_2} \bar{g}(\bar{\boldsymbol{x}}) \le \sqrt{d_e}, \tag{5.3}$$

where $\mathcal{G}_1$ and $\mathcal{G}_2$ are the sets of global minimizers of f and $\bar{g}$, respectively.

For each problem in the test set, we generate three functions f as defined in (5.2) one for each $D = 10, 100, 1000$. We will tackle (P) for each f both directly (we call it '*no embedding*') and applying REGO outlined in Algorithm 1.

Experimental setup (REGO). We compare 'no embedding' and REGO using the three solvers above. Let g_i, s_j, n_j and D_k denote the ith function in the problem set ($g_1 =$ Beale, etc., see Table 3), jth solver ($s_1 =$ DIRECT, $s_2 =$ BARON, $s_3 =$ KNITRO), the total number of problems in the problem set solvable by jth solver ($n_1 = 19$, $n_2 = 15$, $n_3 = 18$) and kth ambient dimension ($D_1 = 10$, $D_2 = 100$, $D_3 = 1000$), respectively. Let f_{ik} denote the D_k-dimensional function with low effective dimensionality constructed from g_i as described previously.

Within 'no embedding' framework, for each pair (s_j, D_k), we solve f_{ik} for $i = 1, 2, \ldots, n_j$ with solver s_j and record the proportion of the problems that attain convergence (see definition in Table 1).

For each f_{ik} ($1 \le i \le n_j$, $1 \le k \le 3$), we apply REGO 100 times every time with a different Gaussian matrix. Thus, in total, for each pair (s_j, D_k) we solve $n_j \times 100$ problems. We record the proportion of problems that attain convergence (see Table 1) out of these $n_j \times 100$ problems.

We also record the number of function evaluations (for DIRECT and KNITRO) and CPU time (for all the three solvers) spent before termination within the two frameworks. For each (s_j, D_k), function evaluations and time are averaged out over $n_j \times 100$ problems within REGO and over n_j problems within 'no embedding'.

We conduct the above experiment for REGO with the following pairs of parameters (d, δ): $(d_e, 8.0 \times \sqrt{d_e})$, $(d_e + 1, 2.2 \times \sqrt{d_e})$, $(d_e + 2, 1.3 \times \sqrt{d_e})$ and $(d_e + 3, 1.0 \times \sqrt{d_e})$. Here, each δ was set to $M\bar{\delta}$, where $M = \sqrt{d_e}$ is an upper bound on μ (see (5.3)) and the value for $\bar{\delta}$ was chosen as the smallest $\bar{\delta}$ that gives at least 90% chance of success based on the curve of R^* in Figure 4.

Experimental setup (solvers). Due to the difference in algorithmic procedures of the solvers, they allow different budget constraints and have different convergence and termination criteria; we present these in Table 1.

Table 1: The table outlines the experimental setup for the three solvers. In the table, f is a function with low effective dimensionality d_e and the global minimum f^*, and ϵ is set to 10^{-3}.

	DIRECT	BARON	KNITRO
Measure of computational cost	function evaluations	CPU seconds	function evaluations, CPU seconds
Budget per problem	$10000 \times d_e$ function evaluations	$200 \times d_e$ CPU seconds	$20 \times d_e$ starting points
Convergence criteria (see Remark 5.1)	$f_D^* \leq f^* + \epsilon$	Convergence: $f_B^U \leq f^* + \epsilon$ Convergence$_{opt}$: $f_B^U \leq f_B^L + \epsilon$	$f_K^* \leq f^* + \epsilon$
Termination criteria	Either on budget or if $\boldsymbol{x}_D^*$ satisfies the convergence criteria	Either on budget or if f_B^U and f_B^L satisfy the convergence$_{opt}$ criteria	On budget

Additional options	`options.testflag=1`
	 `options.maxits=Inf`
	 `options.globalmin`=f^*
		`npsol = 9`
		 `numloc = 0`
		 `BrVarStra = 1`
		 `BrPtStra = 1`
			Default options. Derivatives are allowed. Use of multi-start through `ms_enable`=1.

Remark 5.1. DIRECT, at its every iteration, stores f_D^* — the minimum value of f so far found. BARON, at its every iteration, stores f_B^U and f_B^L — smallest upper bound and largest lower bound so far found for f. As for KNITRO, $f_K^* = \min\{f(\boldsymbol{A}\boldsymbol{y}_1^*), f(\boldsymbol{A}\boldsymbol{y}_2^*), \ldots, f(\boldsymbol{A}\boldsymbol{y}_l^*)\}$, where l is the number of starting points and where $\{\boldsymbol{y}_i^*\}_{1 \leq i \leq l}$ are the local solutions produced by the multi-start procedure.

Remark 5.2. The experiments are done not to compare solvers but to contrast 'no embedding' with REGO. All the experiments were run in MATLAB on the 16 cores (2×8 Intel with hyper-threading) Linux machines with 256GB RAM and 3300 MHz speed.

5.3 Numerical results

(RP) **successful.** We record the proportion of instances for which (RP) is successful. Table 2 presents these percentages for each particular choice of d and D averaged over 19 problems in the test set. We observe that the percentages are very high and appear to be independent of D supporting the conclusions of Corollary 4.2.

Table 2: The table shows average percentages of problems for which (RP) is successful.

d/D	10	100	1000
$d_e + 0$	97.2	97.8	97.3
$d_e + 1$	99.1	98.9	99.3
$d_e + 2$	99.5	99.6	99.8
$d_e + 3$	100	99.9	99.8

REGO vs. no embedding. The results of the experiment comparing REGO and 'no embedding' are presented in Figure 5, Figure 6 and Figure 7 for DIRECT, BARON and KNITRO, respectively. These figures compare average proportions of converged solutions and computational costs produced by REGO and 'no embedding' frameworks for $D = 10, 100, 1000$.
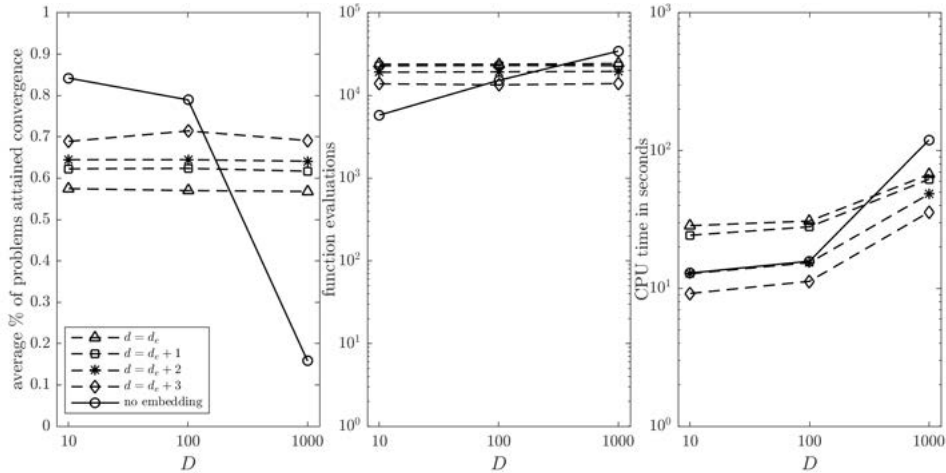
Figure 5: REGO versus 'no embedding' with DIRECT: comparison of frequency of convergence, log of average function evaluations and log of average CPU time (in seconds).

DIRECT (Figure 5). For all the four initialisations of REGO, we observe that the average proportions of problems that attained convergence (see definition in Table 1) are invariant with respect to the ambient dimension. This frequency of convergence is higher within 'no embedding' for $D = 10, 100$, but exhibits a significant drop for $D = 1000$. The average function evaluation count is maintained within REGO, but doubles within 'no embedding' for a tenfold increase in D. Growth in CPU time takes place within both frameworks, being highest for 'no embedding'.

BARON (Figure 6). In comparison with 'no embedding', the frequency of convergence$_{opt}$ is higher within REGO in most cases. We note that BARON's both convergence and convergence$_{opt}$ exhibit invariance with respect to D within REGO. As for 'no embedding', we observe a decrease in the frequencies of both convergence and convergence$_{opt}$. In addition, we observe an increase in CPU time spent within 'no embedding', whilst the time is almost constant within REGO.

KNITRO (Figure 7). We see that the proportion of solved problems is invariant with respect to the ambient dimension within REGO and, surprisingly, within 'no embedding' as well. However, the average number of function evaluations and time spent differ significantly between the two frameworks. With REGO, the average number of function evaluations remain at the same level for all D. Average time grows within both frameworks, but at a higher rate for 'no embedding'. The average time differs by a factor of 70 for $D = 1000$ in favour of REGO. We think that the growth in time within REGO is due to more costly function and derivative evaluations for larger D.

Summary of numerical results

1. (Effects of parameter choices) Our experiments clearly show that the choice of d and δ has a considerable effect on convergence and computational cost of REGO, and that good choices of (d, δ) are dependent on the underlying solver. For example, BARON achieved highest proportion of convergence$_{opt}$ within least amount of time for $(d, \delta) = (d_e, 8\sqrt{d_e})$, whereas DIRECT performed best for $(d_e + 3, \sqrt{d_e})$. KNITRO produced highest proportion of convergence and worst time for $(d_e + 3, \sqrt{d_e})$, and lowest proportion of convergence and best time for $(d_e, 8\sqrt{d_e})$.

2. (Scalability) Within REGO, the proportion of problems solved and/or number of function evaluations are generally invariant with respect to the ambient dimension D. REGO displays good scalability for all three solvers.
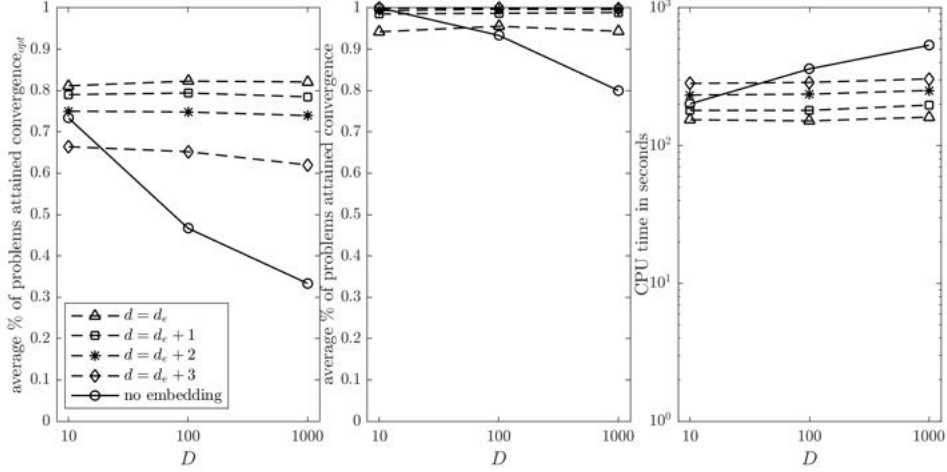
Figure 6: REGO versus 'no embedding' with BARON: comparison of frequency of convergence$_{opt}$/convergence and average CPU time (in seconds).
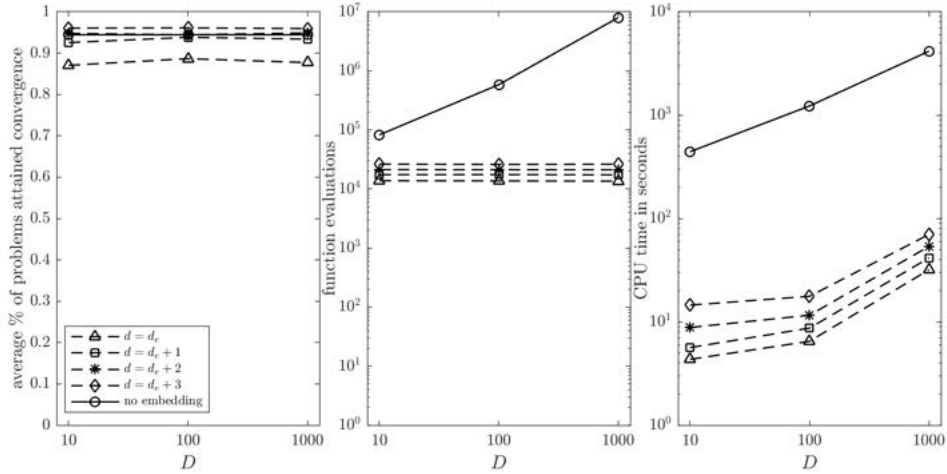

Figure 7: REGO versus 'no embedding' with KNITRO: comparison of frequency of convergence, log of average function evaluations and log of average CPU time (in seconds).

3. (No embedding) Within 'no embedding', as D increases, the proportion of problems that attained convergence$_{opt}$/convergence decreased for BARON and DIRECT. Surprisingly, for the KNITRO's multi-start method, the proportion of solved problems is maintained, but the number of function evaluations and time increased dramatically.

Additional experiments. To see how robust REGO is to the changes in the parameters, we conduct three more experiments presented and discussed in Appendix C. In the first experiment, assuming that μ is bounded by $\sqrt{D}$ (see page 16 for an explanation for this choice), we set δ to $\sqrt{D}\bar{\delta}$ for $\bar{\delta}$ chosen as in the main experiment. The second experiment tests REGO for four different values of d while keeping δ fixed and the third experiment tests REGO for three different values of δ keeping d fixed. In all three experiments, REGO performs well, particularly for BARON and KNITRO, solving most of the problems and exhibiting similar trends as in the main experiment.

6 Conclusions and future work

We study a general algorithmic framework for functions with low effective dimensionality that solves the reduced problem (RP) using a single Gaussian random embedding and a(ny) general global optimization solver. Our precise theoretical findings backed by the numerical experiments show that the success of (RP) is essentially independent of D and mainly depends on the gap between the embedding dimension d and the dimension of effective subspace d_e, and the ratio between the size of $\mathcal{Y}$ (namely δ) and μ (the Euclidean distance to the closest affine subspace of minimizers). REGO with three standard global solvers produced high frequencies of convergence, generally outperforming the respective solver's performance when applied directly to the problems (without the dimensionality reduction) in terms of proportion of problems solved and/or computational cost.

Our in-depth investigations are conceptual in nature, and there is clearly more work that needs to be done to make this framework practically applicable to global optimization problems with special structure. In particular, as outlined on page 16, our REGO approach depends on knowing (an upper bound d on) the effective dimension d_e. Future work may include estimating d_e prior to optimizing, noting that REGO does not need to learn the entire effective subspace only its dimension. One could also estimate d or d_e numerically, as proposed in [38], where d is gradually increased until no significant changes in the best function value found are observed. Our theoretical choices for δ also depend on μ, which again needs estimating. In this case, choosing a box domain for f would provide a rough estimate for μ (as discussed on page 16), with the remark that REGO cannot (yet) guarantee feasibility with respect to given bounds. To achieve the latter, one needs to either add projection operators as in [43, 4, 5], or include the problem constraints in the formulation of (RP) and allow multiple random embeddings as in [34]. Alternative potential directions include investigating other random or deterministic matrix choices for the embeddings, as considered for example in [29]. Real-life problems are often only approximately low dimensional, and so their optimization requires further extensions and analysis of the random embedding framework.

References

[1] M. Ben Salem, F. Bachoc, O. Roustant, F. Gamboa, and L. Tomaso. Sequential dimension reduction for learning features of expensive black-box functions. 2019.

[2] J. Bergstra and Y. Bengio. Random search for hyper-parameter optimization. *Journal of Machine Learning Research*, 13(1):281–305, 2012.

[3] J. M. Bernardo and A. F.M. Smith. *Bayesian theory*. Wiley, 2000.

[4] M. Binois, D. Ginsbourger, and O. Roustant. A warped kernel improving robustness in Bayesian optimization via random embeddings. *arXiv e-prints*, page arXiv:1411.3685, 2014.

[5] M. Binois, D. Ginsbourger, and O. Roustant. On the choice of the low-dimensional domain for global optimization via random embeddings. *arXiv e-prints*, page arXiv:1704.05318, 2017.

[6] R. H. Byrd, J. Nocedal, and R. A. Waltz. *Knitro: An Integrated Package for Nonlinear Optimization*, pages 35–59. Springer US, Boston, MA, 2006.

[7] B. Chen, A. Krause, and R. M. Castro. Joint optimization and variable selection of high-dimensional gaussian processes. In *Proceedings of the 29th International Conference on Machine Learning (ICML-12)*, pages 1423–1430, 2012.

[8] P. Constantine. *Active Subspaces*. SIAM, Philadelphia, PA, 2015.

[9] K.R. Davidson and S. Szarek. Local operator theory, random matrices and banach spaces. pages 317–366, 2001.

[10] L.C.W. Dixon and G.P. Szegö. *Towards Global Optimization*. Elseiver, New York, 1975.

[11] J. Djolonga, A. Krause, and V. Cevher. High-dimensional gaussian process bandits. In *Proceedings of the 26th International Conference on Neural Information Processing Systems*, NIPS'13, pages 1025–1033, 2013.

[12] A. Edelman. Eigenvalues and condition numbers of random matrices. *SIAM Journal on Matrix Analysis and Applications*, 9(4):543–560, 1988.

[13] D. Eriksson, K. Dong, E. H. Lee, D. Bindel, and A. G. Wilson. Scaling gaussian process regression with derivatives. In *Proceedings of the 32nd International Conference on Neural Information Processing Systems*, NIPS'18, pages 6868–6878, 2018.

[14] P.A. Ernesto and U.P. Diliman. MVF—multivariate test functions library in C for unconstrained global optimization. 2005.

[15] K. Fang, S. Kotz, and K. W. Ng. *Symmetric multivariate and related distributions*. London: Chapman and Hall, 1990.

[16] D. E. Finkel. Direct optimization algorithm user guide. 2003. Available at http://www2.peq.coppe.ufrj.br/Pessoal/Professores/Arge/COQ897/Naturais/DirectUserGuide.pdf.

[17] M. Fornasier, K. Schnass, and J. Vybiral. Learning functions of few arbitrary linear parameters in high dimensions. *Foundations of Computational Mathematics*, 12(2):229–262, 2012.

[18] J.M. Gablonsky and C.T. Kelley. A locally-biased form of the direct algorithm. *Journal of Global Optimization*, 21(1):27–37, 2001.

[19] R. Garnett, M. A. Osborne, and P. Hennig. Active learning of linear embeddings for gaussian processes. In *Proceedings of the Thirtieth Conference on Uncertainty in Artificial Intelligence*, UAI'14, pages 230–239, 2014.

[20] A. Gavana. Global optimization benchmarks and AMPGO. Available at http://infinity77.net/global_optimization/.

[21] A.K. Gupta and D.K. Nagar. *Matrix Variate Distributions*. New York: Chapman and Hall/CRC, 2000.

[22] A.K. Gupta and D. Song. Lp-norm spherical distribution. *Journal of Statistical Planning and Inference*, 60 (2):241–260, 1997.

[23] F. Hutter, H. Hoos, and K. Leyton-Brown. An efficient approach for assessing hyperparameter importance. In *Proceedings of the 31st International Conference on International Conference on Machine Learning - Volume 32*, ICML'14, pages I–754–I–762, 2014.

[24] D. R. Jones, C. D. Perttunen, and B. E. Stuckman. Lipschitzian optimization without the Lipschitz constant. *Journal of Optimization Theory and Applications*, 79(1):157–181, 1993.

[25] K. Kandasamy, J. Schneider, and B. Póczos. High dimensional bayesian optimisation and bandits via additive models. In *Proceedings of the 32nd International Conference on International Conference on Machine Learning - Volume 37*, ICML'15, pages 295–304, 2015.

[26] C. G. Knight, S. H. E. Knight, N. Massey, T. Aina, C. Christensen, D. J. Frame, J. A. Kettleborough, A. Martin, S. Pascoe, B. Sanderson, D. A. Stainforth, and M. R. Allen. Association of parameter, software, and hardware variation with large-scale behavior across 57,000 climate models. *Proceedings of the National Academy of Sciences*, 104(30):12259–12264, 2007.

[27] P. M. Lee. *Bayesian Statistics: An Introduction*. John Wiley & Sons, 4th edition, 2012.

[28] C.-L. Li, K. Kandasamy, B. Poczos, and J. Schneider. High dimensional bayesian optimization via restricted projection pursuit models. In *Proceedings of the 19th International Conference on Artificial Intelligence and Statistics*, volume 51 of *Proceedings of Machine Learning Research*, pages 884–892, 2016.

[29] A. Nayebi, A. Munteanu, and M. Poloczek. A framework for Bayesian optimization in embedded subspaces. In *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, pages 4752–4761, 2019.

[30] A. Neumaier, O. Shcherbina, W. Huyer, and T. Vinkó. A comparison of complete global optimization solvers. *Mathematical Programming*, 103(2):335–356, 2005.

[31] E. Neuman. Inequalities and bounds for the incomplete gamma function. *Results in Mathematics*, 63(3): 1209–1214, 2013.

[32] NIST/SEMATECH. e-handbook of statistical methods, 2018. Available at https://www.itl.nist.gov/div898/handbook/eda/section3/eda3666.htm.

[33] H. Qian and Y. Yu. Solving high-dimensional multi-objective optimization problems with low effective dimensions. In *Proceedings of the Thirty-First AAAI Conference on Artificial Intelligence*, AAAI'17, pages 875–881, 2017.

[34] H. Qian, Y.-Q. Hu, and Y. Yu. Derivative-free optimization of high-dimensional non-convex functions by sequential random embeddings. In *Proceedings of the Twenty-Fifth International Joint Conference on Artificial Intelligence*, IJCAI'16, pages 1946–1952, 2016.

[35] P. Rolland, J. Scarlett, I. Bogunovic, and V. Cevher. High-Dimensional Bayesian Optimization via Additive Models with Overlapping Groups. *arXiv e-prints*, page arXiv:1802.07028, 2018.

[36] M. Rudelson and R. Vershynin. Smallest singular value of a random rectangular matrix. *Communications on Pure and Applied Mathematics*, 62(12):1707–1739, 2009.

[37] N. V. Sahinidis. *BARON 14.3.1: Global Optimization of Mixed-Integer Nonlinear Programs,* User's Manual, 2014.

[38] M. L. Sanyang and A. Kabán. Remeda: Random embedding eda for optimising functions with intrinsic dimension. In *Parallel Problem Solving from Nature – PPSN XIV*, pages 859–868, 2016.

[39] S. Surjanovic and D. Bingham. Virtual library of simulation experiments: Test functions and datasets, 2013. Available at https://www.sfu.ca/~ssurjano/.

[40] M. Tawarmalani and N. V. Sahinidis. A polyhedral branch-and-cut approach to global optimization. *Mathematical Programming*, 103:225–249, 2005.

[41] H. Tyagi and V. Cevher. Learning non-parametric basis independent models from point queries via low-rank methods. *Applied and Computational Harmonic Analysis*, 37(3):389–412, 2014.

[42] R. Vershynin. *High-Dimensional Probability: An Introduction with Applications in Data Science*. Cambridge Series in Statistical and Probabilistic Mathematics. 2018.

[43] Z. Wang, F. Hutter, M. Zoghi, D. Matheson, and N. De Freitas. Bayesian optimization in a billion dimensions via random embeddings. *Journal of Artificial Intelligence Research*, 55(1):361–387, 2016.

[44] Z. Wang, C. Gehring, P. Kohli, and S. Jegelka. Batched large-scale bayesian optimization in high-dimensional spaces. In *International Conference on Artificial Intelligence and Statistics (AISTATS)*, 2018.

[45] R. L. Wheeden. *Measure and integral : an introduction to real analysis*. Boca Raton: Chapman and Hall/CRC, 2nd edition, 2015.

A Technical definitions and results

A.1 Gaussian random matrices

Definition A.1 (Gaussian matrix). A Gaussian (random) matrix is a matrix whose each entry is an independent standard normal random variable.

Gaussian matrices have been well-studied with many results available at hand. Here, we mention a few key properties of Gaussian matrices that we use in the analysis; for a collection of results pertaining to Gaussian matrices and other related distributions refer to [21, 42].

Gaussian random matrices are known to be invariant with respect to orthogonal transformations:

Theorem A.2. *(see [21, Theorem 2.3.10]) Let $\boldsymbol{A}$ be an $D \times d$ Gaussian random matrix. If $\boldsymbol{U} \in \mathbb{R}^{D \times p}$, $D \geq p$, and $\boldsymbol{V} \in \mathbb{R}^{d \times q}$, $d \geq q$, are orthogonal, then $\boldsymbol{U}^T \boldsymbol{A} \boldsymbol{V}$ is a Gaussian random matrix.*

A related notion that plays an important role in the study of Gaussian matrices is the Wishart distribution represented by matrix $\boldsymbol{A}^T \boldsymbol{A}$ (or $\boldsymbol{A} \boldsymbol{A}^T$), where $\boldsymbol{A}$ is an overdetermined (underdetermined) Gaussian matrix. A Wishart matrix is positive definite with probability 1:

Theorem A.3. *(see [21, Theorem 3.2.1]) Let $\boldsymbol{A}$ be an $D \times d$ Gaussian random matrix, $D \geq d$. Then, the Wishart matrix $\boldsymbol{A}^T \boldsymbol{A}$ is positive definite with probability 1.*

The immediate consequence of the result is that the Wishart matrix is nonsingular with probability 1.

A.2 Chi-squared random variable

Definition A.4 (Chi-squared random variable). Given a collection $Z_1, Z_2, \ldots, Z_n$ of n independent standard normal variables, a random variable $X = Z_1^2 + Z_2^2 + \cdots Z_n^2$ is said to follow the chi-squared distribution with n degrees of freedom. We denote this by $X \sim \chi_n^2$.

The following lemma provides a notable relationship between the inverse of the Wishart matrix and the chi-squared random variable.

Lemma A.5. *(see [21, Corollary 3.3.13.1.]) Let $\boldsymbol{A}$ be an $D \times d$ Gaussian matrix, $D \geq d$, and let $\boldsymbol{z} \in \mathbb{R}^d$ be a fixed non-zero vector. Then,*

$$\frac{\|\boldsymbol{z}\|^2}{\boldsymbol{z}^T (\boldsymbol{A}^T \boldsymbol{A})^{-1} \boldsymbol{z}} \sim \chi_{D-d+1}^2.$$

In the following lemma we derive an upper bound for the cumulative density function (c.d.f.) of the chi-squared random variable.

Lemma A.6. *Let $X \sim \chi_n^2$. Then,*

$$\mathbb{P}[X \leq x] \leq \frac{4}{n(n+2)\Gamma(n/2)} \left(1 + \frac{n}{2} e^{-x/2}\right) (x/2)^{n/2}.$$

Proof. Recall the c.d.f. of the chi-square random variable (see, e.g., [32]):

$$\mathbb{P}[X \leq x] = \frac{\gamma(n/2, x/2)}{\Gamma(n/2)}$$

for $x > 0$, where $\gamma(n/2, x/2)$ is the lower incomplete gamma function (see [31]) defined as

$$\gamma(n/2, x/2) = \int_0^{x/2} u^{n/2-1} e^{-u} du.$$

We obtain the desired result by applying the following upper bound on $\gamma(n/2, x/2)$ (see [31, Theorem 4.1]),

$$\gamma(n/2, x/2) \leq \frac{4}{n(n+2)} \left(1 + \frac{n}{2} e^{-x/2}\right) (x/2)^{n/2}.$$

$\square$

A.3 The inverse chi-squared random variable

Definition A.7 (Inverse chi-squared random variable). Given $X \sim \chi_n^2$, a random variable $Y = 1/X$ is said to follow the inverse chi-squared distribution with n degrees of freedom. We denote this by $Y \sim 1/\chi_n^2$ (see [27, A5]).

Lemma A.8. *(see [27, A5]) Let $Y \sim 1/\chi_n^2$ and $W = sY$ for a positive real s. Then,*

$$\mathbb{E}[W] = \frac{s}{n-2}$$

provided that $n > 2$.

Lemma A.9. *Let Y and R be two random variables such that $Y \sim 1/\chi_n^2$ and $R = \sqrt{Y}$. Then, the probability density function (p.d.f.) $g(\hat{r})$ of R is given by*

$$g(\hat{r}) = \frac{2^{-n/2+1}}{\Gamma(n/2)}\hat{r}^{-n-1}e^{-1/(2\hat{r}^2)}.$$

Proof. The p.d.f. $g(\hat{r})$ of R satisfies

$$g(\hat{r}) = \frac{d}{d\hat{r}}\,\mathbb{P}[\sqrt{Y} < \hat{r}] = \frac{d}{d\hat{r}}\,\mathbb{P}[Y < \hat{r}^2] = 2\hat{r}h(n,\hat{r}^2), \tag{A.1}$$

where $h(n,\cdot)$ denotes the p.d.f. of the inverse chi-squared random variable Y given by (see, e.g., [27, A5])

$$h(n,y) = \frac{1}{2^{n/2}\Gamma(n/2)}y^{-n/2-1}e^{-1/(2y)} \text{ for } y > 0.$$

$\square$

A.4 Spherically distributed random vectors

Definition A.10. An $D \times 1$ random vector $\boldsymbol{x}$ is said to have a spherical distribution if for every orthogonal $D \times D$ matrix $\boldsymbol{U}$,

$$\boldsymbol{U}\boldsymbol{x} \stackrel{law}{=} \boldsymbol{x}.$$

Below are some useful facts about symmetrically distributed random vectors.

Lemma A.11. *[15, p. 13] Let $\boldsymbol{x}$ and $\boldsymbol{y}$ be random vectors such that $\boldsymbol{x} \stackrel{law}{=} \boldsymbol{y}$ and let $f_i(\cdot)$, $i = 1, 2, \ldots, m$, be measurable functions. Then,*

$$\begin{pmatrix} f_1(\boldsymbol{x}) & f_2(\boldsymbol{x}) & \cdots & f_m(\boldsymbol{x}) \end{pmatrix}^T \stackrel{law}{=} \begin{pmatrix} f_1(\boldsymbol{y}) & f_2(\boldsymbol{y}) & \cdots & f_m(\boldsymbol{y}) \end{pmatrix}^T.$$

Lemma A.12. *([15, Corollary, p. 30]) If $D \times 1$ random vector $\boldsymbol{x}$ has a spherical distribution, then*

$$\boldsymbol{x} \stackrel{law}{=} r\boldsymbol{u},$$

where $\boldsymbol{u}$ is distributed uniformly on the unit sphere S^D and r is a univariate random variable independent of $\boldsymbol{u}$.

Theorem A.13. *([15, Theorem 2.3]) Let $\boldsymbol{x} \stackrel{law}{=} r\boldsymbol{u}$ be a spherically distributed $D \times 1$ random vector with $\mathbb{P}[\boldsymbol{x} = \boldsymbol{0}] = 0$. Then,*

$$\|\boldsymbol{x}\| \stackrel{law}{=} r \text{ and } \|\boldsymbol{x}\|^{-1}\boldsymbol{x} \stackrel{law}{=} \boldsymbol{u},$$

Moreover, $\|\boldsymbol{x}\|$ and $\|\boldsymbol{x}\|^{-1}\boldsymbol{x}$ are independent.

Theorem A.14. *(see [22, Theprem 2.1.].) Let $\boldsymbol{x} \overset{law}{=} r\boldsymbol{u}$ be a spherically distributed $D \times 1$ random vector with $\mathbb{P}[\boldsymbol{x} = \boldsymbol{0}] = 0$, where r is independent of $\boldsymbol{u}$ with p.d.f. $h(\cdot)$. Then, p.d.f. $g(\hat{\boldsymbol{x}})$ of $\boldsymbol{x}$ is given by*

$$g(\hat{\boldsymbol{x}}) = \frac{\Gamma(n/2)}{2\pi^{D/2}} h(\|\hat{\boldsymbol{x}}\|) \|\hat{\boldsymbol{x}}\|^{1-D}.$$

For more details regarding spherical distributions refer to [15, 21, 3].

A.5 The least Euclidean norm solution to the random linear system

The present section establishes key properties of the least Euclidean norm solution to the underdetermined random linear system.

(C) Let $\bar{\boldsymbol{B}}$ be a $d_e \times d$ Gaussian matrix, where $d_e \leq d$, and let $\boldsymbol{z} \in \mathbb{R}^{d_e}$ be a fixed nonzero vector. Denote by $\boldsymbol{y}_2$ the least 2-norm solution to $\bar{\boldsymbol{B}}\boldsymbol{y} = \boldsymbol{z}$.

Lemma A.15. *Given (C), $\boldsymbol{y}_2$ satisfies*

$$\frac{\|\boldsymbol{z}\|_2^2}{\|\boldsymbol{y}_2\|_2^2} \sim \chi_{d-d_e+1}^2.$$

Proof. The least Euclidean norm solution $\boldsymbol{y}_2$ to $\bar{\boldsymbol{B}}\boldsymbol{y} = \boldsymbol{z}$ is given by

$$\boldsymbol{y}_2 = \bar{\boldsymbol{B}}^T (\bar{\boldsymbol{B}}\bar{\boldsymbol{B}}^T)^{-1} \boldsymbol{z}.$$

For its Euclidean norm, we have

$$\|\boldsymbol{y}_2\|_2^2 = (\bar{\boldsymbol{B}}^T (\bar{\boldsymbol{B}}\bar{\boldsymbol{B}}^T)^{-1} \boldsymbol{z})^T \bar{\boldsymbol{B}}^T (\bar{\boldsymbol{B}}\bar{\boldsymbol{B}}^T)^{-1} \boldsymbol{z}$$
$$= \boldsymbol{z}^T (\bar{\boldsymbol{B}}\bar{\boldsymbol{B}}^T)^{-1} \boldsymbol{z},$$

Using Lemma A.5 we obtain the desired result:

$$\frac{\|\boldsymbol{z}\|_2^2}{\|\boldsymbol{y}_2\|_2^2} = \frac{\|\boldsymbol{z}\|^2}{\boldsymbol{z}^T (\bar{\boldsymbol{B}}\bar{\boldsymbol{B}}^T)^{-1} \boldsymbol{z}} \sim \chi_{d-d_e+1}^2.$$

$\square$

Lemma A.16. *Given (C), $\boldsymbol{y}_2$ follows a spherical distribution.*

Proof. Let $\boldsymbol{S}$ be any $d \times d$ orthogonal matrix. Let $f : \mathbb{R}^{d_e d \times 1} \to \mathbb{R}^{d \times 1}$ be a vector-valued function defined as

$$f(\text{vec}(\bar{\boldsymbol{B}})) = \bar{\boldsymbol{B}}^T (\bar{\boldsymbol{B}}\bar{\boldsymbol{B}}^T)^{-1} \boldsymbol{z},$$

where $\text{vec}(\bar{\boldsymbol{B}})$ denotes the $Dd \times 1$ vector $(\bar{\boldsymbol{b}}_1^T \ \bar{\boldsymbol{b}}_2^T \ \cdots \bar{\boldsymbol{b}}_d^T)^T$ with $\bar{\boldsymbol{b}}_i$ being the ith column vector of $\bar{\boldsymbol{B}}$. Using the fact that the inverse of a matrix is equal to the ratio of its adjugate to its determinant we can express f as

$$f(\text{vec}(\bar{\boldsymbol{B}})) = \begin{pmatrix} \frac{p_1(\bar{\boldsymbol{B}})}{q(\bar{\boldsymbol{B}})} & \frac{p_2(\bar{\boldsymbol{B}})}{q(\bar{\boldsymbol{B}})} & \cdots & \frac{p_d(\bar{\boldsymbol{B}})}{q(\bar{\boldsymbol{B}})} \end{pmatrix}^T,$$

where $p_i(\bar{\boldsymbol{B}})$ for $1 \leq i \leq d$ are some polynomials of the entries of $\bar{\boldsymbol{B}}$ and $q(\bar{\boldsymbol{B}})$ is the determinant of $\bar{\boldsymbol{B}}\bar{\boldsymbol{B}}^T$.

We first would like to prove that f is a measurable function. Recall that a function is measurable if and only if each of its components is measurable. It is enough to show that p_1/q is measurable; the same argument will apply to the rest of its components. First, we note that

(i) p_1 and q are measurable;

(ii) q is non-zero almost everywhere.

To prove (i), observe that the polynomials p_1 and q are sums of scalar multiples of products of standard normal random variables, which by definition are measurable. Sums, scalar multiples and products of measurable functions are measurable; hence, p_1 and q must be measurable. To prove (ii), we refer to Theorem A.3, which says that the matrix $\bar{\boldsymbol{B}}\bar{\boldsymbol{B}}^T$ is positive definite with probability 1 implying that all of its eigenvalues are strictly positive with probability 1. Then, (ii) follows from the fact that the determinant of the symmetric square matrix is equal to the product of its eigenvalues. Now, we can apply [45, Theorem 4.10] to deduce that p_1/q is measurable; this completes the proof that f is measurable.

For $\boldsymbol{y}_2 = \bar{\boldsymbol{B}}^T(\bar{\boldsymbol{B}}\bar{\boldsymbol{B}}^T)^{-1}\boldsymbol{z}$, we have

$$\boldsymbol{y}_2 = f(\mathrm{vec}(\bar{\boldsymbol{B}})) \text{ and } \boldsymbol{S}\boldsymbol{y}_2 = f(\mathrm{vec}(\bar{\boldsymbol{B}}\boldsymbol{S}^T)).$$

According to Theorem A.2, $\mathrm{vec}(\bar{\boldsymbol{B}}) \overset{law}{=} \mathrm{vec}(\bar{\boldsymbol{B}}\boldsymbol{S}^T)$. Then, by applying Lemma A.11, we obtain

$$\boldsymbol{y}_2 \overset{law}{=} \boldsymbol{S}\boldsymbol{y}_2.$$

Hence, $\boldsymbol{y}_2$ follows a spherical distribution by Definition A.10. $\qquad\square$

Lemma A.17. *Given (C), the probability density function of $\boldsymbol{y}_2$ is given by*

$$g(\hat{\boldsymbol{y}}) = \pi^{-d/2}\left[\frac{\Gamma(d/2)}{\Gamma(n/2)}\right]\left(\frac{\|\boldsymbol{z}\|}{\sqrt{2}}\right)^n (\hat{\boldsymbol{y}}^T\hat{\boldsymbol{y}})^{-(n+d)/2}e^{-\|\boldsymbol{z}\|^2/(2\hat{\boldsymbol{y}}^T\hat{\boldsymbol{y}})},$$

where $n = d - d_e + 1$.

Proof. The fact that $\boldsymbol{y}_2$ has a spherical distribution is a key ingredient in the proof. To simplify the derivations, let us assume for now that $\|\boldsymbol{z}\| = 1$.

By Lemma A.12,
$$\boldsymbol{y}_2 \overset{law}{=} r\boldsymbol{u},$$

where r is a univariate random variable and where $\boldsymbol{u}$ is a random vector distributed uniformly on S^d; moreover, r and $\boldsymbol{u}$ are independent.

Our first goal is to show that r and $\|\boldsymbol{y}_2\|$ have the same distribution. This fact follows immediately from Theorem A.13 if we show that $\mathbb{P}[\boldsymbol{y}_2 = \boldsymbol{0}] = 0$. Let $W \sim 1/\chi^2_{d-d_e+1}$. According to Lemma A.15, we have
$$\|\boldsymbol{y}_2\|^2 \overset{law}{=} W, \tag{A.2}$$

which we use in the second equation below

$$\mathbb{P}[\boldsymbol{y}_2 = \boldsymbol{0}] = \mathbb{P}[\|\boldsymbol{y}_2\|^2 = 0] = \mathbb{P}[W = 0].$$

Since W is a continuous random variable, $\mathbb{P}[W = 0] = 0$. This proves that

$$r \overset{law}{=} \|\boldsymbol{y}_2\|. \tag{A.3}$$

Combining (A.2) and (A.3), we conclude that r has the same distribution as $W^{1/2}$. The probability density function of $W^{1/2}$ — and, consequently, of r — derived in Lemma A.9 is given by

$$h(\hat{r}) = \frac{2^{-n/2+1}}{\Gamma(n/2)}\hat{r}^{-n-1}e^{-1/(2\hat{r}^2)}. \tag{A.4}$$

Theorem A.14 allows us to express the p.d.f. of $\boldsymbol{y}_2$ in terms of the p.d.f. of r:

$$g(\hat{\boldsymbol{y}}) = \frac{\Gamma(d/2)}{2\pi^{d/2}}(\hat{\boldsymbol{y}}^T\hat{\boldsymbol{y}})^{(1-d)/2}h\big(\sqrt{\hat{\boldsymbol{y}}^T\hat{\boldsymbol{y}}}\big).$$

By using (A.4) for $h(\cdot)$ in the above, we obtain

$$g(\hat{\boldsymbol{y}}) = 2^{-n/2}\pi^{-d/2}\frac{\Gamma(d/2)}{\Gamma(n/2)}(\hat{\boldsymbol{y}}^T\hat{\boldsymbol{y}})^{-(n+d)/2}e^{-1/(2\hat{\boldsymbol{y}}^T\hat{\boldsymbol{y}})}. \qquad (A.5)$$

To derive the p.d.f. for arbitrary non-zero $\boldsymbol{z}$, we consider the linear transformation $\bar{\boldsymbol{y}} = \|\boldsymbol{z}\|\hat{\boldsymbol{y}}$. The Jacobian of the transformation is equal to $1/\|\boldsymbol{z}\|^d$. Thus, the p.d.f. $\bar{g}(\bar{\boldsymbol{y}})$ of $\bar{\boldsymbol{y}}$ satisfies

$$\bar{g}(\bar{\boldsymbol{y}}) = \frac{g(\bar{\boldsymbol{y}}/\|\boldsymbol{z}\|)}{\|\boldsymbol{z}\|^d},$$

which together with (A.5) yields the desired result. $\qquad\square$

B Problem set

Table 3 contains the explicit formula, domain and global minimum of the functions used to generate the high-dimensional test set. The problem set contains 19 problems taken from [20, 14, 39]. Problems that cannot be solved by BARON are marked with '*'. Problems that will not be solved by KNITRO are marked with '°'.

Table 3: The problem set listed in alphabetical order.

| Function	Domain	Global minima
1) Beale [14]	$\boldsymbol{x} \in [-4.5, 4.5]^2$	$g(\boldsymbol{x}^*) = 0$
2) Branin* [14]	$x_1 \in [-5, 10]$ $x_2 \in [0, 15]$	$g(\boldsymbol{x}^*) = 0.397887$
3) Brent [20]	$\boldsymbol{x} \in [-10, 10]^2$	$g(\boldsymbol{x}^*) = 0$
4) °Bukin N.6 [39]	$x_1 \in [-15, -5]$ $x_2 \in [-3, 3]$	$g(\boldsymbol{x}^*) = 0$
5) *Easom [14]	$\boldsymbol{x} \in [-100, 100]^2$	$g(\boldsymbol{x}^*) = -1$
6) Goldstein-Price [14]	$\boldsymbol{x} \in [-2, 2]^2$	$g(\boldsymbol{x}^*) = 3$
7) Hartmann 3 [14]	$\boldsymbol{x} \in [0, 1]^3$	$g(\boldsymbol{x}^*) = -3.86278$
8) Hartmann 6 [14]	$\boldsymbol{x} \in [0, 1]^6$	$g(\boldsymbol{x}^*) = -3.32237$
9) *Levy [39]	$\boldsymbol{x} \in [-10, 10]^4$	$g(\boldsymbol{x}^*) = 0$
10) Perm 4, 0.5 [39]	$\boldsymbol{x} \in [-4, 4]^4$	$g(\boldsymbol{x}^*) = 0$
11) Rosenbrock [39]	$\boldsymbol{x} \in [-5, 10]^3$	$g(\boldsymbol{x}^*) = 0$
12) Shekel 5 [39]	$\boldsymbol{x} \in [0, 10]^4$	$g(\boldsymbol{x}^*) = -10.1532$
13) Shekel 7 [39]	$\boldsymbol{x} \in [0, 10]^4$	$g(\boldsymbol{x}^*) = -10.4029$
14) Shekel 10 [39]	$\boldsymbol{x} \in [0, 10]^4$	$g(\boldsymbol{x}^*) = -10.5364$
15) *Shubert [39]	$\boldsymbol{x} \in [-10, 10]^2$	$g(\boldsymbol{x}^*) = -186.7309$
16) Six-hump camel [39]	$x_1 \in [-3, 3]$ $x_2 \in [-2, 2]$	$g(\boldsymbol{x}^*) = -1.0316$
17) Styblinski-Tang [39]	$\boldsymbol{x} \in [-5, 5]^4$	$g(\boldsymbol{x}^*) = -156.664$
18) Trid [39]	$\boldsymbol{x} \in [-25, 25]^5$	$g(\boldsymbol{x}^*) = -30$
19) Zettl [14]	$\boldsymbol{x} \in [-5, 5]^2$	$g(\boldsymbol{x}^*) = -0.00379$

C   Additional experiments

We conducted three more experiments to test REGO's robustness to changes in the parameters. In all three experiments, the same budget and termination criteria as in the main experiment are used.

(A) In this experiment, we assume that no good estimate for μ is known and that μ can be as large as $\sqrt{D}$ (for example, when $\mathcal{X} = [-1,1]^D$ constraint is imposed). We test REGO with the following parameters: $(d_e, 8.0 \times \sqrt{D}), (d_e + 1, 2.2 \times \sqrt{D}), (d_e + 2, 1.3 \times \sqrt{D})$ and $(d_e + 3, 1.0 \times \sqrt{D})$. Results are presented in Figure 8 in Appendix C.

(B) We fix δ to be $7.5\sqrt{d_e}$ and vary d. The following parameters are used: $(d_e, 7.5 \times \sqrt{d_e}), (d_e + 1, 7.5 \times \sqrt{d_e}), (d_e + 2, 7.5 \times \sqrt{d_e})$ and $(d_e + 3, 7.5 \times \sqrt{d_e})$. Results are presented in Figure 9 in Appendix C.

(C) We fix $d = d_e + 1$ and vary δ $(= 5\sqrt{d_e}, 7.5\sqrt{d_e}, 10\sqrt{d_e})$. The following parameters are used: $(d_e + 1, 5 \times \sqrt{d_e}), (d_e + 1, 7.5 \times \sqrt{d_e})$ and $(d_e + 1, 10 \times \sqrt{d_e})$. In the figures we also include curves for $\delta_{opt} = 2.2\sqrt{d_e}$ taken from the main experiment. Results are presented in Figure 10 in Appendix C.

Conclusions

(A) We test robustness of REGO assuming that μ is equal to $\sqrt{D}$ (which makes δ to be relatively large and dependent on D). Despite this dependence, the frequency of convergence for BARON and KNITRO is high showing mild dependence on D.

(B) The purpose of this experiment is to see how different values of d affect the performance of REGO while δ is kept constant. For larger d, we expect (RP) to be successful with higher chance. Nonetheless, the results show that sometimes, for larger d, REGO's performance may be compromised; this is for example true for BARON's convergence$_{opt}$. Since δ is set to a relatively large value, (RP) is successful with high probability even for smallest d. This suggests that as long as d and δ produce relatively high chance of success of (RP), one should stop increasing their values lest convergence to the global minimum require larger computational resources.

(C) In this experiment, we apply REGO with different values of δ while keeping d constant. The results display no significant differences between the performances with different parameters. Even the results with the optimal δ (used in the main experiment) do not differ considerably from the one with the largest δ except for BARON where the former wins in terms of convergence$_{opt}$ and CPU time. The results of this experiment together with the results in (B) indicate that it is better to increase δ and keep d constant if one wants to increase success of (RP) with minimal increase in computational cost.
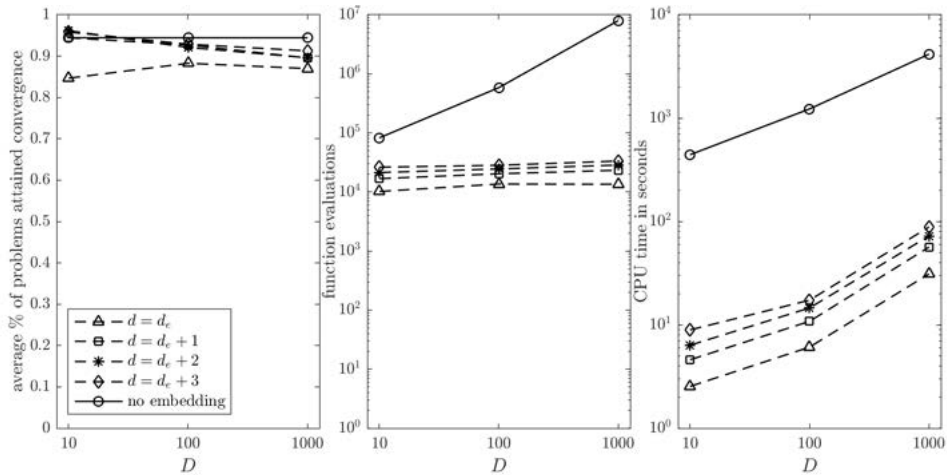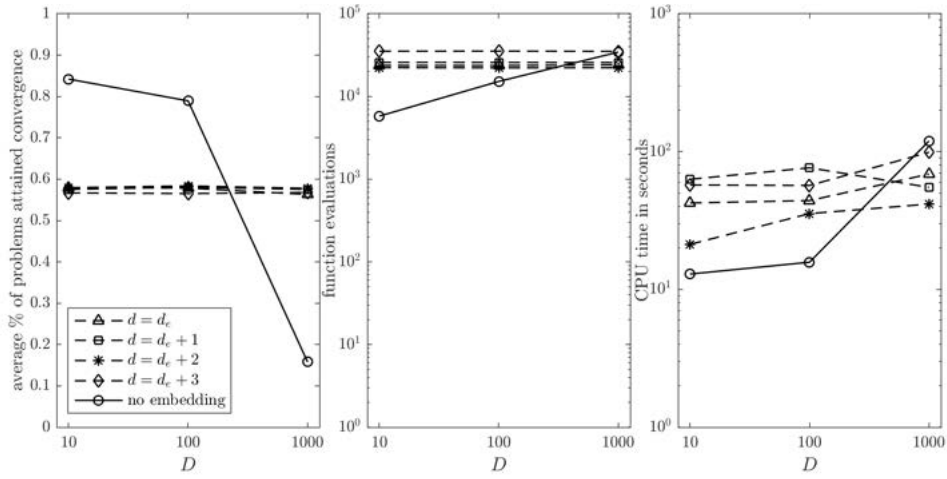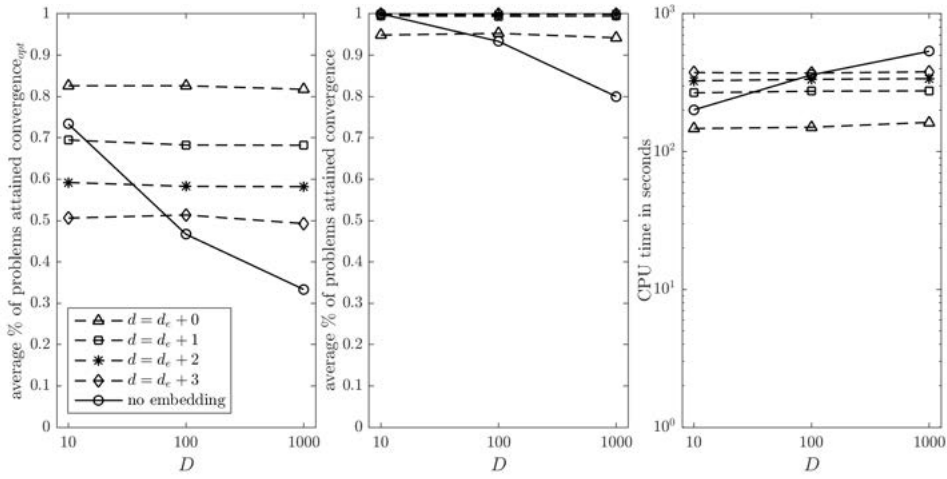
a) DIRECT
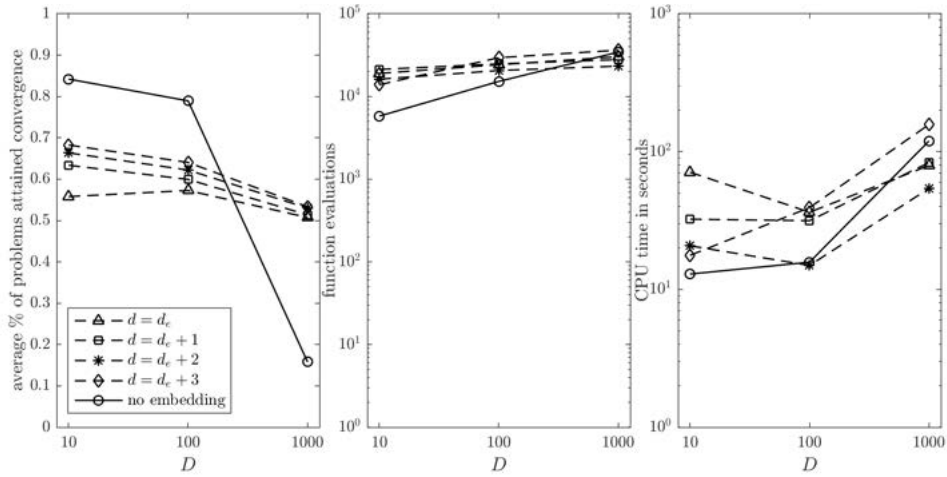
b) BARON
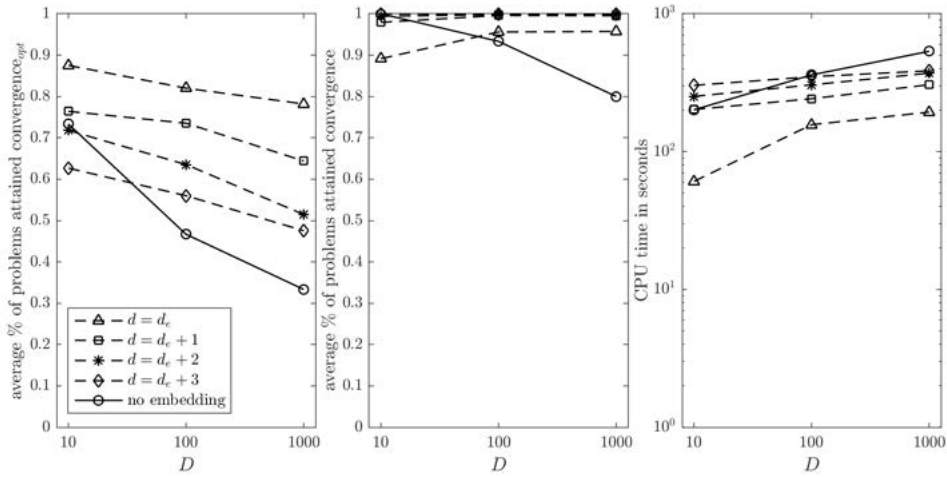
c) KNITRO

Figure 8: Experiment A: REGO with DIRECT, BARON and KNITRO with $(d, \delta) = (d_e, 8.0 \times \sqrt{D}), (d_e + 1, 2.2 \times \sqrt{D}), (d_e + 2, 1.3 \times \sqrt{D})$ and $(d_e + 3, 1.0 \times \sqrt{D})$.
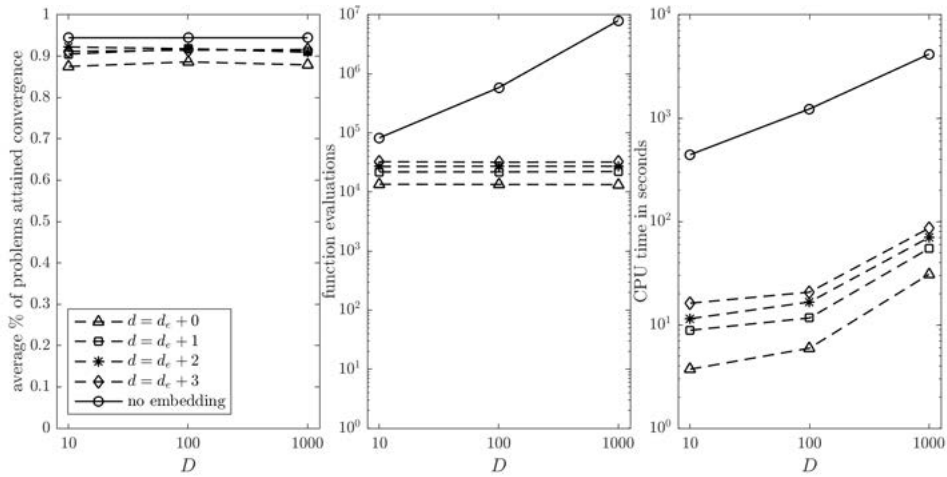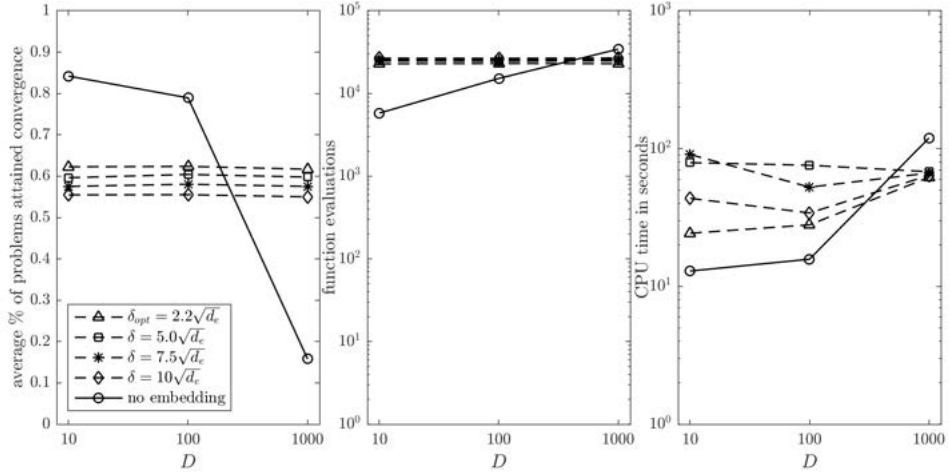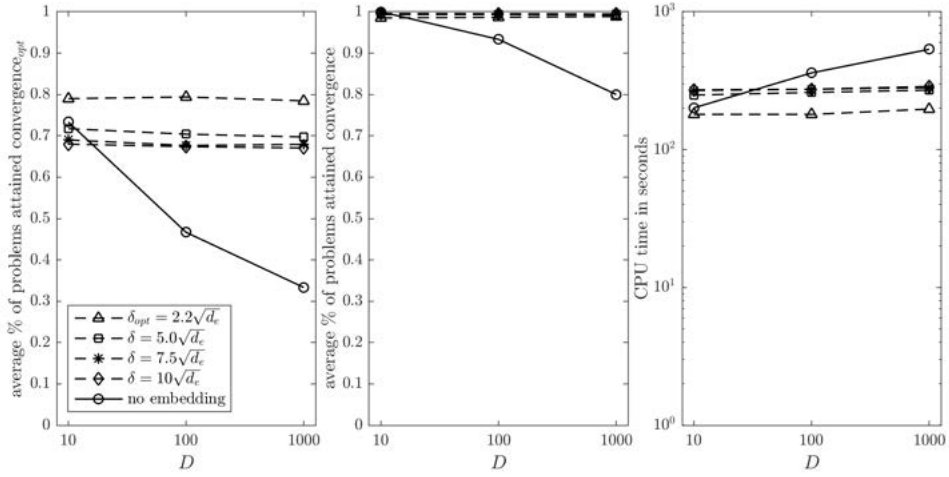
30

a) DIRECT

b) BARON

c) KNITRO

Figure 9: Experiment B: REGO with DIRECT, BARON and KNITRO with $\delta = 7.5\sqrt{d_e}$ fixed and $d = d_e$, $d_e + 1$, $d_e + 2$ and $d_e + 3$.

a) DIRECT
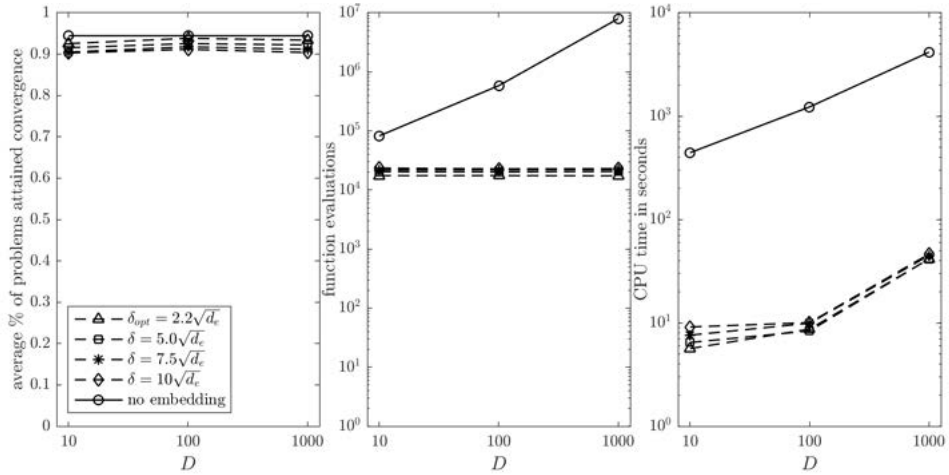

b) BARON

c) KNITRO

Figure 10: Experiment C: REGO with DIRECT, BARON and KNITRO with $d = d_e + 1$ fixed and $\delta = 5\sqrt{d_e}, 7.5\sqrt{d_e}, 10\sqrt{d_e}$ and $2.2\sqrt{d_e}$ (δ_{opt}).