

A Story of Three Levels of Sophistication in SCF/KS-DFT Orbital Optimization Procedures

Daniel Sethio,^{†,‡} Emily Azzopardi,[†] Ignacio Fdez. Galván,^{*,†} and Roland
Lindh^{*,†,¶}

[†]*Department of Chemistry – BMC, Uppsala University, P. O. Box 576, SE-75123 Uppsala,
Sweden*

[‡]*Department of Chemistry – Ångström, Uppsala University, P. O. Box 538, SE-75121
Uppsala, Sweden*

[¶]*Uppsala Center for Computational Chemistry (UC₃), Uppsala University, PO Box 576,
SE-751 23 Uppsala. Sweden*

E-mail: Ignacio.Fernandez@kemi.uu.se; roland.lindh@kemi.uu.se

Abstract

In this report, three versions of SCF/KS-DFT orbital optimization are described and benchmarked. The methods are a modified version of the geometry version of the direct inversion in the iterative subspace approach (which we call r-GDIIS), the modified restricted step rational function optimization method (RS-RFO), and the novel subspace gradient enhanced Kriging method, combined with restricted variance optimization (S-GEK/RVO). The modifications introduced are aimed to improve the robustness and computational scaling of the procedures. In particular, the subspace approach in S-GEK/RVO allows the application to SCF/KS-DFT optimization of a machine technique that has proved successful in geometry optimizations. The performance of the three methods is benchmarked for a large number of small to medium-sized organic molecules, at equilibrium structures and close to a transition state, and a second set of molecules containing closed- and open-shell transition metals. The results indicate the importance of the resetting technique in boosting the performance of the r-GDIIS procedure. Moreover, it is demonstrated that already at the inception of the subspace version of GEK to optimize SCF wave functions, it displays superior and robust convergence properties as compared to standard state-of-the-art SCF/KS-DFT optimization methods.

Introduction

The optimization of the orbitals in computer implementations of the Hartree–Fock–Roothaan procedure¹ has been a central issue in computational chemistry for the last 70 years. A similar procedural approach is also at the core of the optimization of the non-interacting orbital in the the Kohn–Sham approach.² A plethora of reports has been published on suggested improvements – both with respect to efficiency and convergence robustness. Early improvements include a third-order truncated Taylor expansion of the energy parameterized in terms of unitary rotations as suggested by Yaffe and Goddard³, the quadratically convergent SCF (QC-SCF) procedure of Bacskay⁴, and the use of the direct inversion in the iterative subspace (DIIS) method as designed by Pulay^{5,6}.

More recently, a number of variations of the DIIS method have been suggested⁷⁻¹⁴. The augmented Roothaan–Hall approach is an alternative approach, sometimes used in conjunction with the DIIS approach¹⁵⁻¹⁹. A number of restricted-step second-order methods have also been presented and implemented²⁰⁻²⁵. Alternatively, new hybrid methods have been devised to utilize the pros of second order methods and the DIIS approach^{26,27}. Overall these methods generally provide satisfactory convergence statistics. However, as the electronic structure of the target molecule starts to exhibit near-degeneracy effects and open-shell characteristics, it is not uncommon to experience poor or no convergence at all. Further developments of SCF/KS-DFT orbital optimizations procedures are needed to address these issues.

What all of the approaches mentioned above have in common is the use of a surrogate model which is based on a truncated Taylor expansion – usually curtailed after the second-order term. Variations can depend on the parameterization of the energy expression in terms of molecular orbital coefficients, elements of matrices representing unitary rotations of reference orbitals, or one-particle density matrices. Additional contrasts between suggested schemes are due to ad-hoc procedures for trust region and trust radius implementations. These procedures are in general efficient and stable, however, plenty of examples of slow or non-existent convergence are observed in situations where alternative solutions to the Fock equations are closely packed in the parameter space around the desired stationary point.

The aim of this paper is to offer improvements of existing optimization techniques and to introduce new tools in the chase of an efficient and robust orbital optimization procedure. In the latter case the techniques from machine learning (ML) will be explored. Rather than to fit data (orbital coefficients and associated one-particle density- and Fock-matrices) to a fixed surrogate model, a flexible surrogate model will be fitted to the data with no loss of information. In particular, the study will investigate the use of a non-parametric regression approach using Gaussian process regression (GPR) for the optimization of SCF and KS-DFT orbitals. The hypothesis is that such an approach is superior to standard and modified (quasi-)Newton optimization procedures. To explicitly challenge this hypothesis, the novel

approach will be benchmarked against two updated standard methods – the GDIIS^{28,29} and the RS-RFO approach, the former with a new resetting approach (r-GDIIS) and the latter as adapted to SCF optimization. The new ML-based optimization procedure will be built on a version of the restricted variance optimization (RVO) procedure³⁰ specially adapted to orbital optimizations and the large parameter spaces in such procedures, the subspace version of gradient-enhanced Kriging³¹⁻³³ (S-GEK) approach.

The rest of the paper will be structured as follows. A first section will be devoted to the theory and improvements of the GDIIS, RS-RFO and the new adapted S-GEK approach. Another section follows in which an adequate benchmark suite is designed to effectively test the hypothesis. The results are discussed in the following section, in which the acquired benchmark data is presented and critically analyzed. Finally, the report ends with some conclusions and perspectives. As additional material, an appendix is included to describe the startup procedure used in the calculations.

Theory

This section will describe three procedures of SCF/KS-DFT orbital optimizations in some detail – the direct inversion iterative subspace, the restricted-step rational function optimization, and the subspace gradient-enhanced Kriging model in association with restricted-variance optimization. All of them, however, will have in common the parameterization of the SCF wave function or the determinant describing the KS-DFT non-interacting orbitals, which will be briefly described here before the presentation of the different optimization procedures.

Rather than directly minimizing the SCF/KS-DFT energy with respect to the molecular orbital coefficients, the presented methods will start from a set of orthonormal orbitals – some being occupied and some virtual. The optimization procedure will subsequently determine the optimal occupied orbitals via an unitary rotation of the orbitals. It is here noted that the SCF/KS-DFT energy is invariant to unitary rotations of the occupied orbitals among each other. Moreover, the SCF/KS-DFT energy is also invariant to the virtual

orbitals – they just span a complementary space to the one spanned by the occupied orbitals. Henceforth $\{i, j, k\}$, $\{a, b, c\}$, and $\{p, q, r\}$, respectively, will denote occupied, virtual and general orbital indices. It is noted that the optimization procedure focuses exclusively on the rotations between the occupied and virtual subspaces of the SCF/KS-DFT orbitals. Following the description suggested by Jørgensen and co-workers^{34,35}, this parameterization starts with a reference function, a Slater determinant, $|\Psi_0\rangle$, defining the number of electrons and the occupied orbitals,

$$|\Psi_0\rangle = \prod_{i \in 0} a_i^\dagger |\text{vac}\rangle \quad (1)$$

where a_i^\dagger (a_i) is a creation (annihilation) operator creating orbital i , and $\prod_{i \in 0} a_i^\dagger$ is an ordered product of such creation operators acting on the vacuum state, $|\text{vac}\rangle$.

A unitary transformation (restricting it to real rotations) of the orbitals making up the SCF/KS-DFT Slater determinant is subsequently described as

$$|\Psi(\hat{\kappa})\rangle = \exp(-\hat{\kappa}) |\Psi_0\rangle \quad (2)$$

where

$$\hat{\kappa} = \sum_{rs} \kappa_{rs} (a_r^\dagger a_s - a_s^\dagger a_r) \quad (3)$$

where κ is an antisymmetric matrix and $\exp(-\kappa)$ is a unitary matrix.

This renders the SCF/KS-DFT expectation energy to be expressed as

$$E(\kappa) = \langle \Psi_0 | \exp(-\hat{\kappa}) \hat{H} \exp(-\hat{\kappa}) | \Psi_0 \rangle \quad (4)$$

where \hat{H} is the Hamiltonian operator. For the purpose of facilitating the orbital optimization the energy is expressed as a Taylor expansion²⁷ (here explicitly expressed up to second order)

$$E(\kappa) = E(\mathbf{0}) + \mathbf{g}^\top \kappa + \frac{1}{2} \kappa^\top \mathbf{H} \kappa + \dots \quad (5)$$

where the elements of the gradient vector, \mathbf{g} , are expressed as

$$g_{ia} = \frac{\partial E}{\partial \kappa_{ia}} = 2F_{ia} \quad (6)$$

that is, the matrix elements are, with the exception of a factor of 2, identical to the occupied–virtual elements of the Fock matrix in molecular orbital (MO) basis (not necessarily the canonical orbitals). Finally, for the elements of the Hessian matrix, \mathbf{H} (not to be confused with the Hamiltonian), in a quasi-Newton procedure only a reasonable approximation is used

$$H_{ai,bj} \approx 2F_{ab}\delta_{ij} - 2F_{ij}\delta_{ab} \quad (7)$$

in association with some Hessian update methods.

For practical purposes the occupied–occupied and virtual–virtual subblocks of κ are set to zero. Hence, the parameter space of the SCF/KS-DFT orbitals is completely defined by the occupied–virtual or the virtual–occupied subblocks of κ . It is noted that the orbitals generated with the parameterization are not necessarily identical to the canonical SCF/KS-DFT orbitals. A post-convergence diagonalization of the occupied–occupied and the virtual–virtual subblocks of the Fock matrix would, however, generate the canonical occupied and virtual orbitals, respectively.

Direct Inversion in the Iterative Subspace

A number of procedures to optimize the orbitals of a Slater determinant was offered in the early days of the development of the self-consistent field approach of the Hartree–Fock single configuration wave function model. Most of these were either slow to converge or far from robust. This changed to the better in 1980 when Pulay proposed the use of the so-called direct inversion in the iterative subspace (DIIS) method,⁵ a combination of a variable metric update method and a minimization step as a tractable approach. Originally, this approach was introduced in terms of a parameterization using the occupied–virtual block of the Fock matrix, while here the developments are parallel to those suggested by Fischer and Almlöf²⁹. That is, the parameterization is in terms of the occupied–virtual orbital ro-

tations, as mentioned at the start of this section; the Hessian matrix is not explicitly stored, but it is implicitly updated as it is multiplied by a trial vector with an on-the-fly version of the BFGS^{36–39} (Broyden–Fletcher–Goldfarb–Shanno) method; and the GDIIS²⁸ approach is employed. The DIIS method has since its introduction been the subject of many publications and modifications. This report will not dwell into the details of these; however, it is interesting to note that recent publications⁴⁰ indicate that the DIIS method of Pulay actually is a special case of the Anderson mixing-procedure⁴¹ from 1965.

The GDIIS procedure (using this name to denote the DIIS variant described by Fischer and Almlöf²⁹) is a two-step procedure that assumes the presence of a set of n parameter vectors, $(\boldsymbol{\kappa}_i, i = 1, \dots, n)$, and the corresponding set of gradient vectors, \mathbf{g}_i , one for each iteration, i . First, for parameter sets in the quadratic region an approximated error vector, \mathbf{e}_i , can be compiled as

$$\mathbf{e}_i = -\mathbf{H}_n^{-1} \mathbf{g}_i \approx \boldsymbol{\kappa}_f - \boldsymbol{\kappa}_i = \Delta \boldsymbol{\kappa}_i \quad (8)$$

where \mathbf{H}_n is the updated approximate Hessian and $\boldsymbol{\kappa}_f$ is the optimal parameter set. The m last iterations – the iterative subspace – are presumed to be in this quadratic region and are used in the first step of the GDIIS procedure, m is the so-called depth of the DIIS procedure. An improved set of parameters can be computed as

$$\boldsymbol{\kappa}_{m+1}^* = \sum_{i=1}^m c_i \boldsymbol{\kappa}_i \quad (9)$$

by minimizing the norm of the extrapolated error vector

$$\Delta \boldsymbol{\kappa} = \sum_{i=1}^m c_i \mathbf{e}_i \approx \sum_{i=1}^m c_i \Delta \boldsymbol{\kappa}_i \quad (10)$$

under the condition that $\sum_{i=1}^m c_i = 1$. This is achieved by minimizing the following Lagrangian

$$L = \Delta \boldsymbol{\kappa}^\top \Delta \boldsymbol{\kappa} - 2\lambda \left(\sum_{i=1}^m c_i - 1 \right) \quad (11)$$

Here, the first term on the RHS is expressed as

$$\Delta\boldsymbol{\kappa}^\top\Delta\boldsymbol{\kappa} = \sum_{i,j} c_i c_j B_{ij} \quad (12)$$

where $B_{i,j} = \mathbf{e}_i^\top \mathbf{e}_j$ are element of the error matrix. Solving this minimization problem corresponds to finding the solution to the following set of equations

$$\begin{pmatrix} B_{11} & \cdots & B_{1m} & -1 \\ \vdots & \ddots & \vdots & \vdots \\ B_{m1} & \cdots & B_{mm} & -1 \\ -1 & \cdots & -1 & 0 \end{pmatrix} \begin{pmatrix} c_1 \\ \vdots \\ c_m \\ -\lambda \end{pmatrix} = \begin{pmatrix} 0 \\ \vdots \\ 0 \\ -1 \end{pmatrix} \quad (13)$$

which is of the type $\mathbf{B}^a \mathbf{x} = \mathbf{b}$, where \mathbf{B}^a is the augmented error matrix. This is trivially solved as $\mathbf{x} = (\mathbf{B}^a)^{-1} \mathbf{b}$. It is noted that as $\Delta\boldsymbol{\kappa}$ converges towards the zero vector, the parameter set for the next iteration will converge towards the optimal parameter set, $\boldsymbol{\kappa}_r$. The procedure now goes into the second step in which the convergence is checked by computing a second update to the parameter set, now in the full space of the parameter space, $\delta\boldsymbol{\kappa}_{n+1}$, as

$$\delta\boldsymbol{\kappa}_{n+1} = \boldsymbol{\kappa}_{n+1}^* - \boldsymbol{\kappa}_n - \mathbf{H}_n^{-1} \mathbf{g}_{n+1}^* \quad (14)$$

The actual gradient, $\mathbf{g}(\boldsymbol{\kappa}_{n+1})$, is computed for the new parameter set. If the norm of the gradient is below the convergence threshold, the procedure is considered converged. Otherwise the GDIIS procedure is repeated in a subsequent iteration, but now with optionally one more error vector.

Resetting Direct Inversion in the Iterative Subspace

Experience has demonstrated that solving the system of equations of Eq. (13) is prone to numerical instability. This issue can be addressed in at least three different ways. First, the linear dependence can be reduced/eliminated by setting m to a rather small constant integer value (in the current implementation $m_{\max} = 5$). As the GDIIS procedure iterates, error

vectors from earlier iterations are eliminated – thus reducing the risk of linear dependence. Second, as suggested by Chupin et al.⁴⁰, the new error vector is projected onto the manifold of the m previous error vectors, and if the norm of the remainder (a measure of the extent of linear dependence) is below a threshold, m is either reset to one, or incrementally reduced until the condition is no longer fulfilled. This approach is not used in the present benchmarking. Third, a reformulation of the problem is used, as suggested by Sellers⁴², which reduces the numerical problem – the C^2 -DIIS method. Here the optimization problem is cast as an eigenvalue problem

$$\mathbf{B}\mathbf{q}_j = \lambda_j\mathbf{q}_j \quad (15)$$

which implies a normalization of the eigenvectors according to $\sum_{i=1}^m (\mathbf{q}_j)_i^2 = 1$. The selection of the appropriate eigenvector starts by computing the coefficients of the original formulation of the DIIS procedure, from the renormalization $\mathbf{c}_j = \mathbf{q}_j/N_j$, where $N_j = \sum_i (\mathbf{q}_j)_i$, followed by the evaluation of the norm of the extrapolated error vector. In the original implementation the set of coefficients corresponding to the lowest eigenvalue above a specific threshold – to avoid solutions that are numerically suspect – is kept. However, in the present benchmark a modification of this selection procedure is introduced as follows. The norm of \mathbf{c}_j , is computed. A large number corresponds to the elements of the coefficient vector being a series of large numbers with varying sign – a signature of a solution corrupted by numerical instability, which is also reflected in a small eigenvalue. Hence, in the present implementation, if $\mathbf{c}_j^\top \mathbf{c}_j > \delta$ and $\mathbf{c}_j^\top \mathbf{B}\mathbf{c}_j < \sigma$, where $\delta = 100$ and $\sigma = 10^{-5}$, that particular solution is rejected. Otherwise, the solution corresponding to the extrapolated error vector with the lowest norm is selected.

This is not, however, the end of the considerations that need to be attended for optimal convergence. In what follows is a description of conditions which will reset either the DIIS depth or the BFGS update of the approximate Hessian. This will be coined the resetting DIIS approach, r-DIIS. It should be noted that the suggested procedure below has had its precursors in the restarted DIIS method (r-Pulay)¹² which on regular intervals resets the depth of the DIIS, and the method of Fang and Saad⁹ which resets the DIIS depth if the ratio of the residuals of the two last iterations exceeds a threshold. The suggested approach

here, however, contains three conditions which will trigger the resetting mechanism.

First, numerical stability is not the only issue with the DIIS approach, the anharmonic character of the PES is another problem – the norm of the gradient is not linear with respect to the distance from the stationary point. Thus, when the DIIS method is applied to error vectors whose norms differ by orders of magnitude, the predictive power of the approach is lost. In the current implementation this issue is handled by the condition

$$\delta B_{i,i} > B_{n,n}, \quad i = n - m + 1, \dots, n - 1 \quad (16)$$

where $\delta = 10^{-8}$. If the condition is fulfilled, the depth of the DIIS procedure, m , is decreased in steps of one until the condition is not fulfilled or $m = 1$.

Second, having solved these problems there could still be convergence issues due to the qualitative nature of the (G)DIIS procedure – in particular it is a procedure with no reference to the value of the energy. As the method is formulated it is based on the energy function being convex. That is, this strictly leads to that one expects that the gradient norm continuously decreases as the geometry approaches the stationary point. However, this is not always the case. Especially, it is not uncommon for the optimization procedure to evolve along the energy surface such that it is experiencing a shoulder (inflection point) in one direction while being at a minimum in all other directions. While propagating on such a shoulder, the gradient norms can be fairly small. However, as the optimization procedure updates the molecular orbitals beyond the edge of the shoulder a rapid decrease of the energy will initially be associated with a significant increase of the norm of the error vectors. It is clear that the (G)DIIS method in such a case will produce an extrapolated set of coordinates which returns up to the shoulder – a region of error vectors with a low norm. In the present implementation of the GDIIS procedure, tries to detect this case by finding the smallest element $B_{i,i}$ and checking the condition $E_n + \delta < E_i$, where $\delta = 10^{-4}$. Additionally, to ensure the B_{jj} values are in roughly descending order, it is checked whether any $B_{ii}\sigma < B_{j+j,j+1}$, with $\sigma = 15$. If any of these conditions is fulfilled, the depth of the (G)DIIS procedure, m , is reset to 1, and a pure variable metric step is taken based on the gradient of the latest trial.

Third, the updated Hessian is used in the two steps of the GDIIS procedure and thus it is important that it be qualitatively correct. Since the error vectors are also a part of these updates, they will only make sense if the gradients are consistent. Thus, under similar conditions as described above the BFGS procedure might produce updated Hessians which are nonphysical and/or ill-conditioned – usually detected by a series of monotonically smaller and smaller displacements in the GDIIS iterations suddenly interrupted by an unexpectedly large displacement. In the current implementation this is considered to be the case if the norm of the displacement vector is larger than π – i.e., a rotation larger than 180° . The remedy for this behavior is that a new update vector is computed using a reduced depth of the BFGS-update procedure repeatedly until the norm of the displacement vector is acceptable.

Restricted-Step Rational-Function Optimization

In the past, the so-called restricted-step rational-function optimization (RS-RFO) procedure^{43,44} has been applied with success to molecular structure optimizations in association with the use of internal coordinates⁴⁵, an approximate initial Hessian⁴⁶, and the BFGS Hessian update method. Inspired by the 1992 paper by Fischer and Almlöf²⁹, in which it was suggested that the successful use of the DIIS method, parameterized over the occupied–virtual block of the Fock matrices^{5,6}, for SCF orbital optimizations, could be transferred over to a parameterization of the procedure in terms of orbital rotations, a similar adaptation will be executed here. It is noted that such a transfer is obvious – a new alternative parameterization has emerged, and good estimates of the Hessian matrix exist. The only matter that is a possible restriction in the adaptation of RS-RFO to orbital optimization is the size of the parameter space, which for a given system goes from $3N - 6$, in the case of a molecular structure optimization (N being the number of atoms), to $N_{\text{SCF}} = N_{\text{occ}} \times (N_{\text{AO}} - N_{\text{occ}})$ in the case of a SCF orbital optimization (N_{AO} the total number of linearly independent basis functions and N_{occ} the number of occupied orbitals). That is, for a basis set of reasonable quality in association with second-row atoms, the characteristic dimension of the matrices associated with the RFO procedure increases by one order of

magnitude with respect to a typical geometry optimization, and the scaling of the procedure might increase by as much as three orders of magnitude. Moreover, while $3N - 6$ scales linearly with the number of atoms, N_{SCF} scales quadratically. For a successful implementation this issue will have to be mitigated.

The basics of the RS-RFO procedure are now briefly presented here. It is an alternative to a conventional step-restricted truncated second-order Taylor expansion of the energy in which the energy is expressed by a rational function – a Padé [2/2] approximant – as

$$E(\boldsymbol{\kappa}) = E(\boldsymbol{\kappa}_0) + \frac{1}{2} \frac{\begin{pmatrix} 1 & \delta\boldsymbol{\kappa}^\top \end{pmatrix} \begin{pmatrix} 0 & \mathbf{g}^\top \\ \mathbf{g} & \mathbf{H} \end{pmatrix} \begin{pmatrix} 1 \\ \delta\boldsymbol{\kappa} \end{pmatrix}}{\begin{pmatrix} 1 & \delta\boldsymbol{\kappa}^\top \end{pmatrix} \begin{pmatrix} 1 & \mathbf{0}^\top \\ \mathbf{0} & \alpha\mathbf{S} \end{pmatrix} \begin{pmatrix} 1 \\ \delta\boldsymbol{\kappa} \end{pmatrix}} \quad (17)$$

where $\delta\boldsymbol{\kappa} = \boldsymbol{\kappa} - \boldsymbol{\kappa}_0$, \mathbf{S} is a matrix usually set to the unit matrix, and α is a parameter which in the case of unconstrained optimizations is set to unity, but in the case of constrained optimizations it is adjusted to get a displacement with a norm within the step-restriction length.

The stationary points of the RS-RFO equation are found as solutions to eigenvalue equations

$$\begin{pmatrix} 0 & \mathbf{g}^\top \\ \mathbf{g} & \mathbf{H} \end{pmatrix} \begin{pmatrix} \nu_{1,i} \\ \mathbf{v}_i \end{pmatrix} = \lambda_i \begin{pmatrix} 1 & \mathbf{0}^\top \\ \mathbf{0} & \alpha\mathbf{S} \end{pmatrix} \begin{pmatrix} \nu_{1,i} \\ \mathbf{v}_i \end{pmatrix} \quad (18)$$

such that the computed eigenvectors are associated with the displacement vectors as

$$\begin{pmatrix} 1 \\ \delta\boldsymbol{\kappa} \end{pmatrix} = \frac{1}{\nu_{1,i}} \begin{pmatrix} \nu_{1,i} \\ \mathbf{v}_i \end{pmatrix} \quad (19)$$

The critical part of the implementation of the solver to the eigenvalue problem is (i) only the lowest eigenvalue is needed, and (ii) the full Hessian matrix is too large for explicit storage. This is trivially solved as follows. First, the eigenvalue problem will be solved using the iterative method of Davidson⁴⁷, which will significantly reduce the computational expense

and storage requirements as compared to any general procedure finding all eigenvalues. Second, in the iterative procedure the on-the-fly application of the BFGS update, as suggested by Fischer and Almlöf²⁹, was implemented. As in the case for the implementation of the GDIIS method, mentioned above, the depth of the BFGS procedure is modified if the suggested step length is longer than π . In that case the depth of the BFGS update is set to 1 and the step restriction is set to π (the default value is 10.0 if $m \neq 1$, such that poor BFGS updates can be diagnosed and the proper care can be taken).

To conclude this section, it is appropriate to mention the recent implementation by Slattery et al. of a quasi-Newton unitary optimization with trust-region approach.⁴⁸ There are many similarities with what is proposed above. The main differences are that Slattery et al. are using the low memory version of BFGS, the L-BFGS method,⁴⁹ the original step is computed with a conventional quasi-Newton step, an optional line search and a subsequent step restriction is implemented by the use of a trust radius. The trust radius is compiled from the most recent line search.

Gradient-Enhanced Kriging

The gradient-enhanced Kriging^{31–33} (GEK) approach – a Gaussian process regression⁵⁰ variant – is a machine-learning method which has been applied with significant success to, for example, molecular structure optimizations.^{30,51–53} In this report such a GEK implementation especially modified for the case of SCF orbital optimizations is presented. The size of the parameter space, N_{SCF} , however, is a significant problem for an efficient implementation, as reported by C. L. Ritterhof⁵⁴, who demonstrated the GEK procedure applied to SCF orbital optimization can have competitive iteration counts, while the timings are very unfavorable due to the N^3 scaling of the procedure. The author of this report proposes, though, that the general *effective* dimensionality of the SCF optimization problem is much smaller than the formal size. Considering, for example, that the GDIIS procedure normally converges in 15–25 iterations, using a depth of 4–5 iterations, one has to conclude that the effective dimensionality is much smaller than what one expects – this is mainly because the different degrees of freedom are to a large extent uncoupled. Hence, inspired by

the “smallness” of the DIIS formulation and the structure of second-rank Hessian-update methods, a subspace formalism of the GEK procedure (S-GEK), adapted for SCF/KS-DFT orbital optimizations, will be put forward. However, before presenting the space-reduction procedure the basics of the GEK implementation is introduced here. The details of this procedure have been published elsewhere^{30,51,52}.

The GEK is a gradient-enhanced Gaussian process regression, that is, not only the energy but also all elements of the gradient are regressed. Hence, the surrogate model reproduces exactly the energy and the complete gradient vectors at the points of regression – the model is fitted to the data, not the reverse.

The surrogate model is effectively expressed as

$$E(\boldsymbol{\kappa})^* = \mu + \mathbf{v}(\boldsymbol{\kappa})^\top \mathbf{M}^{-1} \mathbf{y} \quad (20)$$

where μ is the bias/trend function (can be a constant or a function of $\boldsymbol{\kappa}$), \mathbf{v} is the generalized covariance vector, \mathbf{M} is the generalized covariance matrix and \mathbf{y} is the generalized value vector. This can also, alternatively be expressed as

$$E^*(\boldsymbol{\kappa}) = \mu + \sum_{i=1}^n w_i v_i(\boldsymbol{\kappa}) + \sum_{i=1}^n \sum_{k=1}^K u_{i,k} \frac{\partial v_i(\boldsymbol{\kappa})}{\partial (\boldsymbol{\kappa})_k} \quad (21)$$

where w_i and $u_{i,k}$ are weights derived from the expression $\mathbf{M}^{-1} \mathbf{y}$, n is the depth of the GEK, that is, the number of sets of coordinates for which the energy and gradient vector is known, and, finally, K is the dimensionality of the parameter space.

Although in Eq. (21) the contributions from $v_i(\boldsymbol{\kappa})$ and $\partial v_i(\boldsymbol{\kappa})/\partial (\boldsymbol{\kappa})_k$ are written separately, in practice they are all collected in a single list, such that each element of the vectors $\mathbf{v}(\boldsymbol{\kappa})$ and \mathbf{y} or the matrix \mathbf{M} refers to either the energy or the derivative at a data point i in some predefined order. For instance, one could sort first the energies for all data points, then all derivatives for the first data point, the derivatives for the second data point, etc.

Thus, the \mathbf{y} vector would be (expressed as the transpose of a row vector for convenience):

$$\mathbf{y} = \left(\mathbf{y}_0^\top \quad \mathbf{y}_1^\top \quad \cdots \quad \mathbf{y}_n^\top \right)^\top \quad (22)$$

$$\mathbf{y}_0^\top = \left(E(\boldsymbol{\kappa}_1) \quad \cdots \quad E(\boldsymbol{\kappa}_n) \right) \quad \mathbf{y}_i^\top = \left(\frac{\partial E(\boldsymbol{\kappa})}{\partial(\boldsymbol{\kappa})_1} \quad \cdots \quad \frac{\partial E(\boldsymbol{\kappa})}{\partial(\boldsymbol{\kappa})_K} \right)_{\boldsymbol{\kappa}=\boldsymbol{\kappa}_i}$$

where $\boldsymbol{\kappa}_i$ indicates the coordinate vector of data point i while $(\boldsymbol{\kappa})_k$ denotes the k th dimension, or the k th component of the $\boldsymbol{\kappa}$ vector. That is, the \mathbf{y} vector collects all the energies and gradients from the n previous iterations, which the surrogate model will reproduce exactly. The \mathbf{M} matrix collects the covariance between the data points, as well as the first and second derivatives:

$$\mathbf{M} = \begin{pmatrix} \mathbf{M}_{00} & \mathbf{M}_{01} & \cdots & \mathbf{M}_{0n} \\ \mathbf{M}_{10} & \mathbf{M}_{11} & \cdots & \mathbf{M}_{1n} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{M}_{n0} & \mathbf{M}_{n1} & \cdots & \mathbf{M}_{nn} \end{pmatrix} \quad \mathbf{M}_{00} = \begin{pmatrix} f(\boldsymbol{\kappa}_1, \boldsymbol{\kappa}_1) & \cdots & f(\boldsymbol{\kappa}_n, \boldsymbol{\kappa}_1) \\ \vdots & \ddots & \vdots \\ f(\boldsymbol{\kappa}_1, \boldsymbol{\kappa}_n) & \cdots & f(\boldsymbol{\kappa}_n, \boldsymbol{\kappa}_n) \end{pmatrix} \quad (23)$$

$$\mathbf{M}_{0i} = \mathbf{M}_{i0}^\top = \begin{pmatrix} \frac{\partial f(\boldsymbol{\kappa}_1, \boldsymbol{\kappa})}{\partial(\boldsymbol{\kappa})_1} & \cdots & \frac{\partial f(\boldsymbol{\kappa}_n, \boldsymbol{\kappa})}{\partial(\boldsymbol{\kappa})_1} \\ \vdots & \ddots & \vdots \\ \frac{\partial f(\boldsymbol{\kappa}_1, \boldsymbol{\kappa})}{\partial(\boldsymbol{\kappa})_K} & \cdots & \frac{\partial f(\boldsymbol{\kappa}_n, \boldsymbol{\kappa})}{\partial(\boldsymbol{\kappa})_K} \end{pmatrix}_{\boldsymbol{\kappa}=\boldsymbol{\kappa}_i} \quad \mathbf{M}_{ij} = \begin{pmatrix} \frac{\partial^2 f(\boldsymbol{\kappa}, \boldsymbol{\kappa}')}{\partial(\boldsymbol{\kappa})_1 \partial(\boldsymbol{\kappa}')_1} & \cdots & \frac{\partial^2 f(\boldsymbol{\kappa}, \boldsymbol{\kappa}')}{\partial(\boldsymbol{\kappa})_K \partial(\boldsymbol{\kappa}')_1} \\ \vdots & \ddots & \vdots \\ \frac{\partial^2 f(\boldsymbol{\kappa}, \boldsymbol{\kappa}')}{\partial(\boldsymbol{\kappa})_1 \partial(\boldsymbol{\kappa}')_K} & \cdots & \frac{\partial^2 f(\boldsymbol{\kappa}, \boldsymbol{\kappa}')}{\partial(\boldsymbol{\kappa})_K \partial(\boldsymbol{\kappa}')_K} \end{pmatrix}_{\substack{\boldsymbol{\kappa}=\boldsymbol{\kappa}_i \\ \boldsymbol{\kappa}'=\boldsymbol{\kappa}_j}}$$

and the vector $\mathbf{v}(\boldsymbol{\kappa})$ collects the covariance and derivatives between an arbitrary point $\boldsymbol{\kappa}$ and the data points $\boldsymbol{\kappa}_i$:

$$\mathbf{v}(\boldsymbol{\kappa}) = \left(\mathbf{v}_0(\boldsymbol{\kappa})^\top \quad \mathbf{v}_1(\boldsymbol{\kappa})^\top \quad \cdots \quad \mathbf{v}_n(\boldsymbol{\kappa})^\top \right)^\top \quad (24)$$

$$\mathbf{v}_0(\boldsymbol{\kappa})^\top = \left(f(\boldsymbol{\kappa}, \boldsymbol{\kappa}_1) \quad \cdots \quad f(\boldsymbol{\kappa}, \boldsymbol{\kappa}_n) \right) \quad \mathbf{v}_i(\boldsymbol{\kappa})^\top = \left(\frac{\partial f(\boldsymbol{\kappa}, \boldsymbol{\kappa}_i)}{\partial(\boldsymbol{\kappa})_1} \quad \cdots \quad \frac{\partial f(\boldsymbol{\kappa}, \boldsymbol{\kappa}_i)}{\partial(\boldsymbol{\kappa})_K} \right)$$

The functions $v_i(\boldsymbol{\kappa})$ are therefore just a shorthand notation for $f(\boldsymbol{\kappa}, \boldsymbol{\kappa}_i)$, and the weights w_i and $u_{i,k}$ in Eq. (21) are obtained as

$$\left(w_1 \quad \cdots \quad w_n \quad u_{1,1} \quad \cdots \quad u_{1,K} \quad \cdots \quad u_{n,1} \quad \cdots \quad u_{n,K} \right) = \left(\mathbf{M}^{-1} \mathbf{y} \right)^\top \quad (25)$$

A crucial element of the GEK procedure is the definition of the covariance function $f(\boldsymbol{\kappa}, \boldsymbol{\kappa}')$ (a symmetric function), which is the building block of the covariance vector and matrix. The covariance function is a mathematical description of the correlation between the values of two data points. Many choices are possible for a valid covariance function⁵⁵, and based on the successful experience with geometry optimization^{30,51,53} we chose a Matérn 5/2 covariance function⁵⁶ with individual characteristic length for each dimension. A metric is defined to measure the distance, in parameter space, between two data points as

$$d(\boldsymbol{\kappa}, \boldsymbol{\kappa}') = \sqrt{\sum_{k=1}^K \left(\frac{(\boldsymbol{\kappa})_k - (\boldsymbol{\kappa}')_k}{l_k} \right)^2} \quad (26)$$

where l_k is the characteristic length, defined below. Note that this reduces to a simple Euclidean distance if all $l_k = 1$. The covariance function then reads

$$f(\boldsymbol{\kappa}, \boldsymbol{\kappa}') = f(d(\boldsymbol{\kappa}, \boldsymbol{\kappa}')) = f(d) = \left(\frac{5}{3}d^2 + \sqrt{5}d + 1 \right) \exp(-\sqrt{5}d) \quad (27)$$

which has a Gaussian-like shape (in fact, a Gaussian or “squared exponential” would also be a possible covariance function to use). The internal regression parameterization is based on the eigenvectors of the approximate Hessian of the latest iteration, that is, the $\boldsymbol{\kappa}$ vector used in the whole of this section is actually $\tilde{\boldsymbol{\kappa}} = \mathbf{U}\boldsymbol{\kappa}$, where \mathbf{U} is the matrix of eigenvectors of the approximate Hessian and $\boldsymbol{\kappa}$ is the vector of “raw” occupied–virtual orbital rotation parameters in the current orbital basis.

When the model is built with a single data point, using a Matérn 5/2 covariance function, the Hessian matrix at that point is diagonal, with eigenvalues:

$$\epsilon_k = \frac{5(\mu - E)}{3l_k^2} \quad (28)$$

This observation is used to derive the associated characteristic length for each dimension, in the representation that diagonalizes the Hessian, as

$$l_k = \sqrt{\frac{5(\mu - E_{\max})}{3\epsilon_k}} \quad E_{\max} = \max_i \{E(\boldsymbol{\kappa}_i)\} \quad (29)$$

considering that in this implementation $\mu - E_{\max} = 10.0 E_h$. This has the effect of reproducing the approximate Hessian when a single data point is used to build the model.

The GEK does not only allow for prediction of energy values, but also, to each such prediction, associate an estimate of the accuracy of that particular value – the expected variance

$$s^2(\boldsymbol{\kappa}) = \frac{(\mathbf{y} - \mathbf{1}\mu)^\top \mathbf{M}^{-1}(\mathbf{y} - \mathbf{1}\mu)}{n} \left[1 - \mathbf{v}(\boldsymbol{\kappa})^\top \mathbf{M}^{-1} \mathbf{v}(\boldsymbol{\kappa}) \right] \quad (30)$$

This facilitates an analytically based robustness measure to be applied to the procedure – rather than to make the optimization procedure restricted by an arbitrary step length, a variance restriction can be applied. This has been expressed in the restricted-variance optimization (RVO), which has with success been used in molecular structure optimizations.^{30,51,53}

Subspace Gradient-Enhanced Kriging

To proceed to the novel contribution reported in this work – the parameter space reduction – let us consider some technical details of the of the DIIS and the quasi-Newton procedure. In the first case, satisfactory convergence rates with DIIS are achieved by projecting the multi-dimensional error vectors, expressed either as displacement vectors or as gradients, of the last few iterations onto a small subspace spanned by the error vectors. This defines the **B** matrix, which is used to solve the DIIS equations. Subsequently, a single quasi-Newton step in the full parameter space is performed on the residual error vector. In the second case, the Hessian update of the quasi-Newton procedure is the fundamental tool towards convergence acceleration. It is noted that, for example, the BFGS second-rank update procedure can technically be described as the following update:

$$\mathbf{H}_{n+1} = \mathbf{H}_n + \alpha \mathbf{u}\mathbf{u}^\top + \beta \mathbf{v}\mathbf{s}^\top \quad (31)$$

where, α and β are some real constants, and $\mathbf{u} = \Delta \mathbf{g}$ and $\mathbf{s} = \Delta \boldsymbol{\kappa} \mathbf{H}$. In particular, it is noted that it is these vectors which communicate efficiently the most significant parts of the coupling between the parameters. It is the availability of an accurate Hessian approximation,

obtained through this procedure, which to a large extent explains the much faster convergence of the quasi-Newton approach compared to a plain conjugate gradient method. Inspired by the efficiency of this approach, a set of unit projection vectors, defined in the full space and a reduced space as \mathbf{e}_i^f and \mathbf{e}_i^r , to define an effective subspace was designed by selecting from (a) the actual displacement vectors of the last $m - 1$ iterations, and (b) the associated gradient difference vectors. To this list is further added the gradient vector at the latest point and the vector corresponding to the predicted displacement of RS-RFO procedure at the current structure. This set of $2m$ vectors in the full space is subsequently orthonormalized to generate \mathbf{e}_i^f , while the corresponding vectors in the reduced space are set to $(\mathbf{e}_i^r)_j = \delta_{ij}$, i.e. they are canonical basis vectors in the reduced space. This defines the projection operator

$$\hat{P} = \sum_i \mathbf{e}_i^r (\mathbf{e}_i^f)^\top \quad (32)$$

In what then follows, the original coordinate vectors, gradient vectors and the approximate Hessian are projected onto this subspace and the resulting entities are subsequently used in a RVO procedure supported by the GEK approximation. For example, a gradient in the full space, \mathbf{g}^f , is expressed as

$$\mathbf{g}^r = \sum_i (\mathbf{e}_i^f)^\top \mathbf{g}^f \mathbf{e}_i^r \quad (33)$$

in the reduced space. The resulting displacement, $\delta\boldsymbol{\kappa}^r$, from solving the optimization in the reduced space is finally expanded up into the full space by

$$\delta\boldsymbol{\kappa}^f = \sum_i (\mathbf{e}_i^r)^\top \delta\boldsymbol{\kappa}^r \mathbf{e}_i^f \quad (34)$$

This procedure, the S-GEK/RVO or S-RVO approach, can be used as a convergence acceleration approach on top of the previously proposed RS-RFO method. It significantly reduces the dimensionality of the S-RVO procedure from $m \times (1 + N_{\text{SCF}})$ to $m \times (1 + 2m)$, making it a viable option for SCF/KS-DFT orbital optimization.

Computational Details

The object of the benchmark suite to be designed is to offer a vigorous test of the merits of the three optimization procedures presented in this study. The benchmarking will explore both SCF and DFT optimizations, this in order to explore performance as a function of the different complexity of the non-linear parameterization of the energy as expressed in SCF vs. DFT. Moreover, to evaluate the performance of the three SCF/DFT orbital optimization methods, a comprehensive data set of 2080 target electronic structures was employed, encompassing both organic molecules and transition-metal complexes. The goal of the data sets was to constitute representative cases of both closed- and open-shell wave functions, to contain cases of strained wave functions as represented by non-equilibrium molecular structures, and include cases with potential near-degeneracy effects as represented by transition-metal (TM) complexes. In this respect, it is the expectation that the data set presented below is not biased and does not contain tests which are cherry-picked to favor one or another of the optimization methods. Moreover, with this selection of test suite, it comes naturally that any subsequent analysis on the merits of the three methods has to be based on a statistical comparison. This also eliminates the significance of any parameter sensitivity some molecular structures might exhibit in individually compiled convergence rates.

Thus, structures beyond stationary points and structures containing transition metals, as well as those beyond the ground electronic states, were utilized to evaluate the robustness of the reported SCF/DFT optimization procedures. Among these, a subset of 500 (265 singlets + 235 doublets) organic molecules was selected from the Sella⁵⁷ database to represent organic molecules composed of 7 to 25 atoms. The original Sella database contains molecular structures close to a transition state structure for both singlet and doublet spin states. These structures were subsequently optimized to either the reactant or the product equilibrium molecular structure, thus doubling the number of molecular structures. The two sets of molecular structures in the singlet spin state were subsequently used for the benchmarking of wave functions in both the singlet and triplet states. Hence, the organic molecules were subjected to calculations in singlet, doublet, and triplet states, involving

both stationary and transition state structures. Additionally, a subset of 275 structures was drawn from the tmQM⁵⁸ dataset to represent transition-metal complexes with fewer than 20 atoms. Here, the transition-metal complexes were computed in both their singlet and triplet electronic states. To summarize: 265 singlet TS-like structures were computed as both singlet and triplet; those structures were optimized to a stable singlet structure, and again the wave function was computed as singlet and triplet; 235 doublet TS-like structures were computed as doublets; after optimization to stable doublet structures the doublet wave function was again computed; 275 transition-metal-complex structures were computed as singlet and triplet. This gives $265 \times 4 + 235 \times 2 + 275 \times 2 = 2080$ total structures for each SCF optimization method. It should be noted that the geometry optimizations were done only once at the HF/3-21G level of theory, and the resulting structures were used for all reported calculations as single points.

All calculations were carried out using the OpenMolcas software package, version 23.06⁵⁹. The evaluation of three distinct SCF procedures was conducted utilizing both HF and DFT-based methods. DFT calculations were done using the B3LYP functional⁶⁰. For organic molecules, the cc-pVDZ basis set⁶¹⁻⁶³ was adopted, while molecules containing transition metals were computed using the ANO-R1 basis set^{64,65} (the use of ANO-R1 in OpenMolcas automatically enables the exact 2-component (X2C)⁶⁶ scalar relativistic Hamiltonian and finite-size nuclei). Singlet calculations were done with the restricted SCF formulation, doublet and triplet calculations with the spin-unrestricted one. The default SCF convergence criteria were utilized (energy change below $10^{-9} E_h$, a maximum absolute value of the occupied–virtual Fock matrix below 1.5×10^{-4} , and a norm of $\Delta\kappa$ below 10^{-3}), along with the RICD approximation⁶⁷, while the default LK procedure⁶⁸ was disabled. For DFT calculations, the default Libxc⁶⁹ implementation and numerical quadrature in OpenMolcas were employed. The calculations were conducted without the imposition of any symmetry constraints. The performance of the three SCF procedures was assessed by comparing the number of SCF iterations. If not otherwise stated, this is the iteration count after the standard SCF startup procedure, which consists of both the “initial guess” of occupied orbitals and a number of preliminary iterations with EDIIS⁸ and DIIS⁴² procedures, as im-

plemented in OpenMolcas and described in the appendix, such that the reported iteration counts refer only to the three methods benchmarked in this work.

Results and Discussion

The results in this section will be presented into two parts. First, the significance of the resetting technique of the r-GDIIS procedure will be assessed as compared to a vanilla GDIIS implementation. Second, the different orbital optimization methods – the r-GDIIS, the RS-RFO and the S-GEK/RVO methods – will be benchmarked against each other.

Performance assessment of the r-GDIIS approach

The suggested resetting technique of the r-GDIIS method is here compared versus a straightforward implementation of the GDIIS with respect to iterations until convergence according to the default threshold. The average number of iterations and the standard deviation of the iteration count for the eight separate sets of benchmark calculations as assessed with the HF and the B3LYP methods are presented in Table 1. Additionally, the table contains information, for each set, on the number of molecules for which the calculation took more than 120 iterations to converge and the number of molecules for which convergence was not reached before 400 iterations (including the SCF startup in this latest count). Note that the statistics is based on the cases in which convergence was reached. Hence, an apparent advantage from the statistics could be a sign that problematic cases did not converge and in this sense did not contribute to the evaluation of the statistical data.

The following three general observations can be noted. First, the B3LYP optimization procedure seems in general to be less problematic to converge as compared to HF in the cases of open-shell calculations. Second, the triplet state transition metal benchmarks stands out as a more challenging task both at the HF and the B3LYP level of theory as compared to the other benchmark sets. Third, convergence is faster for benchmark molecular structures that correspond to equilibrium molecular structures – note that the neither triplet benchmark set corresponds to equilibrium molecular structures, as they are opti-

Table 1: Benchmark results for the eight sets of benchmark calculations compiled with the GDIIS and r-GDIIS approaches. The reported numbers are the average number of iterations post SCF startup (the standard deviation) / the number of molecules for which more than a total of 120 iterations was needed (the number of molecules which did not converge before a total of 400 SCF iterations).

	HF		B3LYP	
	GDIIS	r-GDIIS	GDIIS	r-GDIIS
Singlets	11.2(2.5) / 0(0)	11.3(2.9) / 0(0)	9.4(2.2) / 0(3)	9.8(3.1) / 0(0)
Singlets Opt	7.9(1.0) / 0(0)	7.9(1.0) / 0(0)	8.1(1.4) / 0(0)	8.1(1.4) / 0(0)
Doublets	15.4(5.3) / 0(19)	17.1(8.4) / 0(0)	11.0(2.4) / 0(2)	11.1(3.0) / 0(0)
Doublets Opt	12.9(6.0) / 0(11)	15.2(13.4) / 0(0)	10.0(3.7) / 0(5)	11.4(12.2) / 1(0)
Triplets	11.5(5.9) / 0(4)	12.3(8.1) / 0(0)	7.0(2.5) / 0(0)	7.0(2.5) / 0(0)
Triplets Opt	12.6(15.8) / 1(24)	19.6(32.4) / 8(0)	8.6(4.1) / 0(0)	8.6(4.1) / 0(0)
TM Singlets	12.5(4.9) / 0(4)	13.2(7.6) / 0(0)	10.9(3.6) / 0(1)	11.4(3.8) / 0(5)
TM Triplets	34.0(20.0) / 1(102)	47.8(36.9) / 7(2)	24.6(29.5) / 2(48)	33.5(29.6) / 7(2)

mized for a singlet wave function. This is in line with the purpose of the design of the test suites. Finally, it is clear that the resetting of the DIIS depth is a significant improvement. For example, the resetting mechanism reduces the total number of failed convergence cases from 164 to 2, and from 59 to 7 for the HF and B3LYP methods, respectively. In particular, it is noted that for the triplet transitions metals close to half of the test cases fail to converge with the standard GDIIS – 102 cases out of 275 – whereas the resetting approach more or less eliminates this problem – 2 cases. This is exactly the reason why the transition metal sets were included in the benchmark suite: open-shell transition metal complexes are expected to constitute more of a challenge as compared to molecular systems made up of elements from the first three rows of the periodic table. It is also noted that the resetting approach can introduce a marginal increase in the iteration count but this comes with the benefit that convergence is almost guaranteed. To summarize, in general these results are a clear empirical documentation of the benefits of the resetting procedure. More specifically, for systems with complicated open-shell electronic structures the resetting mechanism can be the difference between frequent failures or not.

Benchmarking of three SCF orbital optimizers

The benchmark comparison between the r-GDIIS approach, the RS-RFO, and the S-RVO optimization procedures is now presented. First, Table 2 contains statistics with respect to average iterations with in each benchmark set, the standard deviation of the same measure, the number of systems converging after more than 120 iterations, and the number of molecules for which the optimization did not converge before a total of 400 SCF iterations. Relying on a single statistical measure to evaluate performance is known to be problematic⁷⁰; in particular, the mean and standard deviation can be very sensitive to extreme values. Therefore, somewhat more detailed data is presented in Figures 1, 2, 3 and 4, in which the distribution of the iteration counts is represented by so-called box-and-whisker plots. Here the box represents the first and third quartile, Q_1 and Q_3 (so that it includes 50 % of the cases), the line inside the box is the median value and second quartile. The whiskers extend to 1.5 times the interquartile range, $IQR = Q_3 - Q_1$, from the start and the end of the box, but always ending on a data point within the range. In the presented results the whiskers have different lengths since there are often no data points which are outside 1.5 times the IQR range to the left of the box, rather the data before Q_1 is rather compressed and is found close the left of the Q_1 marking. Any data points that are outside of the whiskers on either side are considered outliers and are represented explicitly. Note that the number of outliers is a functions of how tight the distribution is. That is, a benchmark can have more outliers as a consequence of the underlying distribution being very tight. Hence, the number of outliers between the two benchmarks should be compared with care. The median and IQR value should be much less sensitive to extreme values than the mean and standard deviation.

For all cases – monitoring both the average iterations and the standard deviations – one can observe the following trends. The KS-DFT optimizations convergence typically faster compared to HF. Moreover, the convergence tends to be swifter for the benchmark sets that represent molecules at equilibrium structures as compared to non-equilibrium structures – i.e. within the singlet and doublet sets. This is in particular true for the S-RVO optimizer. Exceptions exists, for example, the transition metal HF singlets optimizations in general

Table 2: Benchmark results for the eight sets of benchmark sets compiled with the r-GDIIS method, the RS-RFO, and the S-GEK/RVO approaches, respectively. The reported numbers are the average number of iterations post SCF startup (the standard deviation) / the number of molecules for which more than 120 total iterations was needed (the number of molecules which didn't converge before a total of 400 SCF iterations).

	HF		
	r-GDIIS	RS-RFO	S-GEK/RVO
Singlets	11.3(2.9) / 0(0)	11.7(2.3) / 0(0)	10.7(2.0) / 0(0)
Singlets Opt	7.9(1.0) / 0(0)	8.2(1.3) / 0(0)	7.9(1.0) / 0(0)
Doublets	17.1(8.4) / 0(0)	17.6(6.7) / 0(0)	15.1(5.5) / 0(0)
Doublets Opt	15.2(13.4) / 0(0)	17.3(22.1) / 1(0)	14.8(11.7) / 0(0)
Triplets	12.3(8.1) / 0(0)	13.4(6.7) / 0(0)	11.1(5.4) / 0(0)
Triplets Opt	19.6(32.4) / 8(0)	18.7(17.0) / 0(0)	14.7(12.1) / 0(0)
TM Singlets	13.2(7.6) / 0(0)	15.7(13.6) / 0(0)	13.8(10.7) / 0(0)
TM Triplets	47.8(36.9) / 7(2)	49.5(27.2) / 7(0)	41.8(24.3) / 4(0)
	B3LYP		
	r-GDIIS	RS-RFO	S-GEK/RVO
Singlets	9.8(3.1) / 0(0)	11.2(4.4) / 0(0)	9.0(3.0) / 0(0)
Singlets Opt	8.1(1.4) / 0(0)	9.7(2.0) / 0(0)	7.5(1.1) / 0(0)
Doublets	11.1(3.0) / 0(0)	11.8(3.5) / 0(0)	10.5(2.9) / 0(0)
Doublets Opt	11.4(12.2) / 1(0)	12.4(14.9) / 1(0)	10.7(10.1) / 0(0)
Triplets	7.0(2.5) / 0(0)	7.4(2.8) / 0(0)	6.7(2.3) / 0(0)
Triplets Opt	8.6(4.1) / 0(0)	9.8(7.9) / 0(0)	8.1(3.8) / 0(0)
TM Singlets	11.4(3.8) / 0(5)	16.5(7.4) / 0(2)	10.0(4.3) / 0(0)
TM Triplets	33.5(29.6) / 7(2)	31.4(22.9) / 3(0)	24.5(14.5) / 0(0)

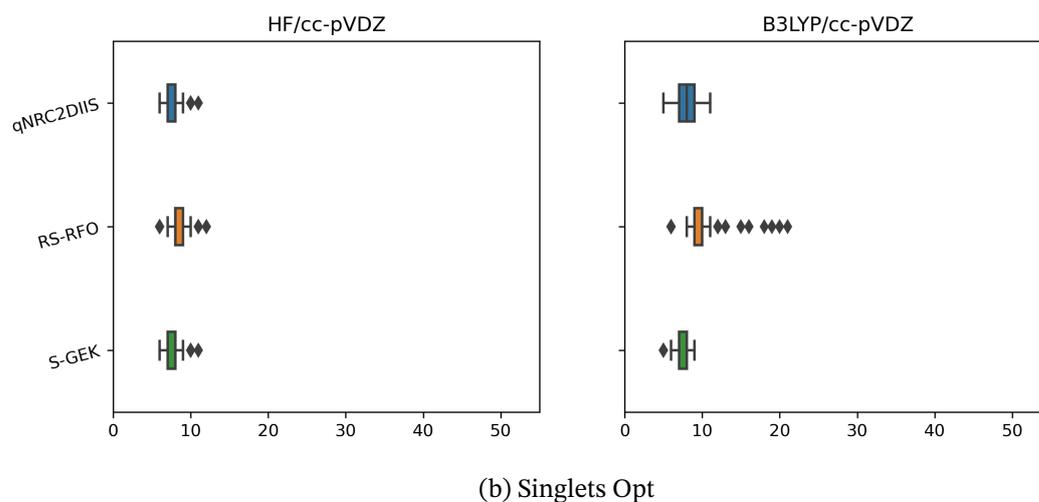
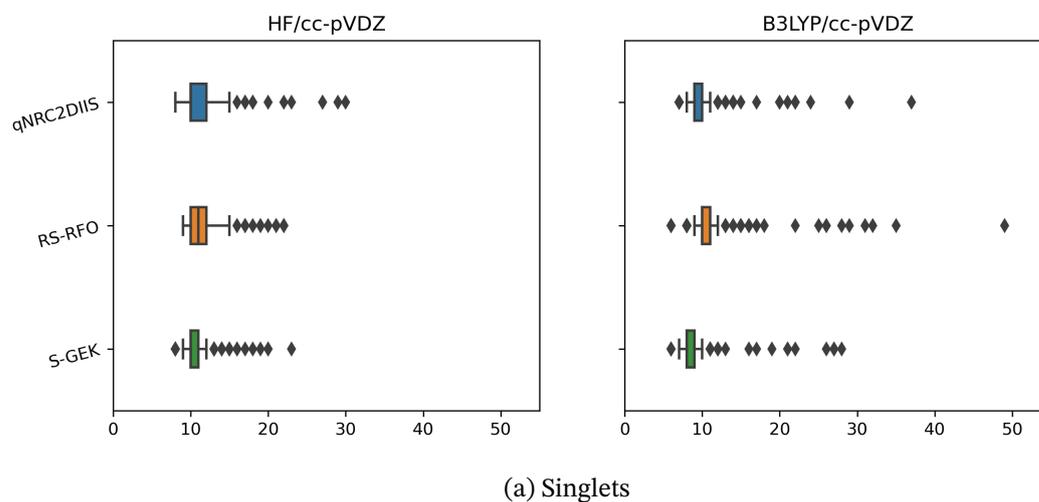


Figure 1: Benchmark iteration counts for the 2 benchmark sets of singlets – the Singlets and Singlets Opt set are presented at the upper and the lower panels, respectively – evaluated at the HF and KS-DFT (B3LYP) level of theory using the r-GDIIS, the RS-RFO, and the S-GEK/RVO SCF/KS-DFT orbital optimization methods, respectively. The x -axis represents the iteration count until convergence. The data is presented with box and whisker plots where the whisker range is derived by the 1.5 IQR-rule. Outliers are plotted as diamond-shaped signs.

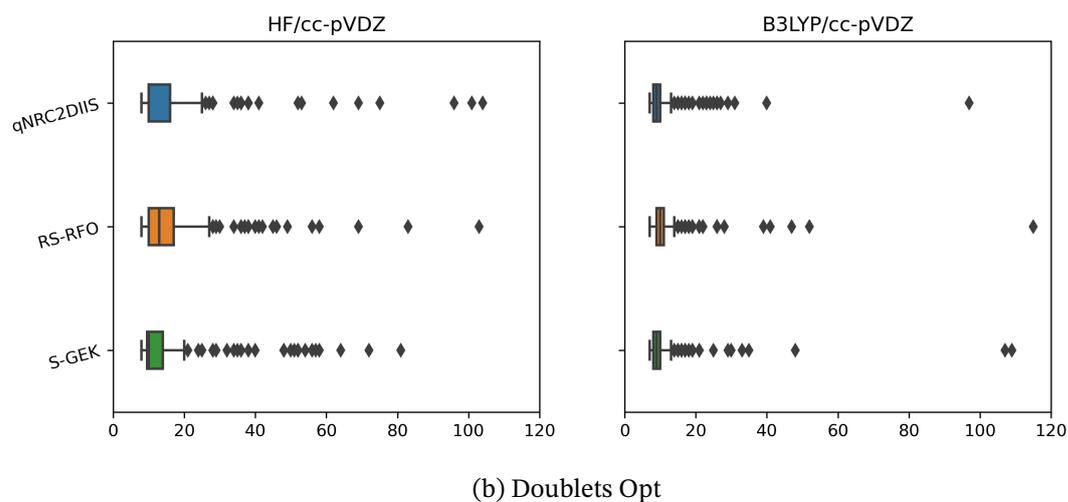
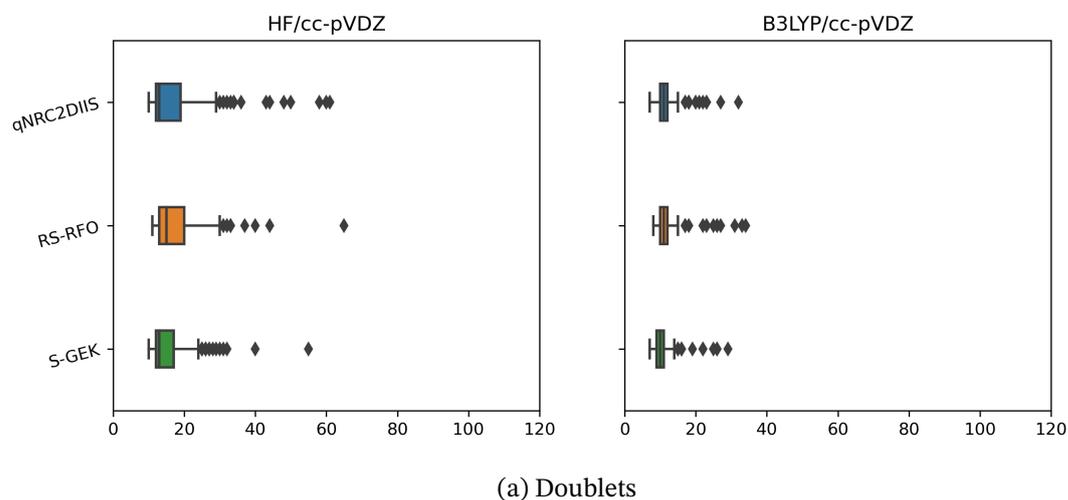


Figure 2: Benchmark iteration counts for the 2 benchmark sets of doublets – the Doublets and Doublets Opt set are presented at the upper and the lower panels, respectively – evaluated at the HF and KS-DFT (B3LYP) level of theory using the r-GDIIS, the RS-RFO, and the S-GEK/RVO SCF orbital optimization methods, respectively. The x-axis represents the iteration count until convergence. The data is presented with box and whisker plots where the whisker range is derived by the 1.5 IQR-rule. Outliers are plotted as diamond-shaped signs.

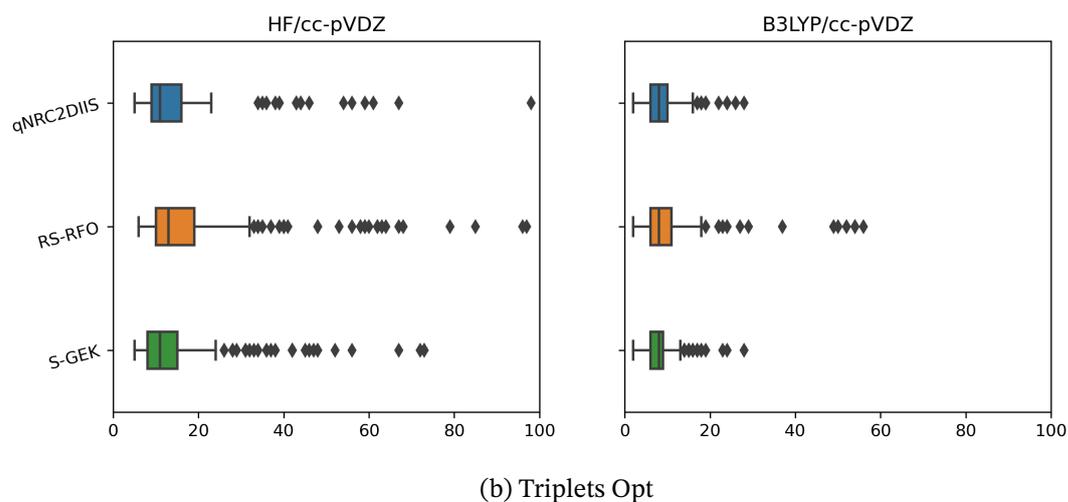
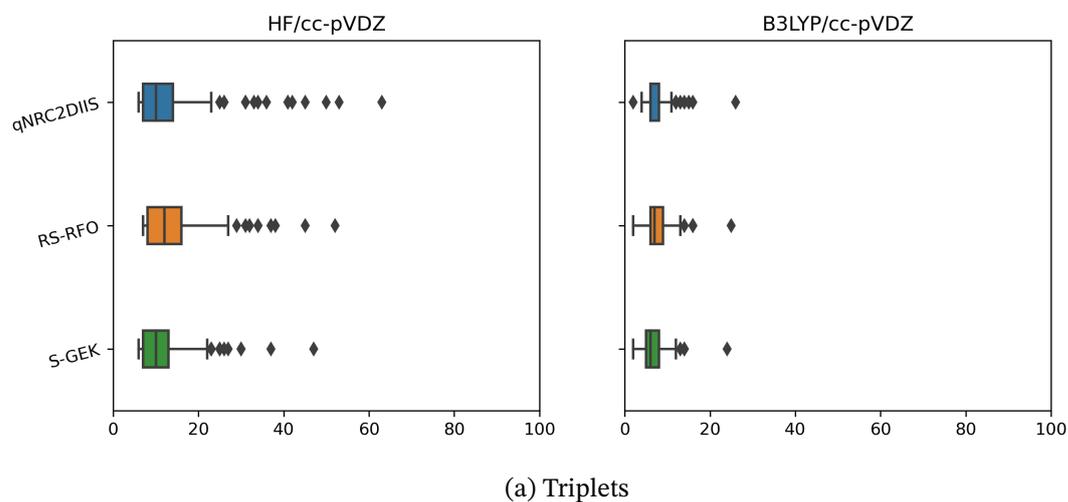


Figure 3: Benchmark iteration counts for the 2 benchmark sets of triplets – the Triplets and Triplets Opt set are presented at the upper and the lower panels, respectively – evaluated at the HF and KS-DFT (B3LYP) level of theory using the r-GDIIS, the RS-RFO, and the S-GEK/RVO SCF/KS-DFT orbital optimization methods, respectively. The x -axis represents the iteration count until convergence. The data is presented with box and whisker plots where the whisker range is derived by the 1.5 IQR-rule. Outliers are plotted as diamond-shaped signs.

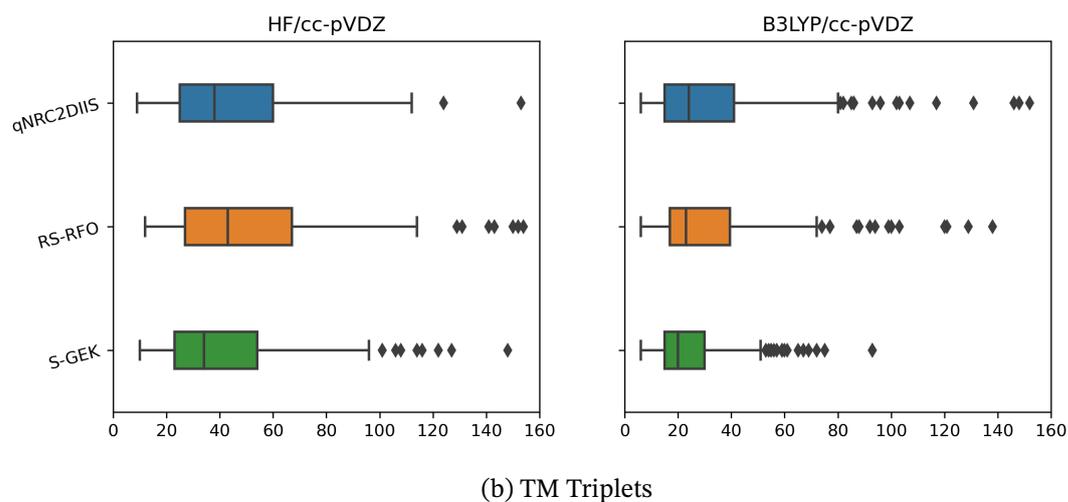
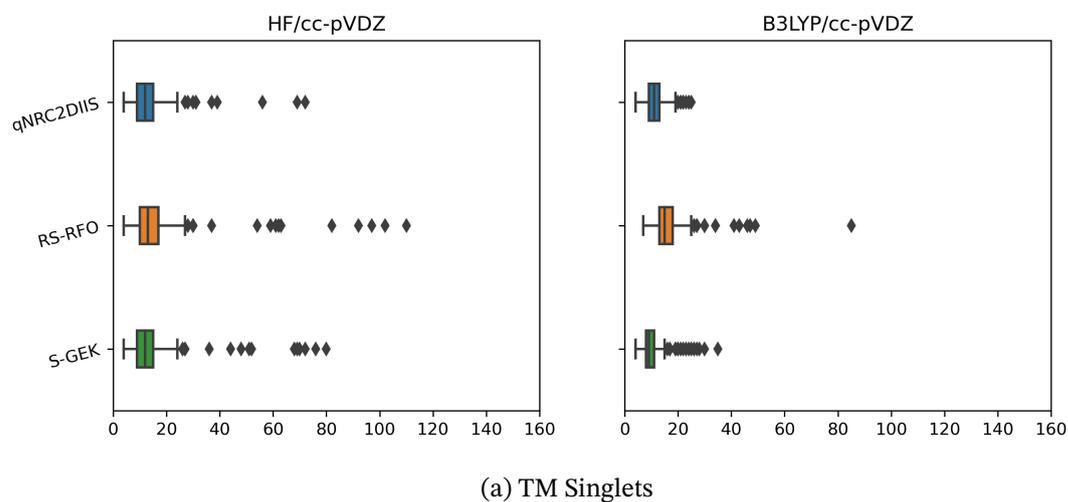


Figure 4: Benchmark iteration counts for the 2 benchmark sets of transition metals – the TM Singlets and TM Triplets set are presented at the upper and the lower panels, respectively – evaluated at the HF and KS-DFT (B3LYP) level of theory using the r-GDIIS, the RS-RFO, and the S-GEK/RVO SCF/KS-DFT orbital optimization methods, respectively. The x -axis represents the iteration count until convergence. The data is presented with box and whisker plots where the whisker range is derived by the 1.5 IQR-rule. Outliers are plotted as diamond-shaped signs.

converge faster than the corresponding KS-DFT optimizations. The most striking result is, however, that for all the eight test sets, using HF or KS-DFT, the S-RVO procedure shows superior performance as compared to both the r-GDIIS and the RS-RFO procedures, with the possible exception of the TM singlets with HF. For the test sets of the organic molecules this advantage is modest but consistent. For the transition metals the S-RVO demonstrates a significant advantage as compared to the two other optimization methods, this in particular for the open-shell (triplets) benchmark test suite – the average iterations are reduced from 47.8 to 41.8 when comparing r-GDIIS and S-GEK/RVO. For the KS-DFT optimization, the corresponding results are reduced from 33.5 to 24.5. The table and the corresponding figures also readily show that the observed dispersion is significantly reduced for the S-RVO method as compared to the two other methods, and the extreme values tend to be lower. It is noted that in some cases there are a handful of outliers beyond the maximum value displayed in the figures, but never for the S-RVO method. Finally, one can observe that the S-RVO procedure is the only one for which there is no single case in which the optimization did not converge. It is evident that the non-parametric surrogate model – although just expressed in a subspace – is a significant step forward toward a swift and robust SCF/KS-DFT orbital optimization method.

Briefly, on comparison of the r-GDIIS versus the RS-RFO method the results are a bit disappointing. The RS-RFO approach, as implemented in this study, does not exhibit any superiority over the r-GDIIS – a comparison with the GDIIS would of course leave a completely different verdict. The only cases where there seems to be some advantage of the RS-RFO as compared to the r-GDIIS approach can be found for the statistics of the KS-DFT optimizations of the transition metals. The origin of the generally slower convergence rate can one only speculate about. Is it that the step restriction is too tight? Considering that both methods are molded in the frame of a second-order method – both are quasi-Newton methods in some respect – one would not expect a significant difference in performance. Hence, the used thresholds for activating the step restriction and the underlying ad-hoc procedure involved in these decisions can be suspected. This will not, however, be analyzed any further here, especially when the restricted variance approach of the RVO procedure

is demonstrating a significant advantage over the step restriction procedure of the RS-RFO method, so the RVO procedure based on S-GEK should be the norm.

In addition to the iteration count, it is also interesting to analyze the quality of the converged wave function or orbitals. We do this by comparing the final energies obtained in the calculations, taking the S-GEK/RVO results as a reference, since it was the method that converged in all cases. Most of the 4160 total structures, considering separately the HF and B3LYP calculations, resulted in the same converged energy (when converged) with the four methods: GDIIS, r-GDIIS, RS-RFO and S-GEK/RVO. Only in 199 cases is the energy difference larger than $5 \times 10^{-7} E_h$. These are represented in Figure 5, where it is evident that most differences are positive, i.e. the S-GEK/RVO method converges to a lower (when not equal) energy than the other methods, except in a dozen of cases. It is also clear that GDIIS and r-GDIIS tend to differ more from S-GEK/RVO than RS-RFO. Moreover, it can be observed that most differences are found in the sets of transition metal complexes (123 cases) and, in general, more in HF than in B3LYP calculations (144 vs. 55).

This section ends with some timing observations and suggestions for future developments. First, it was noted that the RS-RFO approach has a significant longer timings per iteration as compared to the r-GDIIS approach. This most likely has its origin in that the former handles the full parameter space while the r-GDIIS only works in a very limited subspace. Second, the S-GEK/RVO was adopted post the RS-RFO procedure and in this context the S-GEK/RVO did not add significant additional timing to the underlying RS-RFO method. Considering the consistently better performance of the r-GDIIS as compared to RS-RFO – both for the convergence rate and the timings – one should explore the combination of the r-GDIIS and the S-GEK/RVO procedure for even better and faster convergence.

Conclusions

In this report three SCF/KS-DFT orbital optimization schemes have been investigated using a large benchmark test suite to evaluate performance characteristics. First a resetting

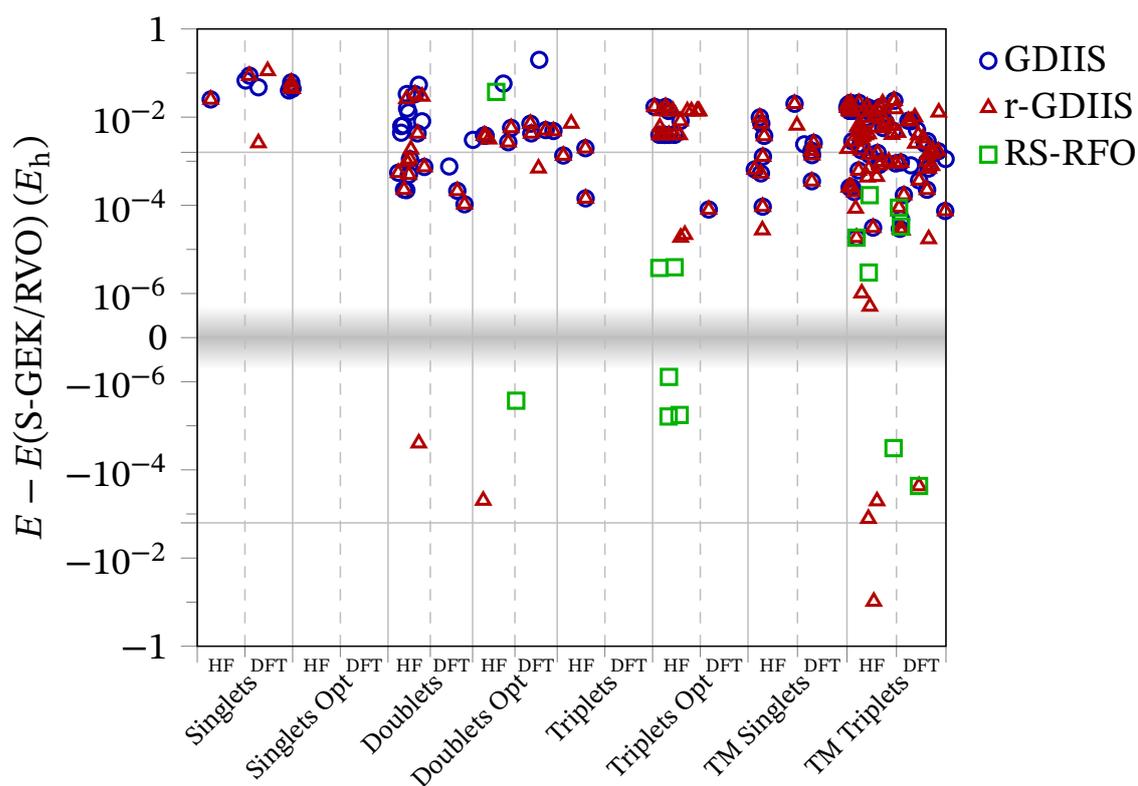


Figure 5: Difference between converged energies with respect to the S-GEK/RVO result. Only differences larger than $5 \times 10^{-7} E_h$ in absolute value are represented. Note that the y axis is divided in positive and negative sections, each of them in logarithmic scale. The horizontal grid lines mark ± 1 kcal/mol.

version of the GDIIS procedure, the r-GDIIS approach, was presented. This approach uses a number of different criteria to decide when the DIIS depth needs to be reset in order to avoid divergence or poor convergence rate. Second, a reduced-step rational function optimization scheme was presented. This method is facilitated by the use of the Davidson procedure in connection with an on-the-fly Hessian update procedure as the rational function equations are solved. Finally, the novel implementation of the gradient-enhanced Kriging adapted to a subspace formalism, S-GEK, was presented and implemented in the context of a restricted-variance optimization (RVO) procedure. The benchmark calculations included in total an excess of 2000 cases – singlet, doublet and triplet state organic molecular systems, and singlet and triplet state transition metal complexes – about half of the structures correspond to equilibrium molecular structures. The results of the benchmarking gave a clear indication that the S-GEK is superior to any of the conventional state-of-the-art methods of SCF/KS-DFT orbital optimization explored in this study. In particular, it was demonstrated that for transition metals the improvements in convergence rates are impressive and robust. Moreover, the new method exhibits sturdy characteristics with respect to handling difficult cases – in all there was not an single instance in which the S-GEK/RVO optimization failed to complete the optimization procedure.

To conclude, the paper suggest that the S-GEK/RVO procedure should be explored in conjunction with the r-GDIIS approach for optimal CPU timings. In addition, considering that the S-GEK/RVO approach is a non-parametric surrogate model there is no obvious requirement that it needs to start in the so-called quadratic region of the parameter space. That is, it should be worthwhile to investigate to what extent the iterations in the SCF startup scheme can be reduced by switching over to the S-GEK/RVO approximation earlier in the optimization procedure.

Acknowledgement

The authors wish to thank Susi Lehtola for helpful feedback and suggestions on the manuscript. The computations were performed at NSC Tetralith provided by the National

Academic Infrastructure for Supercomputing in Sweden (NAISS) and PReSTO funded by the Swedish Research Council through grant agreement no. 2022-06725 (NAISS) and 2018-06479 (PReSTO). Funding from the Swedish Research Council (VR, Grant No. 2020-03182) and Stiftelsen Olle Engkvist Byggmästare (SOEB) Grant No. 211-0019. and the project *AI4Research* at Uppsala University are acknowledged.

Supporting Information Available

Coordinates for all molecular systems computed in this work, converged energy and iteration count for each calculation. Model energy values used for the definition of the initial orbitals.

Appendix

The SCF optimization methods described, implemented and tested in this work are for the most part based on a second-order Taylor expansion, and therefore they work best when the wave function is relatively close to a minimum, such that the energy is approximately quadratic with respect to κ . Over the years, experience has shown that far from a minimum, performance tends to be poor and convergence can be an issue. To address this, several “SCF acceleration” schemes have been proposed whose aim is to guide the optimization in a fast and robust manner towards a quadratic region where efficient optimization methods can perform best. The process from the beginning of a calculation to the point where the final optimization method (GDIIS, r-GDIIS, RS-RFO, or S-GEK/RVO) is activated is what is referred to in this work as “SCF startup”, and includes both the definition of the starting orbitals and the initial SCF iterations. Given that this process is identical regardless of the chosen final method, the iterations used for it are excluded from the iteration counts reported in the main text. Although the startup process has not been the subject of this work, it will be briefly described in this appendix for completeness. It is the default startup in OpenMolcas⁵⁹, except where noted.

Starting Orbitals

The SCF procedure must begin with a starting set of molecular orbitals that define the reference Slater determinant. While in principle any orthonormal set of orbitals could be used, the process will benefit from an initial guess as close as possible to the final solution, facilitating rapid convergence to the ground state and avoiding spurious convergence to unphysical solutions. The performance of any optimization method is therefore dependent on how the starting orbitals are defined. In all cases the default in OpenMolcas⁵⁹ have been used. Nevertheless, since the process by which these starting orbitals are defined has not, to our knowledge, been previously described, we take the opportunity of doing so in this appendix, noting again that this is not strictly part of this work.

The initial (guess) orbitals for the SCF procedure are obtained by diagonalization of a synthetic model Fock matrix F^m . This model Fock matrix is constructed in AO basis, as a sum of atomic blocks, with interatomic blocks set to 0, i.e.

$$F_{\mu\nu}^m = \begin{cases} F_{\mu\nu}^A & \mu, \nu \in A \\ 0 & \mu \in A, \nu \in B, A \neq B \end{cases} \quad (35)$$

where A and B refer to atoms, or centers for basis functions. Each atomic block corresponds to a model Fock operator that is diagonal in a set of orthonormal reference functions:

$$F_{\kappa\lambda}^{A,\text{ref}} = \eta_\kappa \delta_{\kappa\lambda} \quad (36)$$

where η are the energies of the reference functions. The reference atomic Fock matrix is transformed to an arbitrary basis set via the overlaps between the basis set and the reference functions ($S^{A,\text{ref}}$, a rectangular matrix) and among the basis set functions (S^A , a symmetric

matrix):

$$S_{\mu\kappa}^{A,\text{ref}} = \langle \chi_\mu | \chi_\kappa \rangle \quad \mu \in A, \kappa \text{ is a reference function of } A \quad (37)$$

$$S_{\mu\nu}^A = \langle \chi_\mu | \chi_\nu \rangle \quad \mu, \nu \in A \quad (38)$$

$$\mathbf{F}^A = (\mathbf{S}^A)^{-1} \mathbf{S}^{A,\text{ref}} \mathbf{F}^{A,\text{ref}} (\mathbf{S}^{A,\text{ref}})^\top (\mathbf{S}^A)^{-1} \quad (39)$$

Once F^m is obtained by eq. (35), it is first transformed to an orthonormal basis, defined by canonical orthonormalization. If \mathbf{S} is the overlap matrix of the full system in the desired basis, the matrix \mathbf{L} is defined as

$$\mathbf{L} = \mathbf{V} \boldsymbol{\sigma}^{1/2} \quad (40)$$

where \mathbf{V} is the matrix of eigenvectors of \mathbf{S} and $\boldsymbol{\sigma}$ is the diagonal matrix of its eigenvalues. Then the transformed F^m is:

$$\tilde{\mathbf{F}}^m = \mathbf{L}^\top \mathbf{S} \mathbf{F}^m \mathbf{S} \mathbf{L} \quad (41)$$

Note that, in the event of a monoatomic system, the factors $(\mathbf{S}^A)^{-1}$ and \mathbf{S} in eqs. (39) and (41) will cancel out, but in general $\mathbf{S} \neq \mathbf{S}^A$. Diagonalization of $\tilde{\mathbf{F}}^m$ yields a set of eigenvectors and eigenvalues that can be used as starting orbital coefficients and energies for the SCF procedure.

The reference functions from which the atomic Fock matrices, eq. (36), are constructed are quite limited, only valence functions are typically included. Any reasonable calculation will contain many more functions, that thus span a larger Hilbert space. As a consequence, the diagonalization of $\tilde{\mathbf{F}}^m$ produces a large number of orbitals with zero or very small energies, which are not resolved. In order to generate a more useful and well-defined set of orbitals for this “virtual” space, they are subject to a further diagonalization using the kinetic energy operator.

The subset of eigenvectors of $\tilde{\mathbf{F}}^m$ corresponding to eigenvalues larger than $-10^{-3} E_h$, \mathbf{C}_{virt} , is used to obtain the kinetic energy matrix over the virtual space, from the full kinetic energy matrix in AO basis:

$$\mathbf{T}_{\text{virt}} = \mathbf{C}_{\text{virt}}^\top \mathbf{T}_{\text{AO}} \mathbf{C}_{\text{virt}} \quad (42)$$

and they are transformed to diagonalize the \mathbf{T}_{virt} :

$$\tilde{\mathbf{C}}_{\text{virt}} = \mathbf{C}_{\text{virt}} \mathbf{U} \quad (43)$$

where \mathbf{U} are the eigenvectors of \mathbf{T}_{virt} . The energies of these orbitals are taken as the corresponding eigenvalues, shifted by $3.0 E_{\text{h}}$, to ensure that they are well above the rest.

The reference functions and η values used in eq. (36) and (39) are the lowest-lying atomic natural orbitals and their energies, as defined by the ANO-RCC basis set⁷¹⁻⁷³. The energies are defined in the files distributed with the OpenMolcas source code⁵⁹, for convenience, the values relevant for the calculations in this work are given in the supporting information.

Initial Optimization Procedures

Once the starting orbitals are defined, an initial wave function and density matrix is set up by occupying the lowest-lying orbitals, according to the aufbau principle. Alternatively, an initial ‘‘Fermi aufbau’’ procedure is employed, in which the orbitals are assigned fractional occupations consistent with a temperature T_{F} and Fermi energy E_{F} :

$$n_i = \frac{n_{\text{max}}}{1 + \exp\left(\frac{\varepsilon_i - E_{\text{F}}}{T_{\text{F}}}\right)} \quad (44)$$

where n_{max} is 2 for restricted SCF and 1 for unrestricted SCF calculations, ε_i are the orbital energies, and E_{F} is defined in such a way that the sum of n_i matches the total number of electrons in the system, i.e. the only input parameter is T_{F} . The default initial value for T_{F} is $0.5 E_{\text{h}}$, and it is multiplied by 0.46 each iteration. When T_{F} reaches $0.01 E_{\text{h}}$ or lower and the energy change from the previous iteration is below $0.01 E_{\text{h}}$, the aufbau procedure is terminated and orbitals are assigned only 0 or n_{max} occupations. This Fermi aufbau process is enabled automatically whenever the starting orbital energies are not considered reliable enough for a straightforward aufbau occupation. In the calculations presented in this work, this happens only for the triplets (in the Triplets, Triplets Opt and TM Triplets sets) and for

the anionic singlets (cases 000 to 095 in the TM Singlets set).

The optimization method applied during the initial iterations is the “energy-DIIS” (EDIIS)⁸, which, unlike the methods described in the Theory section, is not formulated in terms of orbital rotations, but as a more straightforward SCF approach with successive construction and diagonalization of a Fock matrix⁷⁴. In EDIIS, the Fock matrix that is diagonalized at a given iteration is not the one generated by the “current” density matrix \mathbf{D}_m , but by an interpolated density matrix $\tilde{\mathbf{D}}_m$

$$\tilde{\mathbf{D}}_m = \sum_1^m c_i \mathbf{D}_m, \quad c_i \geq 0, \quad \sum_1^m c_i = 1 \quad (45)$$

The c_i coefficients are obtained by minimizing the energy function

$$E(\tilde{\mathbf{D}}_m) = \sum_1^m c_i E(\mathbf{D}_i) - \sum_{j<i} \text{Tr}((\mathbf{F}_i - \mathbf{F}_j)(\mathbf{D}_i - \mathbf{D}_j)) \quad (46)$$

where \mathbf{F}_i are the Fock matrices corresponding to the density matrices. When $\tilde{\mathbf{D}}_m$ is found, it is used for building an interpolated Fock matrix $\tilde{\mathbf{F}}_m$, and diagonalizing this yields a new set of orbitals used in the next iteration. The EDIIS optimization proceeds until the energy change is below $0.1 E_h$ and the maximum absolute value of the off-diagonal elements of $\tilde{\mathbf{D}}_m$ is smaller than 0.15.

After the EDIIS method has converged, the optimization switches to C^2 -DIIS⁴², in which the coefficients are first constrained to $\sum_1^m c_i^2 = 1$ instead of $\sum_1^m c_i = 1$ (and negative coefficients are allowed, so extrapolations are possible). In contrast to the GDIIS approach described in the Theory section, no quasi-Newton–Raphson-like step is performed after. This method continues until the off-diagonal elements of the density matrix are below 0.075, in absolute value. At that point, the final optimization method (GDIIS, r-GDIIS, RS-RFO or S-GEK/RVO, in this work) is enabled.

References

- (1) Roothaan, C. C. J. New Developments in Molecular Orbital Theory. *Rev. Mod. Phys.* **1951**, *23*, 69–89.
- (2) Kohn, W.; Sham, L. J. Self-Consistent Equations Including Exchange and Correlation Effects. *Phys. Rev.* **1965**, *140*, A1133–A1138.
- (3) Yaffe, L. G.; Goddard, W. A. Orbital optimization in electronic wave functions: equations for quadratic and cubic convergence of general multiconfiguration wave functions. *Phys. Rev. A* **1976**, *13*, 1682–1691.
- (4) Bacskay, G. B. A quadratically convergent Hartree–Fock (QC-SCF) method. Application to closed shell systems. *Chem. Phys.* **1981**, *61*, 385–404.
- (5) Pulay, P. Convergence acceleration of iterative sequences. The case of SCF iteration. *Chem. Phys. Lett.* **1980**, *73*, 393–398.
- (6) Pulay, P. Improved SCF convergence acceleration. *J. Comput. Chem.* **1982**, *3*, 556–560.
- (7) Sellers, H. ADEM-DIOS: an SCF convergence algorithm for difficult cases. *Chem. Phys. Lett.* **1991**, *180*, 461–465.
- (8) Kudin, K. N.; Scuseria, G. E.; Cancès, E. A black-box self-consistent field convergence algorithm: One step closer. *J. Chem. Phys.* **2002**, *116*, 8255–8261.
- (9) Fang, H.; Saad, Y. Two classes of multiseant methods for nonlinear acceleration. *Numer. Linear Algebra Appl.* **2009**, *16*, 197–221.
- (10) Hu, X.; Yang, W. Accelerating self-consistent field convergence with the augmented Roothaan–Hall energy function. *J. Chem. Phys.* **2010**, *132*, 054109.
- (11) Garza, A. J.; Scuseria, G. E. Comparison of self-consistent field convergence acceleration techniques. *J. Chem. Phys.* **2012**, *137*, 054110.
- (12) Pratapa, P. P.; Suryanarayana, P. Restarted Pulay mixing for efficient and robust acceleration of fixed-point iterations. *Chem. Phys. Lett.* **2015**, *635*, 69–74.

- (13) Li, H.; Yaron, D. J. A Least-Squares Commutator in the Iterative Subspace Method for Accelerating Self-Consistent Field Convergence. *J. Chem. Theory Comput.* **2016**, *12*, 5322–5332.
- (14) Hu, W.; Lin, L.; Yang, C. Projected Commutator DIIS Method for Accelerating Hybrid Functional Electronic Structure Calculations. *J. Chem. Theory Comput.* **2017**, *13*, 5458–5467.
- (15) Høst, S.; Olsen, J.; Jansík, B.; Thøgersen, L.; Jørgensen, P.; Helgaker, T. The augmented Roothaan–Hall method for optimizing Hartree–Fock and Kohn–Sham density matrices. *J. Chem. Phys.* **2008**, *129*, 124106.
- (16) Jansík, B.; Høst, S.; Johansson, M. P.; Olsen, J.; Jørgensen, P.; Helgaker, T. A stepwise atomic, valence-molecular, and full-molecular optimisation of the Hartree–Fock/Kohn–Sham energy. *Phys. Chem. Chem. Phys.* **2009**, *11*, 5805.
- (17) Wang, Y. A.; Yam, C. Y.; Chen, Y. K.; Chen, G. Communication: Linear-expansion shooting techniques for accelerating self-consistent field convergence. *J. Chem. Phys.* **2011**, *134*, 241103.
- (18) Chen, Y. K.; Wang, Y. A. LISTb: a *Better* Direct Approach to LIST. *J. Chem. Theory Comput.* **2011**, *7*, 3045–3048.
- (19) Feldmann, R.; Baiardi, A.; Reiher, M. Second-Order Self-Consistent Field Algorithms: From Classical to Quantum Nuclei. *J. Chem. Theory Comput.* **2023**, *19*, 856–873.
- (20) Thøgersen, L.; Olsen, J.; Yeager, D.; Jørgensen, P.; Sałek, P.; Helgaker, T. The trust-region self-consistent field method: Towards a black-box optimization in Hartree–Fock and Kohn–Sham theories. *J. Chem. Phys.* **2004**, *121*, 16–27.
- (21) Thøgersen, L.; Olsen, J.; Köhn, A.; Jørgensen, P.; Sałek, P.; Helgaker, T. The trust-region self-consistent field method in Kohn–Sham density-functional theory. *J. Chem. Phys.* **2005**, *123*, 074103.

- (22) Yang, C.; Meza, J. C.; Wang, L.-W. A Trust Region Direct Constrained Minimization Algorithm for the Kohn-Sham Equation. *SIAM J. Sci. Comput.* **2007**, *29*, 1854–1875.
- (23) Wen, Z.; Milzarek, A.; Ulbrich, M.; Zhang, H. Adaptive Regularized Self-Consistent Field Iteration with Exact Hessian for Electronic Structure Calculation. *SIAM J. Sci. Comput.* **2013**, *35*, A1299–A1324.
- (24) Helmich-Paris, B. A trust-region augmented Hessian implementation for restricted and unrestricted Hartree–Fock and Kohn–Sham methods. *J. Chem. Phys.* **2021**, *154*, 164104.
- (25) Kreplin, D. A.; Werner, H.-J. A combined first- and second-order optimization method for improving convergence of HartreeFock and Kohn–Sham calculations. *J. Chem. Phys.* **2022**, *156*, 214111.
- (26) Høyvik, I.-M. Convergence acceleration for the multilevel Hartree–Fock model. *Mol. Phys.* **2019**, *118*, 1626929.
- (27) Seidl, C.; Barca, G. M. J. Q-Next: A Fast, Parallel, and Diagonalization-Free Alternative to Direct Inversion of the Iterative Subspace. *J. Chem. Theory Comput.* **2022**, *18*, 4164–4176.
- (28) Császár, P.; Pulay, P. Geometry optimization by direct inversion in the iterative subspace. *J. Mol. Struct.* **1984**, *114*, 31–34.
- (29) Fischer, T. H.; Almlöf, J. General methods for geometry and wave function optimization. *J. Phys. Chem.* **1992**, *96*, 9768–9774.
- (30) Raggi, G.; Fdez. Galván, I.; Ritterhoff, C. L.; Vacher, M.; Lindh, R. Restricted-Variance Molecular Geometry Optimization Based on Gradient-Enhanced Kriging. *J. Chem. Theory Comput.* **2020**, *16*, 3989–4001.
- (31) Liu, W.; Batill, S. Gradient-Enhanced Response Surface Approximations Using Kriging Models. 9th AIAA/ISSMO Symposium on Multidisciplinary Analysis and Optimization. 2002.

- (32) Han, Z.-H.; Görtz, S.; Zimmermann, R. Improving variable-fidelity surrogate modeling via gradient-enhanced kriging and a generalized hybrid bridge function. *Aerosp. Sci. Technol.* **2013**, *25*, 177–189.
- (33) Ulaganathan, S.; Couckuyt, I.; Ferranti, F.; Laermans, E.; Dhaene, T. Performance study of multi-fidelity gradient enhanced kriging. *Struct. Multidiscip. Optim.* **2015**, *51*, 1017–1033.
- (34) Dalgaard, E.; Jørgensen, P. Optimization of orbitals for multiconfigurational reference states. *J. Chem. Phys.* **1978**, *69*, 3833–3844.
- (35) Yeager, D. L.; Jørgensen, P. Convergency studies of second and approximate second order multiconfigurational Hartree–Fock procedures. *J. Chem. Phys.* **1979**, *71*, 755–760.
- (36) Broyden, C. G. The Convergence of a Class of Double-rank Minimization Algorithms 1. General Considerations. *IMA J. Appl. Math.* **1970**, *6*, 76–90.
- (37) Fletcher, R. A new approach to variable metric algorithms. *Comput. J.* **1970**, *13*, 317–322.
- (38) Goldfarb, D. A family of variable-metric methods derived by variational means. *Math. Comput.* **1970**, *24*, 23–23.
- (39) Shanno, D. F. Conditioning of quasi-Newton methods for function minimization. *Math. Comput.* **1970**, *24*, 647–647.
- (40) Chupin, M.; Dupuy, M.-S.; Legendre, G.; Séré, É. Convergence analysis of adaptive DIIS algorithms with application to electronic ground state calculations. *ESAIM: Math. Modell. Numer. Anal.* **2021**, *55*, 2785–2825.
- (41) Anderson, D. G. Iterative Procedures for Nonlinear Integral Equations. *J. Assoc. Comput. Mach.* **1965**, *12*, 547–560.
- (42) Sellers, H. The C^2 -DIIS convergence acceleration algorithm. *Int. J. Quantum Chem.* **1993**, *45*, 31–41.

- (43) Banerjee, A.; Adams, N.; Simons, J.; Shepard, R. Search for stationary points on surfaces. *J. Phys. Chem.* **1985**, *89*, 52–57.
- (44) Besalú, E.; Bofill, J. M. On the automatic restricted-step rational-function-optimization method. *Theor. Chem. Acc.* **1998**, *100*, 265–274.
- (45) Lindh, R.; Bernhardsson, A.; Schütz, M. Force-constant weighted redundant coordinates in molecular geometry optimizations. *Chem. Phys. Lett.* **1999**, *303*, 567–575.
- (46) Lindh, R.; Bernhardsson, A.; Karlström, G.; Malmqvist, P.-Å. On the use of a Hessian model function in molecular geometry optimizations. *Chem. Phys. Lett.* **1995**, *241*, 423–428.
- (47) Davidson, E. R. The iterative calculation of a few of the lowest eigenvalues and corresponding eigenvectors of large real-symmetric matrices. *J. Comput. Phys.* **1975**, *17*, 87–94.
- (48) Slattery, S. A.; Surjuse, K.; Valeev, E. F. Economical Quasi-Newton Self Consistent Field Solver. *arXiv.org, e-Print Arch., Phys.* **2023**, 2307.00560, This content is a preprint and has not been peer-reviewed.
- (49) Nocedal, J.; Wright, S. J. *Numerical Optimization*; Springer New York, 2006.
- (50) Pinheiro, M., Jr.; Dral, P. O. *Quantum Chemistry in the Age of Machine Learning*; Elsevier, 2023; pp 205–232.
- (51) Fdez. Galván, I.; Raggi, G.; Lindh, R. Restricted-Variance Constrained, Reaction Path, and Transition State Molecular Optimizations Using Gradient-Enhanced Kriging. *J. Chem. Theory Comput.* **2021**, *17*, 571–582.
- (52) Lindh, R.; Fdez. Galván, I. *Quantum Chemistry in the Age of Machine Learning*; Elsevier, 2023; pp 391–428.
- (53) Fdez. Galván, I.; Lindh, R. Smooth Things Come in Threes: A Diabatic Surrogate Model for Conical Intersection Optimization. *J. Chem. Theory Comput.* **2023**, *19*, 3418–3427.

- (54) Ritterhoff, C. L. The Use of Kriging Within the SCF Procedure. *ChemRxiv* **2020**, doi:10.26434/chemrxiv.11950542.v1 This content is a preprint and has not been peer-reviewed.
- (55) Rasmussen, C. E.; Williams, C. K. I. *Practical Methods of Optimization*; MIT Press, 2006; Chapter 4, pp 79–104.
- (56) Stein, M. L. *Interpolation of Spatial Data*; Springer Series in Statistics 9; Springer New York, 1999.
- (57) Hermes, E. D.; Sargsyan, K.; Najm, H. N.; Zádor, J. Sella, an Open-Source Automation-Friendly Molecular Saddle Point Optimizer. *J. Chem. Theory Comput.* **2022**, *18*, 6974–6988.
- (58) Balcells, D.; Skjelstad, B. B. tmQM Dataset—Quantum Geometries and Properties of 86k Transition Metal Complexes. *J. Chem. Inf. Model.* **2020**, *60*, 6135–6146.
- (59) Li Manni, G.; Fdez. Galván, I.; Alavi, A.; Aleotti, F.; Aquilante, F.; Autschbach, J.; Avagliano, D.; Baiardi, A.; Bao, J. J.; Battaglia, S. et al. The OpenMolcas Web: A Community-Driven Approach to Advancing Computational Chemistry. *J. Chem. Theory Comput.* **2023**,
- (60) Stephens, P. J.; Devlin, F. J.; Chabalowski, C. F.; Frisch, M. J. Ab Initio Calculation of Vibrational Absorption and Circular Dichroism Spectra Using Density Functional Force Fields. *J. Phys. Chem.* **1994**, *98*, 11623–11627.
- (61) Dunning, T. H. Gaussian basis sets for use in correlated molecular calculations. I. The atoms boron through neon and hydrogen. *J. Chem. Phys.* **1989**, *90*, 1007–1023.
- (62) Woon, D. E.; Dunning, T. H. Gaussian basis sets for use in correlated molecular calculations. III. The atoms aluminum through argon. *J. Chem. Phys.* **1993**, *98*, 1358–1371.
- (63) Wilson, A. K.; Woon, D. E.; Peterson, K. A.; Dunning, T. H. Gaussian basis sets for use in correlated molecular calculations. IX. The atoms gallium through krypton. *J. Chem. Phys.* **1999**, *110*, 7667–7676.

- (64) Zobel, J. P.; Widmark, P.-O.; Veryazov, V. The ANO-R Basis Set. *J. Chem. Theory Comput.* **2020**, *16*, 278–294.
- (65) Zobel, J. P.; Widmark, P.-O.; Veryazov, V. Correction to “The ANO-R Basis Set”. *J. Chem. Theory Comput.* **2021**, *17*, 3233–3234.
- (66) Kutzelnigg, W.; Liu, W. Quasirelativistic theory equivalent to fully relativistic theory. *J. Chem. Phys.* **2005**, *123*, 241102.
- (67) Aquilante, F.; Gagliardi, L.; Pedersen, T. B.; Lindh, R. Atomic Cholesky decompositions: A route to unbiased auxiliary basis sets for density fitting approximation with tunable accuracy and efficiency. *J. Chem. Phys.* **2009**, *130*, 154107.
- (68) Aquilante, F.; Pedersen, T. B.; Lindh, R. Low-cost evaluation of the exchange Fock matrix from Cholesky and density fitting representations of the electron repulsion integrals. *J. Chem. Phys.* **2007**, *126*, 194106.
- (69) Lehtola, S.; Steigemann, C.; Oliveira, M. J. T.; Marques, M. A. L. Recent developments in libxc – A comprehensive library of functionals for density functional theory. *SoftwareX* **2018**, *7*, 1–5.
- (70) Weymuth, T.; Reiher, M. The transferability limits of static benchmarks. *Phys. Chem. Chem. Phys.* **2022**, *24*, 14692–14698.
- (71) Widmark, P.-O.; Malmqvist, P.-Å.; Roos, B. O. Density matrix averaged atomic natural orbital (ANO) basis sets for correlated molecular wave functions. *Theor. Chim. Acta* **1990**, *77*, 291–306.
- (72) Roos, B. O.; Lindh, R.; Malmqvist, P.-Å.; Veryazov, V.; Widmark, P.-O. Main Group Atoms and Dimers Studied with a New Relativistic ANO Basis Set. *J. Phys. Chem. A* **2004**, *108*, 2851–2858.
- (73) Roos, B. O.; Lindh, R.; Malmqvist, P.-Å.; Veryazov, V.; Widmark, P.-O. New Relativistic ANO Basis Sets for Transition Metal Atoms. *J. Phys. Chem. A* **2005**, *109*, 6575–6579.

(74) Lehtola, S.; Blockhuys, F.; Van Alsenoy, C. An Overview of Self-Consistent Field Calculations Within Finite Basis Sets. *Molecules* **2020**, *25*, 1218.

TOC Graphic

