Linear-scaling symmetric square-root decomposition of the overlap matrix

Branislav Jansík, Stinne Høst, Poul Jørgensen, and Jeppe Olsen
Lundbeck Foundation Center for Theoretical Chemistry, University of Aarhus, DK-8000 Århus C, Denmark and Department of Chemistry, University of Aarhus, DK-8000 Århus C, Denmark

Trygve Helgaker[a)]
Department of Chemistry, University of Durham, South Road, Durham DH1 3LE, United Kingdom

(Received 1 December 2006; accepted 24 January 2007; published online 23 March 2007)

We present a robust linear-scaling algorithm to compute the symmetric square-root or Löwdin decomposition of the atomic-orbital overlap matrix. The method is based on Newton-Schulz iterations with a new approach to starting matrices. Calculations on 12 chemically and structurally diverse molecules demonstrate the efficiency and reliability of the method. Furthermore, the calculations show that linear scaling is achieved. © *2007 American Institute of Physics*.
[DOI: 10.1063/1.2709881]

I. INTRODUCTION

An electronic-structure calculation begins by choosing an atomic-orbital (AO) basis followed by the determination of an approximate solution to the electronic Schrödinger equation in the selected basis. To obtain the approximate solution, nonlinear, linear, or (generalized) eigenvalue equations have to be solved. The size of these equations generally impose that iterative techniques must be used. To improve the convergence of the iterative algorithms, a transformation from the AO basis to the molecular-orbital (MO) basis is usually performed. This transformation reduces the condition number of the carrier matrix and makes it diagonally dominant (in general with orbital energy differences on the diagonal), making the preconditioning of the equations simple and efficient.

However, a severe disadvantage of MOs for large molecular systems is their nonlocality, making it difficult to reduce the high scaling of the standard electronic-structure models for such systems and, in particular, to achieve linear scaling—the goal of much research in the electronic-structure theory in recent years. The key to such a reduced scaling is to use a basis that preserves the locality of the AO basis, keeping in mind that we should also be able to solve the equations efficiently. A good compromise has become to use an orthogonalized AO (OAO) basis, since the condition number of the carrier matrix is then the same as in the MO basis. The most commonly used OAO basis is the Cholesky basis, pioneered by Millam and Scuseria[1] in their density functional theory Kohn-Sham calculations and also used by Shao *et al.*[2] in their curvy-step method. An additional advantage of the Cholesky basis is that it retains to some extent the diagonal dominance of the carrier matrix in the MO basis.

In principle, however, an infinite number of OAO bases exist. Given that the locality of the OAO basis is important for reducing the scaling of electronic-structure models, it is advantageous to use that particular OAO basis which has the largest similarity with the AO basis. Carlson and Keller[3] have shown that the symmetric square-root or Löwdin basis is the OAO basis that, in a least-squares sense, has the largest similarity with the AO basis. Furthermore, the carrier matrix is as diagonally dominant in the Löwdin basis as in the Cholesky basis. Sałek *et al.*[4] have therefore used this basis in their linear-scaling Hartree-Fock and Kohn-Sham optimization algorithms. Coriani *et al.*[5] also used this basis to obtain linear-scaling molecular response properties in Hartree-Fock and Kohn-Sham theories, as did Manninen *et al.*[6] to obtain second-order Møller-Plesset energies. However, a prerequisite for using the Löwdin basis is that it may be obtained using an algorithm that is efficient and numerically stable. Furthermore, for the algorithm to be useful, it must be of linear complexity.

One such algorithm are the Newton-Schulz iterations,[7–9] which have the attractive features of being *m*th-order convergent and based exclusively on matrix-matrix operations. Linear scaling may thus be obtained if matrix sparsity is exploited. The applicability, however, is restricted since global convergence is not inherent to the Newton-Schulz iterations. It is the purpose of this paper to demonstrate how accelerated global convergence and linear scaling may be achieved using a modified Newton-Schulz algorithm.

The remainder of this paper contains three sections. First, in Sec. II, we develop a globally convergent iterative algorithm for the evaluation of the matrix square root of a positive definite symmetric matrix such as the AO overlap matrix. Next, in Sec. III, numerical examples are given to demonstrate the efficiency and stability of the algorithm and, in particular, that linear scaling with system size can be achieved. Section IV contains some concluding remarks.

II. THE ITERATIVE CALCULATION OF THE MATRIX SQUARE ROOT

The Löwdin decomposition of the overlap matrix $\mathbf{S}$ is defined as

$$\mathbf{S} = \mathbf{S}^{1/2}\mathbf{S}^{1/2}, \tag{1}$$

[a)]Permanent address: Department of Chemistry, University of Oslo, P.O.B. 1033 Blindern, N-0315 Oslo, Norway.

$$\mathbf{I} = \mathbf{S}^{1/2}\mathbf{S}^{-1/2}, \tag{2}$$

where $\mathbf{I}$ is the identity matrix. The traditional way to find $\mathbf{S}^{1/2}$ and $\mathbf{S}^{-1/2}$ is to compute the eigenvalues and eigenvectors of the symmetric positive definite overlap matrix,

$$\mathbf{SV} = \mathbf{V}\boldsymbol{\epsilon}, \tag{3}$$

and then to determine $\mathbf{S}^{1/2}$ and $\mathbf{S}^{-1/2}$ as

$$\mathbf{S}^{1/2} = \mathbf{V}\boldsymbol{\epsilon}^{1/2}\mathbf{V}^{\dagger}, \tag{4}$$

$$\mathbf{S}^{-1/2} = \mathbf{V}\boldsymbol{\epsilon}^{-1/2}\mathbf{V}^{\dagger}. \tag{5}$$

The disadvantage of this approach for large systems is the cubic scaling associated with the diagonalization. Alternatively, $\mathbf{S}^{1/2}$ and $\mathbf{S}^{-1/2}$ may be obtained from Newton-Schulz iterations.[7–9] However, these iterations do not converge for all $\mathbf{S}$. We demonstrate here how $\mathbf{S}$ may be scaled to guarantee convergence. The resulting algorithm is matrix-matrix multiplication driven and linear scaling is achieved when sparse-matrix algebra is used.

A. Newton-Schulz iterations for the inverse matrix square root

To set up a Newton-Schulz scheme for the evaluation of $\mathbf{S}^{-1/2}$, assume that, at the kth iteration, we have generated an approximate inverse matrix square root $\mathbf{Z}_k$ such that

$$\mathbf{Z_k}\mathbf{S}\mathbf{Z}_k = \mathbf{X}_k \approx \mathbf{I}. \tag{6}$$

We further assume that $\mathbf{Z}_k$ commutes with $\mathbf{S}$. We would now like to find a new matrix $\boldsymbol{\zeta}_k$ such that

$$\boldsymbol{\zeta}_k\mathbf{X}_k\boldsymbol{\zeta}_k = \mathbf{I}. \tag{7}$$

Introducing

$$\boldsymbol{\rho}_k = \mathbf{I} - \mathbf{X}_k, \tag{8}$$

we note that $\boldsymbol{\zeta}_k$ may be written in the following form:

$$\boldsymbol{\zeta}_k = (\mathbf{I} - \boldsymbol{\rho}_k)^{-1/2}. \tag{9}$$

We may now write the inverse square root as

$$\mathbf{Z} = \mathbf{Z}_k(\mathbf{I} - \boldsymbol{\rho}_k)^{-1/2}, \tag{10}$$

which is easily seen to satisfy $\mathbf{ZSZ}=\mathbf{I}$ (since $\boldsymbol{\rho}_k$ and $\mathbf{Z}_k$ commute).

The expression Eq. (10) allows us to calculate the exact inverse square root $\mathbf{Z}$ from an approximate inverse square root $\mathbf{Z}_k$. However, it is not immediately useful since it still contains an inverse matrix square root. To avoid the square root, we introduce the following Taylor expansion:

$$(\mathbf{I} - \boldsymbol{\rho}_k)^{-1/2} = \mathbf{I} + \tfrac{1}{2}\boldsymbol{\rho}_k + \tfrac{3}{8}\boldsymbol{\rho}_k^2 + \tfrac{5}{16}\boldsymbol{\rho}_k^3 + \tfrac{35}{128}\boldsymbol{\rho}_k^4 + \cdots, \tag{11}$$

which converges provided that all eigenvalues of $\boldsymbol{\rho}_k$ are smaller than one in absolute value, which is equivalent to requiring that the matrix 2-norm[10] of $\boldsymbol{\rho}_k$ is smaller than one:

$$\|\boldsymbol{\rho}_k\|_2 < 1. \tag{12}$$

Truncating the Taylor expansion at different orders in $\boldsymbol{\rho}_k$, we obtain

$$\mathbf{T}_k^2 = \tfrac{1}{2}(3\mathbf{I} - \mathbf{X}_k), \tag{13}$$

$$\mathbf{T}_k^{(3)} = \tfrac{1}{8}(15\mathbf{I} - 10\mathbf{X}_k + 3\mathbf{X}_k^2), \tag{14}$$

$$\mathbf{T}_k^{(4)} = \tfrac{1}{16}(35\mathbf{I} - 35\mathbf{X}_k + 21\mathbf{X}_k^2 - 5\mathbf{X}_k^3), \tag{15}$$

$$\mathbf{T}_k^{(5)} = \tfrac{1}{128}(315\mathbf{I} - 420\mathbf{X}_k + 378\mathbf{X}_k^2 - 180\mathbf{X}_k^3 + 35\mathbf{X}_k^4). \tag{16}$$

These expressions may be used to set up Newton-Schulz iterations of different orders m for the inverse square-root matrix:

$$\begin{aligned} &\mathbf{Z}_0 = \mathbf{I}, \quad \mathbf{X}_k = \mathbf{Z}_k\mathbf{S}\mathbf{Z}_k, \quad \mathbf{Z}_{k+1} = \mathbf{Z}_k\mathbf{T}_k^{(m)}, \\ &\mathbf{S}^{-1/2} = \lim_{k\to\infty}\mathbf{Z}_k, \quad \text{if } \|\mathbf{S} - \mathbf{I}\|_2 < 1, \end{aligned} \tag{17}$$

where $\mathbf{Z}_k$ commutes with $\mathbf{S}$, as is seen by induction, noting that $\mathbf{Z}_0=\mathbf{I}$ commutes with $\mathbf{S}$. Once we have generated $\mathbf{S}^{-1/2}$ in this manner, we may also obtain $\mathbf{S}^{-1}$ by squaring and $\mathbf{S}^{1/2}$ by multiplication with $\mathbf{S}$. In a slightly different manner, these iterations were first derived by Niklasson.[9]

In the local region, the iteration Eq. (17) exhibits mth-order convergence, with global convergence guaranteed only for $\|\mathbf{S}-\mathbf{I}\|_2<1$, severely restricting its applicability. To converge, we may either try a starting guess $\mathbf{Z}_0 \neq \mathbf{I}$ such that $\|\mathbf{Z}_0\mathbf{S}\mathbf{Z}_0-\mathbf{I}\|_2<1$ or we may apply the scaling scheme described in Sec. II D and Sec. II F. We also note that the convergence constraints are in reality more relaxed, as discussed in Sec. II E.

B. Stabilized Newton-Schulz iterations

In exact algebra, the $\mathbf{Z}_k$ in Eq. (17) are symmetric and commute with $\mathbf{S}$, as seen by induction. In finite-precision algebra, round-off errors occur in $\mathbf{Z}_k$, leading to a loss of symmetry and noncommutation in the course of the iterations. Let us write the kth iterate in the following form:

$$\mathbf{Z}_k = \mathbf{S}^{-1/2} + \boldsymbol{\Delta}_k, \tag{18}$$

$$\boldsymbol{\Delta}_k = \boldsymbol{\Delta}_k^{\parallel} + \boldsymbol{\Delta}_k^{\perp}, \tag{19}$$

where $\boldsymbol{\Delta}_k^{\parallel}$ is a symmetric matrix that may be expressed as a power series in $\mathbf{S}$ and therefore commutes with $\mathbf{S}$, while $\boldsymbol{\Delta}_k^{\perp}$ is the noncommuting part of the error,

$$[\mathbf{S}, \boldsymbol{\Delta}_k^{\parallel}] = 0, \quad [\mathbf{S}, \boldsymbol{\Delta}_k^{\perp}] \neq 0. \tag{20}$$

Inserting Eq. (18) in Eq. (17) and rearranging, we obtain for $m=2$ the following error in iteration $k+1$:

$$\begin{aligned} \boldsymbol{\Delta}_{k+1} &= \tfrac{1}{2}\boldsymbol{\Delta}_k - \tfrac{1}{2}\mathbf{S}^{-1/2}\boldsymbol{\Delta}_k\mathbf{S}^{1/2} + \mathcal{O}(\boldsymbol{\Delta}_k^2) \\ &= \tfrac{1}{2}\boldsymbol{\Delta}_k^{\perp} - \tfrac{1}{2}\mathbf{S}^{-1/2}\boldsymbol{\Delta}_k^{\perp}\mathbf{S}^{1/2} + \mathcal{O}(\boldsymbol{\Delta}_k^2). \end{aligned} \tag{21}$$

In infinite precision, $\boldsymbol{\Delta}_k^{\perp}=0$ at each iteration (assuming $\mathbf{Z}_0=\mathbf{I}$) and quadratic convergence is observed. By contrast, in finite precision, a nonvanishing $\boldsymbol{\Delta}_k^{\perp}$ may be introduced in the course of the iterations. Since Eq. (21) is linear in $\boldsymbol{\Delta}_k^{\perp}$, this error is not removed by the iterations but may on the contrary increase with k, potentially introducing a large error in the solution.

However, it is possible to modify the iterations slightly so as to stabilize $\boldsymbol{\Delta}_k^{\perp}$ close to convergence. Replacing $\mathbf{X}_k$

$=\mathbf{Z}_k\mathbf{S}\mathbf{Z}_k$ in Eq. (17) by the manifestly symmetric expression $\mathbf{X}_k=\mathbf{Z}_k^\dagger\mathbf{S}\mathbf{Z}_k$,[9] we obtain the stabilized Newton-Schulz iterations,

$$\mathbf{Z}_0=\mathbf{I},\quad \mathbf{X}_k=\mathbf{Z}_k^\dagger\mathbf{S}\mathbf{Z}_k,\quad \mathbf{Z}_{k+1}=\mathbf{Z}_k\mathbf{T}_k^{(m)},$$
$$\mathbf{S}^{-1/2}=\lim_{k\to\infty}\mathbf{Z}_k,\quad \text{if } \|\mathbf{S}-\mathbf{I}\|_2<1, \tag{22}$$

which, in infinite precision, converge to the same solution as Eq. (17). In finite precision, the error matrix corresponding to Eq. (21) is now given by

$$\begin{aligned}\mathbf{\Delta}_{k+1}&=\tfrac{1}{2}\mathbf{\Delta}_k^\perp-\tfrac{1}{2}\mathbf{S}^{-1/2}(\mathbf{\Delta}_k^\perp)^\dagger\mathbf{S}^{1/2}+\mathcal{O}(\mathbf{\Delta}_k^2)\\&=\mathbf{\Delta}_k+\mathcal{O}(\mathbf{\Delta}_k^2)+\mathcal{O}(\mathbf{\Delta}_{k-1}^2),\end{aligned} \tag{23}$$

where the second expression follows from the first one applied twice. Sufficiently close to convergence, therefore, the error $\mathbf{\Delta}_k^\perp$ stabilizes and error accumulation is avoided in finite precision. No such result holds for the original iterations [Eq. (17)], as can easily be confirmed by applying Eq. (21) twice.

C. Coupled Newton-Schulz iterations

An alternative way to stabilize convergence is to rearrange Eq. (17) into a pair of coupled iterations, involving $\mathbf{Y}=\mathbf{S}\mathbf{Z}=\mathbf{Z}\mathbf{S}$ in addition to $\mathbf{Z}$. The iteration for $\mathbf{Y}$ is obtained by left multiplication of Eq. (17) by $\mathbf{S}$. The resulting coupled Newton-Schulz iterations of different orders m for the matrix square root and its inverse then take the following form:

$$\mathbf{Z}_0=\mathbf{I},\quad \mathbf{Y}_0=\mathbf{S},$$
$$\mathbf{X}_k=\mathbf{Y}_k\mathbf{Z}_k,\quad \mathbf{Z}_{k+1}=\mathbf{Z}_k\mathbf{T}_k^{(m)},\quad \mathbf{Y}_{k+1}=\mathbf{T}_k^{(m)}\mathbf{Y}_k,$$
$$\mathbf{S}^{-1/2}=\lim_{k\to\infty}\mathbf{Z}_k,\quad \mathbf{S}^{1/2}=\lim_{k\to\infty}\mathbf{Y}_k,\quad \text{if } \|\mathbf{S}-\mathbf{I}\|_2<1. \tag{24}$$

These iterations are formally identical to Eq. (17) but error propagation is controlled in the same manner as in Eq. (23). We note that the number of matrix multiplications involved in the coupled Newton-Schulz iterations [Eq. (24)] is the same as for the iterations of the form of Eq. (17).

For $m=2$, the iterations [Eq. (24)] are identical to the coupled Newton-Schulz iterations for the matrix square root, obtained via the matrix sign function, as described by Higham[7] and references therein.

D. Global convergence by scaling

The Newton-Schulz iterations that have been developed above are guaranteed to converge only for $\|\mathbf{S}-\mathbf{I}\|_2<1$. In cases where $\|\mathbf{S}-\mathbf{I}\|_2\geqslant 1$, we may obtain a convergent series by applying the iterations to the scaled matrix $\lambda\mathbf{S}$ where λ is chosen such that

$$\|\lambda\mathbf{S}-\mathbf{I}\|_2<1. \tag{25}$$

From Eq. (24), we then obtain the following iterations for $\mathbf{S}^{-1/2}$ and $\mathbf{S}^{1/2}$:

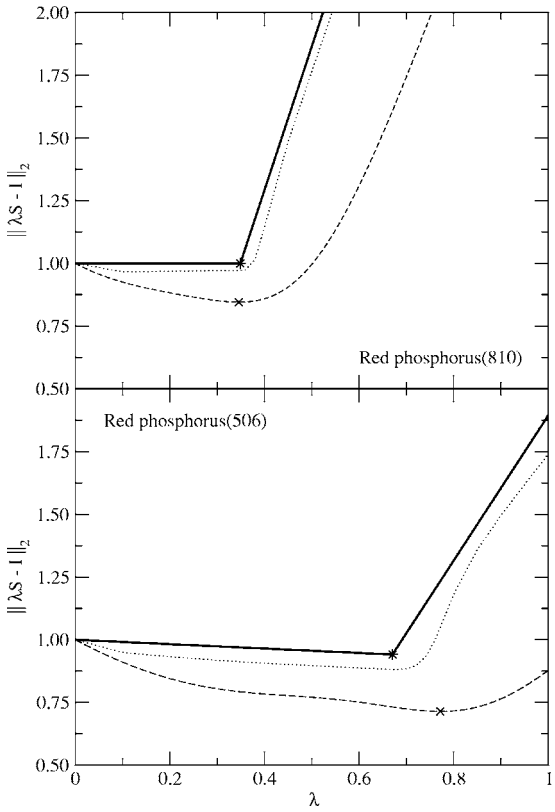

FIG. 1. The 2-norm $\|\lambda\mathbf{S}-\mathbf{I}\|_2$ (full line) and the approximation in Eq. (30) to the 2-norm for $n=1$ (dashed line) and $n=16$ (dotted line) are plotted as a function of the scaling parameter λ. Upper panel: Red phosphorus (810). Lower panel: Red phosphorus (506).

$$\mathbf{Z}_0=\mathbf{I},\quad \mathbf{Y}_0=\mathbf{S},$$
$$\mathbf{X}_k=\lambda\mathbf{Y}_k\mathbf{Z}_k,\quad \mathbf{Z}_{k+1}=\mathbf{Z}_k\mathbf{T}_k^{(m)},\quad \mathbf{Y}_{k+1}=\mathbf{T}_k^{(m)}\mathbf{Y}_k,$$
$$\mathbf{S}^{-1/2}=\lambda^{1/2}\lim_{k\to\infty}\mathbf{Z}_k,\quad \mathbf{S}^{1/2}=\lambda^{-1/2}\lim_{k\to\infty}\mathbf{Y}_k,$$
$$\text{for } \|\lambda\mathbf{S}-\mathbf{I}\|_2<1. \tag{26}$$

The convergence of the iterations depends on the magnitude of $\|\lambda\mathbf{S}-\mathbf{I}\|_2$. We shall now determine the minimizer of $\|\lambda\mathbf{S}-\mathbf{I}\|_2$ and show that, for this minimizer λ_*, Eq. (25) is satisfied. The 2-norm in Eq. (25) may be expressed as

$$\begin{aligned}\|\lambda\mathbf{S}-\mathbf{I}\|_2&=\|\mathbf{V}(\lambda\boldsymbol{\epsilon}-\mathbf{I})\mathbf{V}^\dagger\|_2\\&=\|\lambda\boldsymbol{\epsilon}-\mathbf{I}\|_2=\max(\mathrm{abs}(\lambda\epsilon_i-1)),\end{aligned} \tag{27}$$

where the ϵ_i are the eigenvalues of $\mathbf{S}$. In Fig. 1, we have plotted $\|\lambda\mathbf{S}-\mathbf{I}\|_2$ (the full line) as a function of λ in calculations on red phosphorus (see below for a more detailed description of the molecules). For large λ, the function has the form $\lambda\epsilon_{\max}-1$ and for small λ it becomes $1-\lambda\epsilon_{\min}$, where $\epsilon_{\max}$ and $\epsilon_{\min}$ are the largest and smallest eigenvalues, respectively, of the overlap matrix $\mathbf{S}$. The optimal value λ_* is obtained at the intersection (marked with an asterisk) between the two lines $\lambda_*\epsilon_{\max}-1=1-\lambda_*\epsilon_{\min}$, yielding

$$\lambda_* = \frac{2}{\epsilon_{\max} + \epsilon_{\min}}. \tag{28}$$

At the minimizer λ_*, the 2-norm $\|\lambda\mathbf{S}-\mathbf{I}\|_2$ is obviously smaller than 1 as it is for all λ smaller than λ_*. The Newton-Schulz iteration is therefore guaranteed to converge when λ_* is used.

The determination of λ_* using the above procedure requires that the smallest and largest eigenvalues of the overlap matrix are determined. This may be done using an iterative algorithm. However, we here choose to determine an approximation to λ_*, where we approximate the 2-norm in terms of the Frobenius norm as

$$\|\lambda\mathbf{S}-\mathbf{I}\|_2 \gtrapprox \frac{\|(\lambda\mathbf{S}-\mathbf{I})^2\|_F}{\|\lambda\mathbf{S}-\mathbf{I}\|_F} = \sqrt{\frac{\lambda^4 \operatorname{Tr}(\mathbf{S}^4) - 4\lambda^3 \operatorname{Tr}(\mathbf{S}^3) + 6\lambda^2 \operatorname{Tr}(\mathbf{S}^2) - 4\lambda \operatorname{Tr}(\mathbf{S}) + \operatorname{Tr}(\mathbf{I})}{\lambda^2 \operatorname{Tr}(\mathbf{S}^2) - 2\lambda \operatorname{Tr}(\mathbf{S}) + \operatorname{Tr}(\mathbf{I})}}, \tag{29}$$

as explained below. The evaluation of the right-hand side of this expression is straightforward, requiring only the evaluation of $\mathbf{S}^2$ and some trace operations. It yields a good approximation to the 2-norm from below, as indicated by the $\gtrapprox$ sign. Equation (29) is plotted as a function of λ in Fig. 1 (the dashed line). The minimization of the right-hand side of Eq. (29) is a simple one-parameter minimization problem of a smooth function. The value obtained by this minimization λ_{app} is marked with a cross in Fig. 1.

The use of λ_{app} may lead to a 2-norm that is greater than one and the iterative algorithm may then diverge. If this happens, we know that $\lambda_{\text{app}} > \lambda_*$ and simply decrease λ_{app} by a factor of 0.9. This procedure can be repeated if the series still diverges.

We shall now justify the approximation to the 2-norm made in Eq. (29). An approximation to the 2-norm of a Hermitian matrix $\mathbf{A}$ may be obtained from

$$\|\mathbf{A}\|_2 = \lim_{n\to\infty} \frac{\|\mathbf{A}^{n+1}\|_F}{\|\mathbf{A}^n\|_F} \gtrapprox \frac{\|\mathbf{A}^2\|_F}{\|\mathbf{A}\|_F}, \tag{30}$$

where the Frobenius norm is defined as

$$\|\mathbf{A}\|_F^2 = \operatorname{Tr}(\mathbf{A}^\dagger\mathbf{A}) = \sum_i \eta_i^2, \tag{31}$$

where η_i is an eigenvalue of $\mathbf{A}$. To verify Eq. (30), we introduce $\mathbf{B}=\mathbf{A}/\|\mathbf{A}\|_2$ and take its Frobenius norm,

$$\|\mathbf{B}^n\|_F = \frac{\|\mathbf{A}^n\|_F}{\|\mathbf{A}\|_2} = \left(\frac{\sum \eta_i^{2n}}{\eta_{\max}^{2n}}\right)^{1/2}, \tag{32}$$

where $\eta_{\max}$ is the numerically largest eigenvalue of $\mathbf{A}$. For large n, we have

$$\lim_{n\to\infty} \|\mathbf{B}^n\|_F = \sqrt{\mu_{\max}}, \tag{33}$$

where $\mu_{\max}$ is the multiplicity of $\eta_{\max}$. Using Eqs. (32) and (33), we then find that

$$\lim_{n\to\infty} \frac{\|\mathbf{B}^{n+1}\|_F}{\|\mathbf{B}^n\|_F} = \lim_{n\to\infty} \frac{\|\mathbf{A}^{n+1}\|_F}{\|\mathbf{A}^n\|_F\|\mathbf{A}\|_2} = 1, \tag{34}$$

which proves that Eq. (30) is valid.

Many other approximations have been suggested to the 2-norm. In the context of Newton-Schulz iterations, Németh and Scuseria[11] have proposed to use the Geshgorin circular theorem (GCT), which for a Hermitian matrix reads

$$\|\mathbf{A}\|_2 \approx \max_i \left(A_{ii} + \sum_{i\neq j} |A_{ij}|\right). \tag{35}$$

Furthermore, a variety of iterative estimates such as the power method are available.[10] In our numerical section, we shall discuss the number of iterations required with exact and approximate 2-norms.

E. Monotonically convergent Newton-Schulz iterations

Inspired by the work of Németh and Scuseria,[11] we note that we may take an alternative approach to determine the scaling parameter λ. Instead of focusing on the convergence of expansion Eq. (11), we concentrate on the mapping properties of the Newton-Schulz iterations for the $\mathbf{X}_k$ matrices. We may consider these matrices as represented in the basis of their eigenvectors. The mapping functions $f^{(m)}$ for the mth-order iteration show how the eigenvalues of $\mathbf{X}_k$ matrix are mapped to the eigenvalues of $\mathbf{X}_{k+1}$ in an iteration step. Without scaling, the eigenvalues of $\mathbf{X}_0$ are equal to those of $\mathbf{S}$, while the converged $\mathbf{X}_k$ becomes the unit matrix with all eigenvalues equal to 1. The mapping functions for the eigenvalues of $\mathbf{X}_k$

$$f^{(2)}(\epsilon) = \tfrac{1}{4}\epsilon(-3+\epsilon)^2, \tag{36}$$

$$f^{(3)}(\epsilon) = \tfrac{1}{64}\epsilon(15-10\epsilon+3\epsilon^2)^2, \tag{37}$$

$$f^{(4)}(\epsilon) = \tfrac{1}{256}\epsilon(35-35\epsilon+21\epsilon^2-5\epsilon^3)^2, \tag{38}$$

$$f^{(5)}(\epsilon) = \tfrac{1}{16\,384}\epsilon(315-420\epsilon+378\epsilon^2-180\epsilon^3+35\epsilon^4)^2 \tag{39}$$

allow us to observe how the eigenvalues of $\mathbf{X}_0=\mathbf{S}$ are gradually transformed to unity in the course of the iterations. These functions were obtained in a straightforward manner from an expression that may be derived for $\mathbf{X}_{k+1}$. In Fig. 2, we have plotted the mapping functions $f^{(2)}(\epsilon)$ and $f^{(3)}(\epsilon)$, noting that $f^{(4)}(\epsilon)$ closely resembles $f^{(2)}(\epsilon)$, while $f^{(5)}(\epsilon)$ shares the profile of $f^{(3)}(\epsilon)$. From an analysis of these functions and the requirement that the eigenvalues of $\mathbf{X}_k$ con-

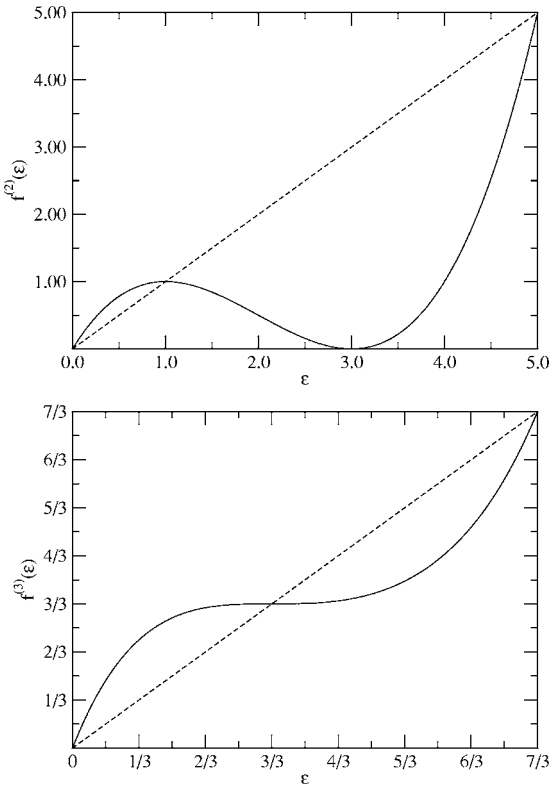

FIG. 2. Eigenvalue-mapping functions in the convergent range. Upper panel: The eigenvalue-mapping function $f^{(2)}(\epsilon)$. Lower panel: The eigenvalue-mapping function $f^{(3)}(\epsilon)$.

verge to 1, the global convergence intervals of the Newton-Schulz iterations can be readily understood—see Table I. All functions provide a one-to-one mapping in the eigenvalue range [0, 1]. Provided the eigenvalues of $\mathbf{X}_k$ are in that range, the ordered sequence of eigenvectors does not change as the iterations proceed. This feature is called monotonic convergence.[11] For $m=3$ and $m=5$, monotonic convergence is not restricted to the interval (0, 1] but occurs in the entire convergent range of eigenvalues $(0, 7/3)$ and $(0, \approx 2.23)$ (see Table I and Fig. 2).

For $m=2$ we have for ϵ in the interval (1, 5) a decrease in the eigenvalues in each iteration, leading eventually to eigenvalues smaller than one and monotonic convergence. The exception is if an eigenvalue of 3 is accidentally obtained in which case this eigenvalue is mapped to a zero eigenvalue leading to a divergent series. Convergence is in any case guaranteed if the eigenvalues of $\mathbf{X}_0$ are in the range (0, 1). Setting

TABLE I. Convergent intervals of ϵ for mth-order Newton-Schulz iterations. First column: convergence is guaranteed within the interval. Second column: convergence is possible, but not guaranteed within the interval. No convergence is possible beyond the interval.

m	Guaranteed	Possible
2	(0, 3)	(3, 5)
3	(0, 7/3)	⋯
4	(0, ≈2.53)	(≈2.53, 3)
5	(0, ≈2.23)	⋯

$$\lambda = \|\mathbf{S}\|_2^{-1}, \tag{40}$$

this requirement is satisfied.

F. Intermediate scaling

The scaling procedure described in Sec. II D provides global convergence by ensuring that the expansion Eq. (11) is convergent for the $\boldsymbol{\rho}_0$ starting matrix and therefore also for $\boldsymbol{\rho}_k$ in each iteration. When the minimizer λ_* is used, the convergence of Eq. (11) is optimal for $\boldsymbol{\rho}_0$. We may require the convergence to be optimal also for each iteration k and thus for each $\boldsymbol{\rho}_k$. To satisfy this requirement, we determine a new λ_k in each iteration such that

$$\|\lambda_k \mathbf{X}_k - \mathbf{I}\|_2 = \min. \tag{41}$$

Such a scaling of $\mathbf{X}_k$ results in an optimal convergence of the expansion Eq. (11) at each iteration, for the reasons discussed in Sec. II D. With intermediate scaling, the coupled Newton-Schulz iterations take the following form:

$$\mathbf{Z}_0 = \mathbf{I}, \quad \mathbf{Y}_0 = \mathbf{S},$$

$$\mathbf{X}_k = \lambda_k \mathbf{Y}_k \mathbf{Z}_k, \quad \mathbf{Z}_{k+1} = \lambda_k^{1/2} \mathbf{Z}_k \mathbf{T}_k^{(m)}, \quad \mathbf{Y}_{k+1} = \lambda_k^{1/2} \mathbf{T}_k^{(m)} \mathbf{Y}_k,$$

$$\mathbf{S}^{-1/2} = \lim_{k\to\infty} \mathbf{Z}_k, \quad \mathbf{S}^{1/2} = \lim_{k\to\infty} \mathbf{Y}_k. \tag{42}$$

The minimizer λ_k may be evaluated as described in Sec. II D,

$$\lambda_k = \frac{2}{\epsilon_{\max}^k + \epsilon_{\min}^k}, \tag{43}$$

where ϵ_k are extremal eigenvalues of $\mathbf{Y}_k \mathbf{Z}_k$. In this context, it is advantageous to work with monotonically convergent iterations ($m=3$ and $m=5$) as the initial $\epsilon_{\max}$ and $\epsilon_{\min}$ may then safely be updated using the following mapping functions:

$$\epsilon_{(k+1)} = f^{(m)}(\lambda_k \epsilon_k). \tag{44}$$

G. Computational scaling of the algorithm

The above algorithm consists of a sequence of Newton-Schulz iterations of a given order m. Each Newton-Schulz iteration requires $m+1$ matrix-matrix multiplications, m scaling operations, and m matrix-matrix additions. One dot product operation is required to check for convergence.

The determination of the scaling parameter λ requires an effort that depends on the method chosen. The evaluation of λ_{app} requires one matrix-matrix multiplication and three dot products. The determination of λ_* requires $\epsilon_{\max}$ and $\epsilon_{\min}$. This typically requires 5-30 matrix-vector multiplications, depending on the chosen iterative method and the desired precision. If intermediate scaling is used with monotonic convergence, λ_k must be reevaluated at each iteration. The entire algorithm is thus exclusively based on matrix-matrix operations.

For sufficiently large systems, the matrices become sparse with a linear growth in the number of nonzero elements with system size. For large systems the use of sparse-

matrix algebra will therefore cause the computational time to scale linearly with the system size. More detailed discussion of this topic may be found in Ref. 12.

III. NUMERICAL ILLUSTRATIONS

A. Sample molecules

To illustrate the convergence and stability of the iteration method for the matrix square root and its inverse developed in Sec. II, we here report calculations on 12 structurally different molecules. These fall into four categories (the number in parentheses indicates the number of atoms):

- Organic molecules: 1. fullerene C_{60} (60), 2. polysaccharide (438), a two-dimensional network structure, 3. polysaccharide (875), a two-dimensional network structure, 4. DNA fragment (583), one loop of a double helix, containing adenine thymine and cytosine guanine base pairs, and 5. PRC (1082), a model of a photosystem reaction center found in green plants;
- inorganic molecules: 6. red phosphorus (506), a one-dimensional chain, 7. red phosphorus (810), nine one-dimensional chains in a three-dimensional packing, 8. α crystobalite (948), a three-dimensional network structure of quartz, and 9. β tridymite (1314), another three-dimensional network structure of quartz;
- transition-metal complexes: 10. Rh complex (25), described in Ref. 13, and 11. InAg complex (176), a $In_{18}Ag_{26}Cl_6C_{10}S_{36}P_{20}H_{60}$ model of a nanosized metal cluster;[14]
- random molecule: 12. random molecule (64), molecule generated by 64 random points in space restricted to $10\times10\times10$ Å with limits to how close the individual atoms can be together. Random charges corresponding to elements of the first four periods were assigned to the points.

We used the $6-31G$ basis for the organic and inorganic molecules. For the Rh complex, we used the STO-3G basis for rhodium and the AhlrichsVDZ basis[15] for the remaining atoms; for the InAg complex, we used the $3-21G$ basis augmented with the Huzinaga polarization functions for In and Ag and the $6-31G$ basis for the other atoms. For the random molecule, we used the $3-21G$ basis.

B. Stability and performance

To test the performance of the scaled Newton-Schulz algorithm, we have carried out second- and third-order calculations ($m=2$ and $m=3$, respectively), with and without intermediate scaling. In the calculations without intermediate scaling, we have used both exact λ_* and approximative λ_{app} and $\lambda_{\text{GCT}}=2\|\mathbf{S}\|_2^{-1}$, with the 2-norm approximated by Eq. (35). All iterations were carried out to a 10^{-10} precision. In the calculations with intermediate scaling, we have used iterative methods to compute the highest and lowest eigenvalues required to obtain λ_k. For $m=2$, the iterative estimate to the precision of 10^{-5} has been recomputed in each iteration; for $m=3$, the mapping function [Eq. (37)] has been used to obtain the update on λ_k [Eq. (44)]. The results are summarized in Table II, where, for each molecule, we report $\|\mathbf{S}-\mathbf{I}\|_2$, λ_*, λ_{app} and λ_{GCT} together with $\|\lambda_*\mathbf{S}-\mathbf{I}\|_2$, $\|\lambda_{\text{app}}\mathbf{S}-\mathbf{I}\|_2$, $\|\lambda_{\text{GCT}}\mathbf{S}-\mathbf{I}\|_2$, and the number of Newton-Schulz iterations $N^{(m)}$, denoting intermediate scaling by subscript int. The $\|\mathbf{S}-\mathbf{I}\|_2$ norm is typically 5 ± 1, with 10 for the InAg complex as the extreme among the real molecular systems. Scaling by λ has to be employed in all cases to make the Newton-Schulz iterations converge.

The λ_{app} values compare favorably with the optimal λ_* values. Typically, the differences are only 0.03 ± 0.02, with the largest difference of 0.1015 occurring for red phosphorus (506). The good agreement can be understood from Fig. 1, where we have plotted $\|\lambda\mathbf{S}-\mathbf{I}\|_2$ (the full line) as a function of λ along with Frobenius approximations to $\|\lambda\mathbf{S}-\mathbf{I}\|_2$, obtained using Eq. (30) with $n=1$ (dashed line) and $n=16$ (dotted line). Even though the dashed line is not a good approximation to the full line, the minimum λ_* (marked with a star) is close to λ_{app} (marked with a cross). This appears to be a general behavior, occurring for all overlap matrices that we have tested. The use of a higher-order Frobenius approximation ($n=16$) results in closer agreement with the exact curve. For our purposes, the first-order Frobenius approximation ($n=1$) is sufficient.

For most systems $\lambda_{\text{app}}>\lambda_*$, which is unfortunate as it often leads to $\|\lambda_{\text{app}}\mathbf{S}-\mathbf{I}\|_2>1$. In the third column of Table II, we compare $\|\lambda\mathbf{S}-\mathbf{I}\|_2$ for $\lambda=\lambda_*$, λ_{app}, and λ_{GCT}. When the optimal scaling parameter is used, $\|\lambda_*\mathbf{S}-\mathbf{I}\|_2$ is about 0.99. This value is only slightly below 1.0 and occurs since the overlap matrices usually have their smallest eigenvalue ϵ_{min} close to zero. For λ_{app}, the $\|\lambda_{\text{app}}\mathbf{S}-\mathbf{I}\|_2$ values are generally a bit above 1, thereby violating the condition Eq. (25). In practice, this is not a problem since the convergence interval is broader than (0, 1), see Table I. The reduction of λ_{app} by a factor of 0.9 as described in Sec. II D was only needed for $m=3$ on the InAg complex (176) and the random molecule (64). Since λ_{GCT} systematically underestimates λ_*, $\|\lambda_{\text{GCT}}\mathbf{S}-\mathbf{I}\|_2<1$ always holds, although this comes at the small price of one more iteration in some cases.

Between 9 and 23 iterations, typically 15, second-order Newton-Schulz iterations are required for convergence to 10^{-10}, almost independently of whether λ_*, λ_{app}, or λ_{GCT} is used. The third-order Newton-Schulz iterations are slightly more sensitive to the λ approximation as its convergence interval is more narrow, but the total number of iterations is reduced to between 6 and 16, typically about 11.

As regards to the intermediate scaling counterparts, we observe a large gain for the second-order iterations, where the number of iterations is reduced by 20%–30%, depending on condition number. However, this reduction is slightly offset by having to recompute extremal eigenvalues of the unscaled $\mathbf{X}_k$ in each iteration. We may avoid this extra work by assuming that there is always an eigenvalue very close to 1 and that the lowest eigenvalue remains lowest in each (nonmonotonic) iteration. The iterative estimate may then be replaced by the update Eq. (44) using the mapping function [Eq. (36)].

The third-order Newton-Schulz iterations with intermediate scaling have the advantage of being intrinsically mono-

TABLE II. Calculations using the scaled *m*th-order Newton-Schulz iterative algorithm for the 12 sample molecules. The optimal λ_* (first line), the approximate λ_{app} (second line), and λ_{GCT} (third line) scaling parameters have been used. $N^{(m)}$ is the number of *m*th-order Newton-Schulz iterations and the subscript int denotes intermediate scaling. The numbers in parentheses are the number of atoms.

	$\|\mathbf{S}-\mathbf{I}\|_2$	λ	$\|\lambda\mathbf{S}-\mathbf{I}\|_2$	$N^{(2)}$	$N_{int}^{(2)}$	$N^{(3)}$	$N_{int}^{(3)}$
			Fullerene C_{60}				
λ_*	5.9925	0.2860	0.9999	17	12	11	10
λ_{app}		0.2928	1.0472	17		11	
λ_{GCT}		0.2102	0.9999	17		12	
			Polysaccharide (438)				
λ_*	5.1582	0.3247	0.9994	14	11	10	9
λ_{app}		0.3477	1.1411	14		10	
λ_{GCT}		0.1914	0.9996	15		10	
			Polysaccharide (875)				
λ_*	5.1582	0.3247	0.9994	14	11	10	9
λ_{app}		0.3473	1.1385	14		10	
λ_{GCT}		0.1872	0.9996	15		11	
			DNA fragment (583)				
λ_*	4.9322	0.3371	0.9997	15	11	11	9
λ_{app}		0.3468	1.0570	15		11	
λ_{GCT}		0.2004	0.9998	16		11	
			PRC model (1082)				
λ_*	5.7104	0.2980	0.9999	16	12	11	10
λ_{app}		0.3321	1.2287	16		11	
λ_{GCT}		0.1485	0.9999	17		12	
			Red phosphorus (506)				
λ_*	1.8941	0.6707	0.9410	8	7	6	6
λ_{app}		0.7722	1.2348	8		6	
λ_{GCT}		0.5174	0.9544	9		7	
			Red phosphorus (810)				
λ_*	4.7282	0.3490	0.9991	14	10	10	9
λ_{app}		0.3454	0.9991	14		10	
λ_{GCT}		0.2352	0.9994	14		10	
			Crystobalite (984)				
λ_*	4.4027	0.3697	0.9973	13	9	9	8
λ_{app}		0.3747	1.0244	13		9	
λ_{GCT}		0.2186	0.9984	13		9	
			Tridymite (1314)				
λ_*	5.1940	0.3229	0.9998	15	11	11	10
λ_{app}		0.3622	1.2436	15		11	
λ_{GCT}		0.1991	0.9999	16		11	
			Rh complex (25)				
λ_*	5.6723	0.2997	0.9998	16	11	11	10
λ_{app}		0.3078	1.0538	16		11	
λ_{GCT}		0.1858	0.9999	16		11	
			InAg complex (176)				
λ_*	9.9571	0.1825	0.9999	20	14	14	12
λ_{app}		0.2492	1.7308	20		20	
λ_{GCT}		0.1121	0.9999	21		15	
			Random molecule (64)				
λ_*	15.9537	0.1180	0.9999	23	15	16	14
λ_{app}		0.1737	1.9441	23		27	
λ_{GCT}		0.0674	0.9999	24		16	

tonically convergent. We therefore only need to compute the extremal eigenvalues in the first step and then use the mapping function to obtain λ_k. However, the gain of intermediate scaling is rather small for $m=3$, usually saving only one iteration. The gain becomes larger for ill-conditioned matrices.

The best way to compare the performance of different-order Newton-Schulz iterations is to compare the total number of matrix-matrix multiplications required. We need $m+1$ matrix-matrix multiplications per iteration, as discussed in Sec. II G. (In the first iteration, this number could be reduced to $m-1$ by removing multiplications with the unit matrix.) From the results in Table II, we then observe the following: For well-conditioned overlap matrices, the difference in the number of iterations between $m=2$ and $m=3$ is negligible. Moreover, intermediate scaling brings little improvement. This is best illustrated with the calculations on red phosphorus (506). The best approach is therefore to use $m=2$ without intermediate scaling for well conditioned overlap matrices. By contrast, as the overlap matrix becomes ill conditioned, the higher-order methods perform better and intermediate scaling becomes beneficial, as is best illustrated by the calculations on the InAg complex (176) and the random molecule (64). For these two molecules the number of matrix-matrix multiplications is substantially different for $m=2$ and $m=3$, with and without intermediate scaling. The second-order method with intermediate scaling appears to be the most efficient in these cases.

Based on these results and also on our experience and understanding of these methods, we set up a hierarchy of the different orders m based on the number of matrix-matrix multiplications $M^{(m)}$ required to converge, denoting intermediate scaling by the subscript int. For a well-conditioned overlap matrix, the hierarchy is $M^{(2)}=M^{(2)}_{\mathrm{int}}<\cdots<M^{(5)}=M^{(5)}_{\mathrm{int}}$; for an ill-conditioned matrix, it becomes $M^{(2)}_{\mathrm{int}}<\cdots<M^{(5)}_{\mathrm{int}}<M^{(5)}<\cdots<M^{(2)}$.

C. Examples of linear-scaling calculations

Linear scaling is demonstrated by carrying out calculations on a series of polysaccharides and polyalanine peptides. The polysaccharides were obtained from polysaccharide (438) by doubling its size using inversion symmetry and by reducing the number of saccharide components, giving systems containing 45, 86, 148, 230, 375, 438, and 850 atoms. For polyalanines, a series of 15 polyalanines was generated by connecting alanine units by a peptidic bond, giving chains containing between 103 and 1503 atoms. The $6-31G$ basis set was used for both the polysaccharides and polyalanines. The largest polysaccharide and polyalanine contain about 5000 and 8000 basis functions, respectively. The CPU times required to compute the $\mathbf{S}^{1/2}$ and $\mathbf{S}^{-1/2}$ using diagonalization [Eqs. (3)–(5)] and the scaled second-order Newton-Schulz iterations [Eq. (26)] were measured. For the Newton-Schulz iterations, timings are given both with full- and sparse-matrix algebra. Sparsity was exploited using the block sparse-matrix (BSM) implementation of Ref. 16. The results are collected in Fig. 3.

Diagonalization (full diamonds) is about 2.5 times faster than the Newton-Schulz scheme when full matrix algebra (full squares) is used. With BSM algebra (empty squares), linear scaling is obtained with system size both for the polysaccharides and polyalanines. For the polysaccharides, diagonalization is more efficient for the small systems with a crossover at about 750 atoms. For the polyalanines, the crossover occurs for much smaller systems. This difference is caused by a structural difference in the two systems. Whereas diagonalization timings depend only on the number of basis functions (proportional to the number of atoms) and thus are virtually the same for polysaccharides and polyalanines that contain the same number of atoms; the performance of the BSM iteration depend critically on the structure of the molecules, being much more efficient for regular one-dimensional structures such as the polyalanines than for irregular structures such as the polysaccharides, where the sparsity of the overlap matrix is relatively small and unevenly distributed.

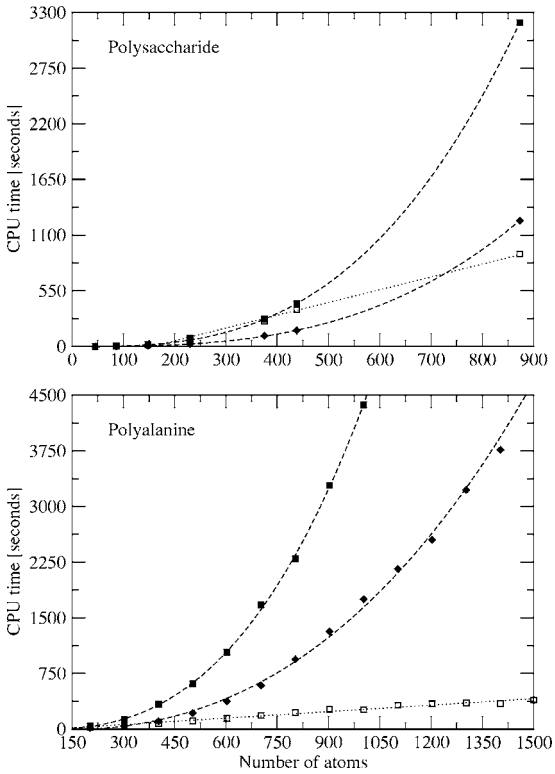

FIG. 3. CPU timings for evaluating the Löwdin decomposition for the polysaccharides and the polyalanines. Timings are given for the scaled second-order Newton-Schulz iteration method using full matrix algebra (full squares), BSM algebra (empty squares), and when diagonalization is used (full diamonds). Upper panel: Polysaccharides. Lower panel: Polyalanines.

The scaled Newton-Schulz iterations are of practical interest even when full matrix algebra is used because matrix-matrix operations can be efficiently parallelized, achieving a nearly linear speedup with the number of processors. This is particularly true on shared-memory systems, where the communication bottleneck is eliminated. Diagonalization can also be partially parallelized. The efficiency, however, is not nearly as good as for the matrix-matrix operations. Already for a few (4–6) processors, the Newton-Schulz iterations will thus result in a faster throughput than diagonalization.

IV. SUMMARY

We have shown that the symmetric square-root or Löwdin decomposition of the overlap matrix $\mathbf{S}$ can be carried out at linear cost. Using scaled Newton-Schulz iterations, the evaluation of $\mathbf{S}^{1/2}$ and $\mathbf{S}^{-1/2}$ is expressed in terms of iterations that are mth-order convergent. With a proper scaling of the overlap matrix, the Newton-Schulz procedure is guaranteed to converge globally.

Test calculations on 12 structurally different molecules demonstrate the efficiency and stability of the proposed Newton-Schulz algorithm, and that linear scaling is achieved with sparse-matrix algebra. The best approach is $m=2$ without intermediate scaling for well conditioned and $m=2$ with intermediate scaling for ill-conditioned overlap matrices.

Conventionally, $\mathbf{S}^{1/2}$ and $\mathbf{S}^{-1/2}$ are calculated by diagonalization of the overlap matrix. On a single processor, diagonalization is two to three times faster than the scaled Newton-Schulz method, when full-matrix algebra is used. However, the Newton-Schulz method is matrix-matrix multiplication driven and thus trivial to parallelize and becomes faster already when a few processors are used. Moreover, when sparse-matrix algebra is used, linear scaling makes the Newton-Schulz method faster than diagonalization for systems containing several hundred atoms depending on the structure of the considered system.

With the scaled Newton-Schulz method presented in this paper, the calculation of the matrix square root (and its inverse) of the overlap matrix can be carried out, even for very large systems. As a result, the transformation to the Löwdin basis can now be achieved reliably and at little cost for such systems, making the development of efficient linear-scaling methods for large molecular systems easier.

ACKNOWLEDGMENTS

The authors gratefully acknowledge the Lundbeck Foundation and Forskningsrådet for Natur og Univers for providing funds for our research. One of the authors (T.H.) acknowledges support from the Norwegian Research Council through a Strategic University Program in Quantum Chemistry (Grant No. 154011/420).

[1] J. M. Millam and G. E. Scuseria, J. Chem. Phys. **106**, 5569 (1997).
[2] Y. Shao, C. Saravanan, M. Head-Gordon, and C. A. White, J. Chem. Phys. **118**, 6144 (2003).
[3] B. C. Carlson and J. M. Keller, Phys. Rev. **105**, 102 (1957).
[4] P. Sałek, S. Reine, F. Pawlowski *et al.*, J. Chem. Phys. (to be published).
[5] S. Coriani, S. Høst, B. Jansík, L. Thøgersen, J. Olsen, and P. Jørgensen, J. Chem. Phys. (submitted).
[6] P. Manninen, V. Weijo, P. Jørgensen, O. Christiansen, and J. Olsen, J. Chem. Phys. (submitted).
[7] N. J. Higham, Numer. Algorithms **15**, 227 (1997).
[8] C. Kenny and A. J. Laub, SIAM J. Matrix Anal. Appl. **12**, 273 (1991).
[9] A. M. N. Niklasson, Phys. Rev. B **70**, 193102 (2004).
[10] N. J. Higham, Numer. Math. **62**, 539 (1992).
[11] K. Németh and G. E. Scuseria, J. Chem. Phys. **113**, 6035 (2000).
[12] S. Goedecker and G. E. Scuseria, Comput. Sci. Eng. **5**, 14 (2003).
[13] L. Thøgersen, J. Olsen, D. Yeager, P. Jørgensen, P. Sałek, and T. Helgaker, J. Chem. Phys. **121**, 16 (2004).
[14] R. Ahlrichs, A. Eichhöfer, D. Fenske, O. Hampe, M. M. Kappes, P. Nava, and J. Olkowska-Oetzel, Angew. Chem., Int. Ed. **43**, 3823 (2004).
[15] A. Schäfer, H. Horn, and R. Ahlrichs, J. Chem. Phys. **97**, 2571 (1992).
[16] E. H. Rubensson and P. Sałek, J. Comput. Chem. **26**, 1628 (2005).