The efficient optimization of molecular geometries using redundant internal coordinates

Vebjørn Bakken and Trygve Helgaker[a)]
Department of Chemistry, University of Oslo, P.O.B. 1033 Blindern, N-0315 Oslo, Norway

(Received 1 July 2002; accepted 27 August 2002)

The optimization of *ab initio* molecular geometries is discussed. Based on comparisons of 30 minimizations and 15 saddle-point optimizations, the most efficient combination of coordinate system, approximate and exact Hessians, and step control is determined. Use of a proposed set of extra-redundant internal coordinates is shown to reduce the number of geometry steps significantly relative to the use of redundant coordinates. Various update schemes are tested for minimum and saddle-point optimizations, including combination formulas. The complete expressions for the first and second derivatives of the Wilson B matrix are presented, thereby avoiding the need to calculate this by finite-difference methods. The presented scheme appears to be the most efficient, robust and generally applicable scheme to date. © *2002 American Institute of Physics.*
[DOI: 10.1063/1.1515483]

I. INTRODUCTION

An important step in many applications of *ab initio* quantum-chemistry methods to problems of chemical interest is the optimization of the molecular geometry so as to determine minima (corresponding to stable molecular structures) or saddle-points (corresponding to molecular transition states). Over the years, a large number of methods have been developed and refined for such optimizations. Although based on the standard techniques of optimization theory and numerical analysis, the most efficient methods are highly adapted to the special requirements of molecular geometry optimizations. This is particularly true for the choice of coordinate systems and for the choice of Hessians in quasi-Newton theory, both of which can critically affect the performance of the optimization.

In the present paper, the optimization of *ab initio* molecular potential-energy surfaces is reviewed and analyzed so as to establish the most efficient scheme for the localization of minima and saddle points. Based on a comparison of 30 minimizations and 15 saddle-point optimizations, the best combination of coordinate system, Hessian, and step control is determined. Extra-redundant internal coordinates are introduced and shown to reduce the number of geometry steps significantly relative to the use of standard redundant coordinates. In addition, a variety of Hessian updates are considered for minimum and saddle-point optimizations, including combination formulas. The proposed optimization scheme, which is slightly different for minimizations and saddle-point optimizations, appears to be the most efficient and robust scheme to date.

For any optimization based on the use of internal coordinates such as the redundant and extra-redundant coordinates, the Wilson B matrix plays an important role. In particular, for the transformations between Cartesian coordinates (in which the energy, gradient and Hessians are calculated) and the internal coordinates (in which the geometry step is determined), the derivatives of the Wilson B matrix are needed. In the present paper, analytical expressions for these derivatives are given to second order, avoiding their calculation by finite difference.

The redundant internal coordinates were introduced by Peng *et al.*[1] We here combine these and the extra-redundant coordinates with the model Hessian of Lindh *et al.*,[2] noting that Eckert *et al.* achieved good performance by using the model Hessian with natural internal coordinates.[3] Previous work has focused on the performance of minimizations, although Baker and Chan have presented benchmarks for saddle-point optimizations.[4] In this work, we benchmark transition-state optimizations as well as minimizations.

The scheme to be presented here scales as $O(N^3)$—that is, cubically in the number of atoms. For molecules of moderate size studied by conventional *ab initio* methods, the cost of setting up the internal coordinate system and performing the necessary transformations is negligible compared with the cost of calculating the energy, the gradient and possibly the Hessian. However, for very large systems, the cost of setting up the coordinate system and determining the geometry step may become a bottleneck in the calculation. Methods have been developed that take care of this problem by reducing the scaling.[5,6]

After a discussion of coordinate systems and the derivatives of the Wilson B matrix in Sec. II, we consider approximate Hessians and their updates in Sec. III. Next, in Sec. IV, we discuss step control and convergence criteria. Then, in Sec. V, we carry out a detailed comparison of the different schemes, determining the best combination of coordinate systems, approximate Hessians, and step control for minimizations and saddle-point optimizations. Conclusions are given in Sec. VI.

II. COORDINATE SYSTEMS

The choice of coordinate system is critical for the efficiency of a geometry optimization. Since gradients and Hes-

a)Electronic mail: trygve.helgaker@kjemi.uio.no

sians are usually calculated in Cartesian space, the most straightforward choice is *Cartesian coordinates*. However, many studies have shown that Cartesians are generally outperformed by well-chosen sets of *internal coordinates*—that is, bond lengths, bond angles, and dihedral angles. Internal coordinate systems can be classified as redundant or nonredundant. In a nonredundant internal coordinate system, the number of coordinates is equal to the number of internal degrees of freedom $3N-6(5)$, where N is the number of atoms in the system; in a redundant set, there are more coordinates than degrees of freedom.

The *Z-matrix coordinates* are a typical example of nonredundant internal coordinates.[7] Although less and less frequently used for optimizations,[1,8,9] they are still useful for specifying molecular geometries. A difficulty with using Z-matrix coordinates for optimizations, in particular for ring systems, is to decide which primitive internal coordinates to include and which to exclude. A poor choice of internals may seriously degrade the performance of an optimizer.

The *natural internal coordinates* solve this problem by forming local linear combinations of primitive internals.[10,11] The coordinates can be set up automatically by applying a set of rules, although the number of rules and special cases is quite extensive. However, the performance of these coordinates is excellent.[3,9] In this study, we focus on *redundant internal coordinates*[1] and (nonredundant) *delocalized internal coordinates*.[12] In the following, we consider first the redundant internal coordinates and their derivatives in Secs. II A and II B; next, we consider the delocalized internal coordinates in Sec. II C.

A. Redundant internal coordinates

The redundant internal coordinates are defined in Ref. 1. We shall here briefly review the definitions and consider some points concerning this choice of coordinate system.

1. Setting up the redundant internal coordinates for a molecule

The key to setting up the redundant internal coordinates is to determine what atoms are bonded to each other and the nature of each bond. The assignment of bonds depends on available (tabulated) values for covalent and van der Waals radii as functions of atomic number.

Regular bonds are assigned to all pairs of atoms where the interatomic distance is less than or equal to 1.3 times the sum of their respective covalent radii. Once the regular bonds are in place, we identify isolated, disconnected fragments. If two (or more) such fragments are found, the shortest bond between two atoms belonging to separate fragments defines an interfragment bond coordinate. Other interfragment distances that are either less than 2 Å or less than 1.3 times the interfragment bond coordinate constitute the auxiliary interfragment bonds. Finally, we check for hydrogen bonds by examining all bonds where a hydrogen atom is bonded to a small, electronegative atom X where X=N, O, F, P, S, Cl. If the distance from this hydrogen atom to another small, electronegative atom Y is greater than the sum of the covalent radii of H and Y but less than 0.9 times the sum of their van der Waals radii, and if the angle X–H···Y is greater than 90°, then a hydrogen bond is assigned between H and Y.

We also propose that an *extra-redundant coordinate system* can be created by assigning auxiliary bonds to all pairs of atoms where the distance between the atoms is less than 2.5 times the sum of their covalent radii. This will generate bond coordinates between atoms having a common neighbor—for example, across a bond angle. We shall return to this alternative coordinate system later.

Once the bonds are in place, the bending coordinates are easily generated. Bond angles are assigned to all sets of three atoms A, B, and C where A is bonded to B and B to C. Only regular bonds, interfragment bonds, and hydrogen bonds are allowed to generate bond angles—auxiliary bonds and auxiliary interfragment bonds do not generate angles. Bending coordinates where the angle is nearly linear ($\angle_{ABC}>175°$) require special attention. In these cases, a second orthogonal bending coordinate is assigned, ensuring that the linear structure is stabilized.

Dihedral angles (twisting coordinates) are assigned in the same way as angles. All sets of four atoms A, B, C, and D, where A is bonded to B, B is bonded to C, and C is bonded to D are included, provided $\angle_{ABC}\neq 180°$ and $\angle_{BCD}\neq 180°$. Again, only regular bonds, interfragment bonds and hydrogen bonds generate twisting coordinates—auxiliary bonds and auxiliary interfragment bonds do not.

If a molecule contains four or more atoms and no dihedral angles are found by using the above scheme, an additional search is carried out to ensure that out-of-plane bending is added. Arbitrary combinations of four atoms are tried until a well-defined dihedral angle is found (none will be found if the molecule is linear). Adding just one dihedral angle could break the molecular symmetry. Therefore, all well-defined permutations of the chosen dihedral angle are added as well ($\angle_{ABDC}$, $\angle_{BCDA}$, etc.), generating at most 12 unique twisting coordinates.

2. Wilson's B matrix

Wilson's B matrix gives the relationship between the redundant internal coordinates and the Cartesian coordinates.[13] Its elements are given by

$$B_{ij}=\frac{\partial q_i}{\partial x_j}, \tag{1}$$

where the q_i are the internal coordinates and the x_j are the Cartesian displacement coordinates of the atoms. The B matrix is thus a rectangular matrix; the number of rows is equal to the number of internal coordinates, and the number of columns equal to the number of Cartesian coordinates. Explicit expressions for the elements of the B matrix Eq. (1) are given in Sec. II B.

Small changes in the Cartesian coordinates transform into small changes in the internal coordinates as

$$\mathbf{B}\delta\mathbf{x}=\delta\mathbf{q}. \tag{2}$$

The inverse transformation is given by

$$\mathbf{B}^{+}\delta\mathbf{q}=\delta\mathbf{x}, \tag{3}$$

where the pseudo-inverse (or the generalized inverse) of a rectangular matrix has been introduced. The generalized inverse exists for any rectangular matrix, whether or not it is rank deficient. In general, it is obtained from a singular-value decomposition of **B**; for a square nonsingular matrix, it reduces to the standard inverse.

The properties of the generalized inverse are the following: If the linear equations Eq. (2) are overdetermined, then the pseudo-inverse $\mathbf{B}^+$ in Eq. (3) returns the solution $\delta\mathbf{x}$ that minimizes the norm of the residual $\|\mathbf{B}\delta\mathbf{x}-\delta\mathbf{q}\|$; if, on the other hand, the system is underdetermined, then the pseudo-inverse minimizes the norm of the solution $\|\delta\mathbf{x}\|$. Furthermore, from the theory of generalized inverses, it follows that the matrix

$$\mathbf{P}=\mathbf{B}\mathbf{B}^+ \tag{4}$$

constitutes a projection matrix onto the range of **B**. Applied to any physical or unphysical set of internal displacement coordinates $\delta\mathbf{q}$, it produces a unique vector

$$\delta\tilde{\mathbf{q}}=\mathbf{P}\delta\mathbf{q} \tag{5}$$

that is physically valid in the sense that it is consistent with a variation in the Cartesian coordinates of the molecule.

3. Transformations between Cartesian and redundant internal coordinates

From the Wilson B matrix and its pseudo-inverse, we may set up the relations between the gradients and the Hessians in the internal and Cartesian coordinate systems in the following manner:

$$\mathbf{g}_x=\mathbf{B}^t\mathbf{g}_q\,, \tag{6}$$

$$\mathbf{H}_x=\mathbf{B}^t\mathbf{H}_q\mathbf{B}+\mathbf{K}, \tag{7}$$

where **K** is given by

$$K_{jk}=\sum_i\,[\mathbf{g}_q]_i B'_{ijk}=\sum_i\,[\mathbf{g}_q]_i\frac{\partial^2 q_i}{\partial x_j\partial x_k}. \tag{8}$$

By applying the generalized inverse to Eqs. (6) and (7), the opposite transformations are obtained:

$$\mathbf{g}_q=(\mathbf{B}^t)^+\mathbf{g}_x\,, \tag{9}$$

$$\mathbf{H}_q=(\mathbf{B}^t)^+(\mathbf{H}_x-\mathbf{K})\mathbf{B}^+. \tag{10}$$

Clearly, to transform the Hessian, the elements of **B** must be differentiated with respect to the Cartesian coordinates. The derivatives are given in Sec. II B.

4. Projection of the gradient and the Hessian

In a redundant coordinate system, care must be exercised to ensure that the gradient is physically valid—that is, that it corresponds to a valid change in the internal coordinates. Mathematically, a valid gradient must belong to the range of **B**. From a nonvalid gradient, a valid one can then be generated by projection with Eq. (4) and similarly for the Hessian

$$\tilde{\mathbf{g}}_q=\mathbf{P}\mathbf{g}_q\,, \tag{11}$$

$$\tilde{\mathbf{H}}_q=\mathbf{P}\mathbf{H}_q\mathbf{P}. \tag{12}$$

To ensure valid gradients and Hessians, these projections may be carried out at any point during the optimization. In particular, it is essential to perform this projection on inexact, initial Hessians and on updated Hessians, as well as on interpolated gradients (generated from line searches). On explicitly calculated gradients and Hessians, the projection may also be performed to remove any numerical noise introduced in the course of coordinate transformations.

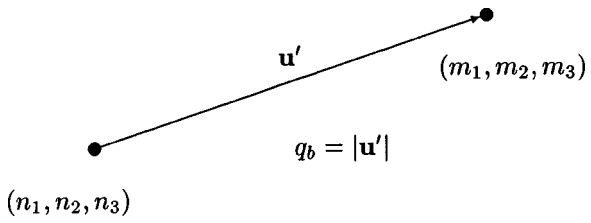

FIG. 1. The bond between two atoms.

5. Transformation of geometry steps

Once a step in internal coordinates has been determined, it must be converted to Cartesian coordinates—the reference frame in which energies, gradients, and Hessians are calculated. However, since the Cartesian coordinates are rectilinear and the internal coordinates curvilinear, there is no simple transformation of finite displacements. Instead, the transformation of the geometry step must be done iteratively.

Consider a system with initial Cartesian coordinates $\mathbf{x}_0$ and initial internal coordinates $\mathbf{q}_0$. For a step $\tilde{\mathbf{s}}_q$ in internal coordinates, the first estimate of the new Cartesian coordinates is given by

$$\mathbf{x}_1=\mathbf{x}_0+\mathbf{B}^+\tilde{\mathbf{s}}_q\,. \tag{13}$$

From the new Cartesian coordinates $\mathbf{x}_1$, a set of updated internal coordinates $\mathbf{q}_1$ is determined. Next, the difference between the requested and actual changes in the internal coordinates is calculated as

$$\Delta\mathbf{q}_k=\tilde{\mathbf{s}}_q-(\mathbf{q}_k-\mathbf{q}_0), \tag{14}$$

where $k=1$ in the first iteration, taking care to remove any multiples of 360° from the angles. The Cartesian displacements are then refined by transforming this difference in the same manner as the initial step:

$$\mathbf{x}_{k+1}=\mathbf{x}_k+\mathbf{B}^+\Delta\mathbf{q}_k\,. \tag{15}$$

Equations (14) and (15) define our iterative procedure. Convergence is declared when the root-mean-square change in the Cartesian coordinates $\mathbf{B}^+\Delta\mathbf{q}_k$ becomes less than 10^{-6}, if this root-mean-square change differs by less than 10^{-12} from that of the previous iteration, or if the number of iterations exceeds 25. During the refinement, $\Delta\mathbf{q}_k$ is compared with $\Delta\mathbf{q}_1$. In the rare cases when $\Delta\mathbf{q}_k>\Delta\mathbf{q}_1$, we revert to the initial estimate $\mathbf{x}_1$ in Eq. (13); otherwise, $\mathbf{x}_k$ is used for the next iteration of the optimization.

B. Derivatives of the redundant internal coordinates

1. Bond lengths

We begin by considering a redundant internal coordinate q_b that corresponds to a bond stretch, sometimes referred to as a stretching coordinate. Figure 1 shows a bond between

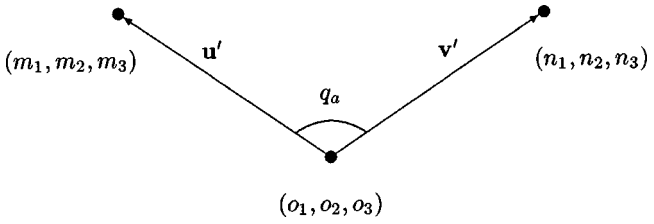

FIG. 2. The angle between two bonds.

two atoms **m** and **n** with Cartesian coordinates (m_1,m_2,m_3) and (n_1,n_2,n_3). The bond vector is given by

$$\mathbf{u}'=(u_1',u_2',u_3')=\mathbf{m}-\mathbf{n}=(m_1-n_1,m_2-n_2,m_3-n_3), \tag{16}$$

and the associated bond length is denoted by $\lambda_u=|\mathbf{u}'|$. Differentiating Eq. (16) with respect to the Cartesian coordinates, we obtain

$$\frac{\partial q_b}{\partial a_i}=\zeta_{amn}u_i; \quad a=m,n; \quad i=1,2,3, \tag{17}$$

where we have introduced the sign factor

$$\zeta_{amn}=(\delta_{am}-\delta_{an})=\begin{cases}1 & \text{if } a=m\\ -1 & \text{if } a=n\end{cases} \tag{18}$$

and the normalized bond vector with components (u_1,u_2,u_3) as

$$\mathbf{u}=\frac{\mathbf{u}'}{|\mathbf{u}'|}. \tag{19}$$

Apart from the sign factor, the derivative of the stretching coordinate Eq. (17) is thus given entirely by the normalized bond vector **u**. Further differentiation yields the second derivative of the bond stretching coordinate

$$\frac{\partial^2 q_b}{\partial a_i \partial b_j}=(-1)^{\delta_{ab}}\frac{u_iu_j-\delta_{ij}}{\lambda_u}; \quad a,b=m,n; \quad i,j=1,2,3, \tag{20}$$

where λ_u is the bond length Eq. (16).

2. Bond angles

Next, we consider the bond angle or bending coordinate q_a between the vectors from **o** to **m** and from **o** to **n** in Fig. 2. The bond vectors $\mathbf{u}'$ and $\mathbf{v}'$ are given by

$$\mathbf{u}'=\mathbf{m}-\mathbf{o}, \tag{21}$$

$$\mathbf{v}'=\mathbf{n}-\mathbf{o}, \tag{22}$$

and the associated normalized bond vectors and bond lengths are denoted by **u** and **v** and by λ_u and λ_v, respectively. In terms of these entities, the bond angle $0\leqslant q_a\leqslant\pi$ is given by

$$q_a=\arccos\left(\frac{\mathbf{u}'\cdot\mathbf{v}'}{\lambda_u\lambda_v}\right)=\arccos(\mathbf{u}\cdot\mathbf{v}). \tag{23}$$

The derivatives of a bond angle are more complicated than those of a bond length, partly because of the inequivalence of the central atom **o** with terminal atoms **m** and **n**. To express these derivatives in a compact manner, we introduce the vector $\mathbf{w}'$ perpendicular to **u** and **v** as

$$\mathbf{w}'=\begin{cases}\mathbf{u}\times\mathbf{v} & \text{if } \mathbf{u}\nparallel\mathbf{v}\\ \mathbf{u}\times[1,-1,1] & \text{if } \mathbf{u}\parallel\mathbf{v} \text{ and } \mathbf{u},\mathbf{v}\nparallel[1,-1,1]\\ \mathbf{u}\times[-1,1,1] & \text{if } \mathbf{u}\parallel\mathbf{v} \text{ and } \mathbf{u},\mathbf{v}\parallel[1,-1,1],\end{cases} \tag{24}$$

where the linear case ($q_a=180°$) is taken care of explicitly. The vectors $[1,-1,1]$ and $[-1,1,1]$ have been arbitrarily chosen to ensure that a perpendicular vector is generated. The associated normalized coordinate is denoted **w**.

The first derivatives of a bond angle with respect to Cartesian coordinates can now be expressed in the following manner, valid for all angles:

$$\frac{\partial q_a}{\partial a_i}=\zeta_{amo}\frac{[\mathbf{u}\times\mathbf{w}]_i}{\lambda_u}+\zeta_{ano}\frac{[\mathbf{w}\times\mathbf{v}]_i}{\lambda_v};$$

$$a=m,n,o; \quad i=1,2,3. \tag{25}$$

Next, using the relations

$$\cos(q_a)=\mathbf{u}\cdot\mathbf{v}; \quad \sin(q_a)=\sqrt{1-(\mathbf{u}\cdot\mathbf{v})^2}, \tag{26}$$

we arrive at the following expression for the second derivatives of a bond angle:

$$\begin{aligned}\frac{\partial^2 q_a}{\partial a_i\partial b_j}=&\zeta_{amo}\zeta_{bmo}\frac{u_iv_j+u_jv_i-3u_iu_j\cos(q_a)+\delta_{ij}\cos(q_a)}{\lambda_u^2\sin(q_a)}\\ &+\zeta_{ano}\zeta_{bno}\frac{v_iu_j+v_ju_i-3v_iv_j\cos(q_a)+\delta_{ij}\cos(q_a)}{\lambda_v^2\sin(q_a)}\\ &+\zeta_{amo}\zeta_{bno}\frac{u_iu_j+v_jv_i-u_iv_j\cos(q_a)-\delta_{ij}}{\lambda_u\lambda_v\sin(q_a)}\\ &+\zeta_{ano}\zeta_{bmo}\frac{v_iv_j+u_ju_i-v_iu_j\cos(q_a)-\delta_{ij}}{\lambda_u\lambda_v\sin(q_a)}\\ &-\frac{\cos(q_a)}{\sin(q_a)}\frac{\partial q_a}{\partial a_i}\frac{\partial q_a}{\partial b_j}; \quad a,b=m,n,o; \quad i,j=1,2,3,\end{aligned} \tag{27}$$

where the last term contains the first derivatives. Note that, in the linear case $q_a=180°$, the second derivatives are undefined because of the factor $\sin(q_a)$ in the denominators; the components of the derivative are then simply set to zero.

3. Dihedral angles

In Fig. 3, the dihedral angle q_d is introduced as the angle between the plane defined by the three points **m**, **o**, and **p**, and the plane defined by **n**, **p**, and **o**. Conventionally, q_d is restricted to the interval $-\pi<q\leqslant\pi$. Looking along the vector from **o** to **p**, the dihedral angle is said to be positive if the vector between **o** and **m** must be turned clockwise (by an angle less than or equal to π) to coincide with the vector from **p** to **n**.

To calculate the dihedral angle and its derivatives, we introduce the three bond vectors

$$\mathbf{u}'=\mathbf{m}-\mathbf{o}, \tag{28}$$

$$\mathbf{v}'=\mathbf{n}-\mathbf{p}, \tag{29}$$

$$\mathbf{w}'=\mathbf{p}-\mathbf{o}. \tag{30}$$

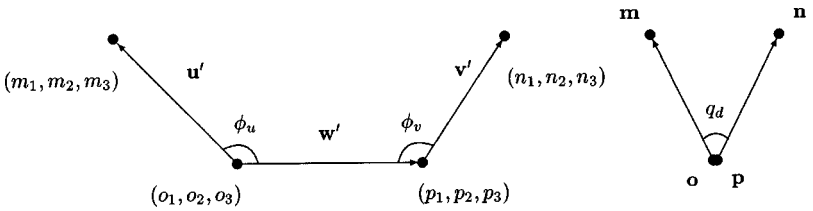

FIG. 3. The dihedral angle between four atoms, viewed from the side and along the central bond between **o** and **p**.

The corresponding bond lengths are λ_u, λ_v, and λ_w, while the normalized bond vectors are denoted **u**, **v**, and **w**. The expression for the dihedral (twisting) coordinate now becomes

$$\cos(q_d) = \frac{(\mathbf{u}\times\mathbf{w})\cdot(\mathbf{v}\times\mathbf{w})}{\sin(\phi_u)\sin(\phi_v)}. \tag{31}$$

The bond angles ϕ_u and ϕ_v are given in Fig. 3 and satisfy the relations

$$\cos(\phi_u) = \mathbf{u}\cdot\mathbf{w}; \quad \sin(\phi_u) = \sqrt{1-(\mathbf{u}\cdot\mathbf{w})^2}, \tag{32}$$

$$\cos(\phi_v) = -\mathbf{v}\cdot\mathbf{w}; \quad \sin(\phi_v) = \sqrt{1-(\mathbf{v}\cdot\mathbf{w})^2}. \tag{33}$$

The dihedral angle q_d is undefined if at least one of the angles ϕ_u and ϕ_v is equal to 0° or 180°, implying that the atoms **m**, **o**, and **p** or the atoms **o**, **p**, and **n** lie on the same line.

Upon differentiation, we find that the first and second derivatives of the dihedral angle q_d are given by the relations

$$\frac{\partial q_d}{\partial a_i} = \zeta_{amo}\frac{[\mathbf{u}\times\mathbf{w}]_i}{\lambda_u \sin^2(\phi_u)} + \zeta_{apn}\frac{[\mathbf{v}\times\mathbf{w}]_i}{\lambda_v \sin^2(\phi_v)} + \zeta_{aop}\left(\frac{[\mathbf{u}\times\mathbf{w}]_i \cos(\phi_u)}{\lambda_w \sin^2(\phi_u)} - \frac{[\mathbf{v}\times\mathbf{w}]_i \cos(\phi_v)}{\lambda_w \sin^2(\phi_v)}\right); \quad a=m,n,o,p; \quad i=1,2,3, \tag{34}$$

$$\begin{aligned}
\frac{\partial^2 q_d}{\partial a_i \partial b_j} &= \zeta_{amo}\zeta_{bmo}\frac{(1+P_{ij})[\mathbf{u}\times\mathbf{w}]_i(w_j\cos(\phi_u)-u_j)}{\lambda_u^2 \sin^4(\phi_u)} + \zeta_{anp}\zeta_{bnp}\frac{(1+P_{ij})[\mathbf{v}\times\mathbf{w}]_i(w_j\cos(\phi_v)-v_j)}{\lambda_v^2 \sin^4(\phi_v)} \\
&+ (\zeta_{amo}\zeta_{bop}+\zeta_{apo}\zeta_{bom})\frac{(1+P_{ij})[\mathbf{u}\times\mathbf{w}]_i(w_j-2u_j\cos(\phi_u)+w_j\cos^2(\phi_u))}{2\lambda_u\lambda_w \sin^4(\phi_u)} \\
&+ (\zeta_{anp}\zeta_{bpo}+\zeta_{apo}\zeta_{bnp})\frac{(1+P_{ij})[\mathbf{v}\times\mathbf{w}]_i(w_j+2u_j\cos(\phi_v)+w_j\cos^2(\phi_v))}{2\lambda_v\lambda_w \sin^4(\phi_v)} \\
&+ \zeta_{aop}\zeta_{bpo}\frac{(1+P_{ij})[\mathbf{u}\times\mathbf{w}]_i(u_j+u_j\cos^2(\phi_u)-3w_j\cos(\phi_u)+w_j\cos^3(\phi_u))}{2\lambda_w^2 \sin^4(\phi_u)} \\
&+ \zeta_{aop}\zeta_{bop}\frac{(1+P_{ij})[\mathbf{v}\times\mathbf{w}]_i(v_j+v_j\cos^2(\phi_v)+3w_j\cos(\phi_v)-w_j\cos^3(\phi_v))}{2\lambda_w^2 \sin^4(\phi_v)} \\
&+ (1-\delta_{ab})(\zeta_{amo}\zeta_{bop}+\zeta_{apo}\zeta_{bom})(j-i)(-1/2)^{|j-i|}\frac{w_k\cos(\phi_u)-u_k}{\lambda_u\lambda_w \sin(\phi_u)} \\
&+ (1-\delta_{ab})(\zeta_{ano}\zeta_{bop}+\zeta_{apo}\zeta_{bom})(j-i)(-1/2)^{|j-i|}\frac{w_k\cos(\phi_v)-v_k}{\lambda_v\lambda_w \sin(\phi_v)} \quad a,b=m,n,o,p; \quad i,j=1,2,3; \quad k\neq i,j,
\end{aligned} \tag{35}$$

where the permutation operator P_{ij} exchanges the indices i and j. The expression for the second derivatives may look awkwardly big, but further inspection reveals that most of the terms are built up from the same components. Since the last two terms in Eq. (35) contribute only when i and j refer to different Cartesian directions, k is uniquely defined as the third Cartesian component. As there is no coupling between the terminal atoms **m** and **n**, we conclude that

$$\frac{\partial^2 q_d}{\partial m_i \partial n_j} = 0; \quad i,j=1,2,3 \tag{36}$$

which also follows as a special case from Eq. (35).

C. Delocalized internal coordinates

A nonredundant internal coordinate system can quite easily be constructed from the eigenvectors of the B matrix with nonzero eigenvalues. These delocalized internal coordinates[12] are linear combinations of the primitive internals, In general these combinations are nonlocal, hence the name "delocalized."

The eigenvectors give the relation between delocalized and redundant internal coordinates. From the decomposition

of the redundant B matrix, we can construct a nonredundant B matrix that behaves in much the same way as the redundant B matrix, determining the delocalized coordinates in terms of Cartesian coordinates. The transformation equations derived here also hold for the nonredundant B matrix, which can be regarded as a special case of the redundant matrix.

III. HESSIANS, INITIAL CHOICE AND UPDATING

In optimization theory, the information provided by the Hessian about the local curvature of the potential-energy surface is utilized in different ways. The Newton method, in particular, makes full use of the Hessian, which is calculated exactly at each iteration, leading to a rapid convergence. However, although analytical Hessians are available for a wide range of electronic-structure models, their calculation is still quite expensive. By contrast, in methods such as the steepest-descent and conjugate-gradient methods, the information provided by the Hessian is discarded altogether, leading to a rather slow convergence.

Usually, however, the most efficient approach to optimization is to calculate the Hessian approximately, based on any information available about the potential-energy surface. Quasi-Newton methods, in particular, proceed in the same manner as Newton's method but use an approximate, updated Hessian rather than the exact Hessian at each iteration. If such methods are to succeed, the reduced computational cost at each iteration must more than offset the increase in the number of iterations for an optimization due to less accurate information about the potential-energy surface.

A. Initial Hessian

In quasi-Newton methods, the Hessian is updated based on information gained about the curvature of the potential-energy surface in the course of the optimization. The question then arises how to choose the Hessian in the first iteration, when no previous information about the curvature is available. One approach is to calculate the Hessian exactly at the first geometry, ensuring that the first step is a good one and that the updated Hessian generated in the subsequent iterations are good approximations to the exact Hessian. An exact initial Hessian is particularly useful in Cartesian coordinates, in which the Hessian is highly nondiagonal and difficult to model.

In internal coordinates, by contrast, the Hessian elements are closely related to the molecular structure, making it easier to estimate the Hessian reliably. Moreover, the couplings among the internal coordinates are usually small, making the quality of the model Hessian less critically dependent on the choice of nondiagonal elements. In fact, in internal coordinates, even a diagonal Hessian often works well, in particular when the diagonal elements are scaled based on the character of the coordinate. Typical values are (in atomic units) 0.5 for bond lengths, 0.2 for bond angles, and 0.1 for dihedral angles.

1. Model Hessian

Taking this idea of an approximate Hessian one step further, Lindh *et al.* proposed the following model Hessian for geometry optimizations.[2] Assuming a simple harmonic potential about the equilibrium geometry for each primitive internal coordinate, we write down the contributions to the classical energy from stretches, bends, and dihedral angles

TABLE I. Parameters of the simple harmonic model Hessian of Lindh *et al.*[2]. For a pair of atoms i and j, the value of the parameters α_{ij} and $r_{\mathrm{ref},ij}$ are determined by the periods to which the atoms belong. In addition three fixed parameters $k_r=0.45$, $k_\phi=0.15$ and $k_r=0.005$ are used. All parameters are given in atomic units.

		1st period	2nd period	3rd period
α_{ij}	1st period	1,000	0.3949	0.3949
	2nd period	0.3949	0.2800	0.2800
	3rd period	0.3949	0.2800	0.2800
$r_{\mathrm{ref},ij}$	1st period	1.35	2.10	2.53
	2nd period	2.10	2.87	3.40
	3rd period	2.53	3.40	3.40

$$E_s(\mathbf{q})=\tfrac{1}{2}{\sum_{ij}}' \; k_{ij}(r_{ij}-r_{0,ij})^2, \tag{37}$$

$$E_b(\mathbf{q})=\tfrac{1}{2}{\sum_{ijk}}' \; k_{ijk}(\theta_{ijk}-\theta_{0,ijk})^2, \tag{38}$$

$$E_v(\mathbf{q})=\tfrac{1}{2}{\sum_{ijkl}}' \; k_{ijkl}(\tau_{ijkl}-\tau_{0,ijkl})^2, \tag{39}$$

where the summations are over all atoms. Note that the internal coordinates are treated as independent, their energy contributions depending only on the force constants k_{ij}, k_{ijk}, and k_{ijkl}; on the values of the internal coordinates at the current geometry, r_{ij}, θ_{ijk}, and τ_{ijkl}; and on the reference values of the coordinates at the equilibrium geometry $r_{0,ij}$, $\theta_{0,ijk}$, and $\tau_{0,ijkl}$. The force constants are approximated as

$$k_{ij}=k_r\rho_{ij}; \quad k_{ijk}=k_\phi\rho_{ij}\rho_{jk}; \quad k_{ijkl}=k_\tau\rho_{ij}\rho_{jk}\rho_{kl}, \tag{40}$$

where

$$\rho_{ij}=\exp[\alpha_{ij}(r_{\mathrm{ref},ij}^2-r_{ij}^2)]. \tag{41}$$

This function falls off rapidly with the distance between the atoms, ensuring that the contribution from two widely separated atoms is small. As a final simplification, α_{ij} and $r_{\mathrm{ref},ij}^2$ are chosen to depend only on the period to which i and j belong. Parameters for all combinations of first, second, and third row elements are found in Table I. A total of 15 independent parameters are needed to define this model Hessian.

B. Updating the Hessian

There are several different formulas for updating the Hessian. All work by making adjustments to the previous Hessian $\mathbf{G}_{i-1}$ based on the coordinate difference (step) $\Delta\mathbf{x}_i=\mathbf{x}_i-\mathbf{x}_{i-1}$ and gradient difference $\Delta\mathbf{g}_i=\mathbf{g}_i-\mathbf{g}_{i-1}$ between the current and the previous geometries.[14–16] The most straightforward updating formula is the Murtagh–Sargent or symmetric rank-one (SR1) formula

$$\mathbf{G}_i^{\mathrm{MS}}=\mathbf{G}_{i-1}+\frac{(\Delta\mathbf{g}_i-\mathbf{G}_{i-1}\Delta\mathbf{x}_i)(\Delta\mathbf{g}_i-\mathbf{G}_{i-1}\Delta\mathbf{x}_i)^{\mathrm{t}}}{(\Delta\mathbf{g}_i-\mathbf{G}_{i-1}\Delta\mathbf{x}_i)^{\mathrm{t}}\Delta\mathbf{x}_i}. \tag{42}$$

A problem with this formula is that the denominator may become small and cause numerical instability. However, since it does not enforce (or preserve) a positive definite Hessian, it can be used for both minimizations and saddle point optimizations.

An example of a rank-two update is the Powell-symmetric-Broyden (PSB) formula

$$\mathbf{G}_i^{\mathrm{PSB}} = \mathbf{G}_{i-1} + \frac{(\Delta\mathbf{g}_i - \mathbf{G}_{i-1}\Delta\mathbf{x}_i)\Delta\mathbf{x}_i^{\mathrm{t}} + \Delta\mathbf{x}_i(\Delta\mathbf{g}_i - \mathbf{G}_{i-1}\Delta\mathbf{x}_i)^{\mathrm{t}}}{\Delta\mathbf{x}_i^{\mathrm{t}}\Delta\mathbf{x}_i} - \frac{(\Delta\mathbf{x}_i^{\mathrm{t}}\Delta\mathbf{g}_i - \Delta\mathbf{x}_i^{\mathrm{t}}\mathbf{G}_{i-1}\Delta\mathbf{x}_i)\Delta\mathbf{x}_i\Delta\mathbf{x}_i^{\mathrm{t}}}{(\Delta\mathbf{x}_i^{\mathrm{t}}\Delta\mathbf{x}_i)^2}. \tag{43}$$

This formula is suitable for finding both minima and saddle points. By contrast, the rank-two Broyden–Fletcher–Shanno–Goldfarb (BFSG) formula

$$\mathbf{G}_i^{\mathrm{BFGS}} = \mathbf{G}_{i-1} + \frac{\Delta\mathbf{g}_i\Delta\mathbf{g}_i^{\mathrm{t}}}{\Delta\mathbf{g}_i^{\mathrm{t}}\Delta\mathbf{x}_i} - \frac{\mathbf{G}_{i-1}\Delta\mathbf{x}_i\Delta\mathbf{x}_i^{\mathrm{t}}\mathbf{G}_{i-1}}{\Delta\mathbf{x}_i^{\mathrm{t}}\mathbf{G}_{i-1}\Delta\mathbf{x}_i}, \tag{44}$$

guarantees (under certain weak conditions) a positive definite matrix provided $\mathbf{G}_{i-1}$ is positive definitive, making it useful for minimizations but unsuitable for saddle-point optimizations.

Bofill has proposed to combine two updating formulas, so as to avoid the numerical instability of the MS formula by switching to the PSB update when the MS denominator becomes small[17]

$$\mathbf{G}_i = (1-\phi_i)\mathbf{G}_i^{\mathrm{MS}} + \phi_i\mathbf{G}_i^{\mathrm{PSB}}, \tag{45}$$

$$\phi_i = 1 - \frac{(\Delta\mathbf{x}_i^{\mathrm{t}}\mathbf{E}_i)^2}{(\Delta\mathbf{x}_i^{\mathrm{t}}\Delta\mathbf{x}_i)(\mathbf{E}_i^{\mathrm{t}}\mathbf{E}_i)}; \quad \mathbf{E}_i = \Delta\mathbf{g}_i - \mathbf{G}_{i-1}\Delta\mathbf{x}_i. \tag{46}$$

Bofill's update is intended for saddle-point optimizations. We here investigate a similar combination formula, replacing PSB with BFGS in the equations above:

$$\mathbf{G}_i = (1-\phi_i)\mathbf{G}_i^{\mathrm{MS}} + \phi_i\mathbf{G}_i^{\mathrm{BFGS}}, \tag{47}$$

where ϕ_i is defined in Eq. (46).

IV. STEP CONTROL AND CONVERGENCE CRITERIA

Various methods for step control exists, for minimizations and saddle-point optimizations. We here review the most popular ones. Finally, we discuss convergence criteria.

A. Newton method

Assuming we have analytical gradients and Hessians available, our Taylor expansion of the potential-energy surface is correct to second order. The Newton step then take us to the minimum of this model

$$\mathbf{s}_i = -\mathbf{H}_i^{-1}\mathbf{g}_i. \tag{48}$$

The inverse Hessian is required to determine the Newton step, making it convenient to transform the gradient and the Hessian to the Hessian eigenvector basis

$$\tilde{\mathbf{g}}_i = \mathbf{V}^{\mathrm{t}}\mathbf{g}_i, \tag{49}$$

$$\tilde{\mathbf{H}}_i = \mathbf{V}^{\mathrm{t}}\mathbf{H}_i\mathbf{V}. \tag{50}$$

Here $\mathbf{V}$ is the matrix of eigenvectors and $\tilde{\mathbf{H}}_i$ the Hessian in the diagonal representation. The components of the Newton step can then be written as

$$\tilde{s}_k = -\frac{\tilde{g}_k}{\tilde{H}_{kk}}. \tag{51}$$

For zero eigenvalues, the corresponding step component is set to zero. Finally, the diagonal step is transformed back to the original representation

$$\mathbf{s}_i = \mathbf{V}\tilde{\mathbf{s}}_i. \tag{52}$$

In the quasi-Newton method, the exact Hessian $\mathbf{H}$ is replaced by the approximate Hessian $\mathbf{G}$ in these equations.

B. Step restrictions and trust region

From Eq. (51), it is clear that small eigenvalues can give large Newton steps, sometimes leading to divergence. To guarantee global convergence, we must restrict the step length. Clearly, since our approximation to the true potential is based on a second-order Taylor expansion, it is valid only in a restricted region about the point of expansion: the *trust region*, assumed to have the shape of a hypersphere with a *trust radius* τ.[14,18] In the *Newton trust-region method*, we restrict our steps to be smaller than or equal to the trust radius.

C. The level-shifted trust-region Newton method

The trust radius τ must be given a reasonable initial value, but it is updated during the optimization based on how well the quadratic model describes the potential-energy surface. At each iteration, the energy predicted by the quadratic model at the previous iteration is compared with the exact energy. If the ratio is close to one, τ is increased; if it is poor, the τ is reduced.

If the step determined by Eq. (51) is smaller than τ, the Newton step is used. Otherwise, we use Lagrange's method of undetermined multipliers to minimize our model function under the restriction that the step length is equal to τ—that is, we locate the minimum on the hypersphere of radius τ. In practice, this is done by introducing a level-shift parameter μ that satisfies the condition

$$\sqrt{\mathbf{g}_i^{\mathrm{t}}(\mathbf{H}_i - \mu\mathbf{I})^{-2}\mathbf{g}_i} = \tau. \tag{53}$$

To reach a minimum, we choose μ to be less than the lowest eigenvalue. In the diagonal representation, the step then becomes

$$\tilde{s}_k = -\frac{\tilde{g}_k}{\tilde{H}_{kk} - \mu}. \tag{54}$$

D. Rational function

Another popular second-order optimization method is the rational function (RF) approach,[19] closely related to the eigenvector-following (EF) method.[20] The RF method is based on the construction of the following rational function:

$$E(\mathbf{s}_i)-E_i=\frac{\mathbf{g}_i^t\mathbf{s}_i+\frac{1}{2}\mathbf{s}_i^t\mathbf{H}_i\mathbf{s}_i}{1+\mathbf{s}_i^t\mathbf{S}\mathbf{s}_i}. \tag{55}$$

The matrix $\mathbf{S}$ can be chosen freely but is usually set equal to the unit matrix. Differentiation of Eq. (55) then yields

$$\begin{bmatrix}\mathbf{H}_i & \mathbf{g}_i\\ \mathbf{g}_i^t & 0\end{bmatrix}\begin{bmatrix}\mathbf{s}_i\\ 1\end{bmatrix}=\nu\begin{bmatrix}\mathbf{s}_i\\ 1\end{bmatrix}, \tag{56}$$

where the matrix on the left is the *augmented Hessian* $\mathbf{H}_i^\dagger$. The augmented step vector $\mathbf{s}_i^\dagger$ is an eigenvector of $\mathbf{H}_i^\dagger$ with eigenvalue ν. In the RF method, ν plays the role of the level-shift parameter μ in the trust-region Newton method. By using the eigenvector belonging to the lowest eigenvalue of the augmented Hessian, a step towards a minimum is ensured. Close to the stationary point, ν approaches zero and $\mathbf{s}_i$ approaches the Newton step.

In practice, the RF step is determined by diagonalizing $\mathbf{H}_i^\dagger$, selecting the eigenvector belonging to the lowest eigenvalue, which is scaled so that the last element becomes equal to one. The step is then obtained by removing this last element. The RF method can be combined with the idea of a trust radius, using τ as a limit either for the whole step length $|\mathbf{s}|$ or for the largest step component.

E. Geometrical DIIS

A third possibility is the geometrical version of the direct inversion of the iterative subspace (DIIS) method,[21] related to the DIIS algorithms normally used for wave-function optimizations. Although some testing of the geometrical DIIS (GDIIS) algorithm is presented later, we do not discuss this method further here.

F. Line searches

The step-control methods can be combined with line search,[14] a more accurate search for a minimum along the step vector

$$\mathbf{x}_{i+1}=\mathbf{x}_i+\alpha\mathbf{s}_i. \tag{57}$$

In practice, an exact line search is not attempted; rather, a partial (inexact) line search is carried out based on the information already available. From the energies and gradients at the current and previous points, a quartic polynomial can be fitted under the condition that the second derivative is positive at both points.[22] The minimum of this model is then taken as an update of the geometry. At this geometry, the gradient is determined by interpolation rather than calculation.

G. Image function

The methods presented so far are geared towards minimizations, enforcing an eigenvalue structure that resembles that of a minimum by the use of level-shift parameters μ and ν. To locate saddle points, we must modify our methods slightly. In particular, to make the level-shifted Newton method suitable for optimizing transition states, we introduce the idea of an image function.[23] Given some function f, we assume that an image function $\overline{f}$ can be constructed in such a way that one of the saddle points of f coincides with a minimum of $\overline{f}$. The saddle point of f is then determined by minimizing $\overline{f}$.

In the diagonal representation, the image function is easily introduced by changing the sign of the lowest eigenvalue and its corresponding gradient element

$$\tilde{\mathbf{g}}_b=\begin{bmatrix}-\tilde{g}_1\\ \vdots\\ \tilde{g}_n\end{bmatrix},\quad \tilde{\mathbf{H}}_b=\begin{bmatrix}-\tilde{H}_{11} & & 0\\ & \ddots & \\ 0 & & \tilde{H}_{nn}\end{bmatrix}. \tag{58}$$

A usual trust-region minimization is then performed with the image function, corresponding to maximizing the inverted mode and minimizing all other modes. In principle, any of the Hessian eigenvalues can be inverted, but only the lowest modes lead to low-lying transition states.

H. Partitioned rational function

The RF method can also be modified for saddle-point optimizations. For saddle-point optimizations, the rational function is partitioned, separating the reaction mode (to be maximized) from the transverse modes (to be minimized).[19] In this partitioned RF (PRF) method, the lowest eigenvalue is usually chosen to represent the reaction coordinate, giving us two independent level-shift parameters

$$\begin{bmatrix}\tilde{H}_{11} & \tilde{g}_1\\ \tilde{g}_1 & 0\end{bmatrix}\begin{bmatrix}\tilde{s}_1\\ 1\end{bmatrix}=\nu_{\mathrm{R}}\begin{bmatrix}\tilde{s}_1\\ 1\end{bmatrix}, \tag{59}$$

$$\begin{bmatrix}\tilde{H}_{22} & & 0 & \tilde{g}_2\\ & \ddots & & \vdots\\ 0 & & \tilde{H}_{nn} & \tilde{g}_n\\ \tilde{g}_2 & \cdots & \tilde{g}_n & 0\end{bmatrix}\begin{bmatrix}\tilde{s}_2\\ \vdots\\ \tilde{s}_n\\ 1\end{bmatrix}=\nu_{\mathrm{T}}\begin{bmatrix}\tilde{s}_2\\ \vdots\\ \tilde{s}_n\\ 1\end{bmatrix}. \tag{60}$$

If the resulting step exceeds the trust radius, it is scaled down as in the regular RF method.

I. Convergence criteria

Geometry optimizations are iterative procedures, making it necessary to define convergence criteria to be fulfilled when an acceptable geometry has been reached. What is considered acceptable clearly depends on the purpose of the optimization—just the calculation of the energy or the calculation of molecular properties. The latter usually requires tighter convergence.

A widely used set of criteria are those of Baker.[9] Baker's requirements are that no component of the gradient should be greater than $3.0\times10^{-4}\,E_{\mathrm{h}}/a_0$, and that either the change in energy from the previous iteration should be less than $1.0\times10^{-6}\,E_{\mathrm{h}}$ or no step components should be greater than $3.0\times10^{-4}\,a_0$. The use of these thresholds normally yields energies within $1.0\times10^{-6}\,E_{\mathrm{h}}$.

In DALTON, we have a flexible system of three adjustable thresholds: one for the energy change, one for the norm of the gradient, and one for the norm of the step. The number of criteria that must be fulfilled can also be adjusted, the default

TABLE II. The performance of Cartesian, redundant internal and delocalized internal coordinates, with exact and updated Hessians. The numbers quoted are the total number of iterations needed to minimize the 30 test cases in the STO-3G basis with Baker's convergence criteria. The level-shifted Newton method controls the steps, no line searches are performed, and the Hessian updating is done with the BFGS formula.

	Cartesian coordinates	Redundant internal coordinates	Delocalized internal coordinates
Exact Hessians	125	111	114
Initial Hessian exact	200	154	176
Initial model Hessian	303	213	219
Initial Cartesian unit matrix	768[a]	848	868

[a]Benzidine did not converge to the correct stationary point.

being two. However, for easy comparison with other published results, we use Baker's criteria in the present paper.

V. RESULTS AND DISCUSSION

A. Coordinate systems

First, we consider how different coordinate systems affect the optimization. Cartesian coordinates, redundant internal coordinates, and delocalized internal coordinates have all been investigated with different choices of Hessians—see Table II. Except in the second-order optimizations (exact Hessians throughout), the BFGS updating formula was used. Also, the steps were controlled by the level-shifted Newton method, with no line searches employed. The number of iterations given in the table corresponds to the total number of iterations needed for all 30 test molecules. Initial geometries and energies are listed in the supplementary material.[24]

These results show that the factor that has the most profound effect on the performance is the treatment of the Hessian, as discussed in the next subsection. Among the coordinate systems, the redundant internal coordinates perform best, closely followed by delocalized internal coordinates. Use of Cartesian coordinates leads to slow convergence except when a Cartesian unit matrix is used in the first iteration. In this case, however, benzidine failed to converge to the correct stationary point; also, the difference between the Cartesian and internal coordinates arises mostly from one system (dimethylpentane). Because of the slightly inferior performance of the delocalized internal coordinates compared with the redundant internal coordinates, we shall in the following concentrate on the redundant internal coordinates, using the Cartesian coordinates for reference.

B. Choice of Hessian

In assessing the relative performance of the first- and second-order methods, it is not sufficient to compare the total number of iterations since the cost of calculating the energy, gradient and Hessian is significantly higher than that of calculating just the energy and the gradient. A more relevant measure of performance is therefore the total CPU time required for convergence. Since relative CPU times are more important than absolute CPU times, we shall in the following use the CPU time of the second-order optimization in Cartesian coordinates (arbitrarily set to 100) as the reference for each of the 30 molecules.

The average CPU times for different Hessians in Cartesian and redundant internal coordinates are given in Table III. Since the total CPU time is dominated by the two or three largest molecules, it it was decided that the averages of relative CPU times constitute a more balanced measurement. On the other hand, since the savings are more important for the larger molecules than for the smaller ones, we present the average relative CPU times as triplets representing all 30 molecules, the 20 largest molecules, and the 10 largest molecules, respectively. Of the three numbers, the last one gives

TABLE III. Various choices of exact, updated and approximate Hessians. Total iterations are the number of steps needed to minimize all of the 30 test cases with Baker's convergence criteria in the STO-3G basis. All CPU times are relative to the CPU usage of a second-order minimization in Cartesian coordinates, which for each molecule is set equal to 100. The triplets indicate the average CPU time of all 30 molecules, of the 20 largest molecules, and of the 10 largest molecules, respectively. The level-shifted Newton method controls the steps, no line searches are performed, and BFGS updating is used.

	Cartesian coordinates		Redundant internal coordinates	
	Total iter.	Avg. CPU time	Total iter.	Avg. CPU time
Exact Hessians	125	100/100/100	111	98/91/82
Initial Hessian exact	200	58/52/48	154	58/48/40
Initial model Hessian	311	61/61/57	213	50/40/36
Initial simple model Hessian[a]	319	62/60/56	270	59/48/43
Initial red. internal unit matrix	583	104/108/98	499	94/84/77
Initial Cartesian unit matrix	768	128/134/138	848	149/153/163

[a]Diagonal Hessian in redundant internal coordinates, where bond lengths are given the value 0.5, bond angles 0.2, and dihedral angles 0.1.

TABLE IV. Various choices of exact, updated, and approximate Hessians. Total iterations are the number of steps needed to minimize all of the 30 test cases with Baker's convergence criteria in the 6-31G* basis. All CPU times are relative to the CPU usage of a second-order minimization in Cartesian coordinates, which for each molecule is set equal to 100. The triplets indicate the average CPU time of all 30 molecules, the 20 largest molecules, and the 10 largest molecules, respectively. The level-shifted Newton method controls the steps, no line searches are performed, and the updating of approximate Hessians is done with the BFGS updating formula.

	Cartesian coordinates		Redundant internal coordinates	
	Total iter.	Avg. CPU time	Total iter.	Avg. CPU time
Exact Hessians	134	100/100/100	115	95/87/77
Initial Hessian exact	198	43/37/32	174	43/35/28
Initial model Hessian	294	33/28/22	214	29/20/16
Initial simple model Hessian	323	35/30/24	277	33/26/21

the best indication of the total absolute CPU cost of all 30 optimizations.

We note that, for a small basis such as STO-3G, the quasi-Newton method does not guarantee CPU savings relative to the Newton method—since the Hessians are relatively inexpensive in this basis, Newton's method outperforms the quasi-Newton method when a crude estimate of the Hessian is used (Cartesian unit matrix). With better estimates of the Hessian, however, the quasi-Newton method gives a significant performance gain. Overall, the best performance is achieved by using the model Hessian in combination with redundant internal coordinates. The optimization of the 10 largest molecules is then on average 64% faster than Newton' s method (50% for all 30 molecules), even though the total number of iterations increases significantly. We emphasize that, unlike in the work by Lindh *et al.*,[2] the model Hessian is calculated only at the initial geometry. We also explored the possibility of calculating the model Hessian at each iteration and updating using previous gradients, but this gave no improvements over the simpler scheme presented here.

The simple model Hessian—which is diagonal with elements 0.5, 0.2 and 0.1 for bond lengths, bond angles and dihedral angles, respectively—performs surprisingly well in comparison with the full model Hessian. It is also worth noting that, in redundant internal coordinates, the use of the model Hessian is only a few percent faster than the calculation of the initial Hessian exactly; in Cartesian coordinates, it is the other way round. Again, this is due to cheap Hessians. As we shall see later, the situation is different in a larger basis.

In general, the CPU savings increase with the size of the molecule, particularly in redundant internal coordinates. Part of the reason for this behavior is that, for the small molecules, there is a small but significant overhead associated with setting up the internal coordinate system and carrying out the necessary transformations (a tenth of a second or so out of a CPU time of a few seconds). For the larger systems, this overhead becomes insignificant.

The small STO-3G basis is not sufficiently flexible for most production work. The performance of the different methods has, therefore, been tested with the larger and more realistic 6-31G* basis set. The number of iterations is similar to (but slightly higher than) that of the STO-3G basis—see Table IV.

More interestingly, the differences between various Hessians now become larger. Again, the best choice is the model Hessian in combination with redundant internal coordinates. For the 10 largest molecules, there is a 84% reduction in CPU cost relative to the Newton method in Cartesian coordinates. Moreover, for the larger molecules, the model Hessian reduces computational time by about a factor of two

TABLE V. Performance of the various Hessian updating algorithms. The total number of iterations needed to minimize all of the 30 test cases in the STO-3G basis with Baker's convergence criteria are shown. The level-shifted Newton method controls the steps and no line searches are performed.

	Cartesian coordinates			Redundant internal coordinates		
Updating algorithm	Initial Hessian exact	Initial model Hessian	Initial simple model Hess.	Initial Hessian exact	Initial model Hessian	Initial simple model Hess.
SR1 (MS)	317	441	456	182	257	331
PSB	234	386	403	175	257	286
Bofill	236	374	396	169	222	281
DFP	240	429	591[a]	158	216	545[b]
BFGS	200	311	319	154	213	270
BFGS/SR1	234	324	317	158	206	262

[a]Histidine was stopped after 150 iterations without fulfilling the convergence criteria. At this point, the molecule was about 20 μE_h from the minimum energy, but convergence was slow.

[b]Both histidine and menthone were stopped after 150 iterations, neither of them fulfilling the convergence criteria. Both molecules were close to their respective minima, but convergence was extremely slow.

TABLE VI. Various algorithms for controlling the step. The numbers quoted are the total number of iterations needed to minimize the 30 test cases in the STO-3G basis with Baker's convergence criteria. No line searches are performed and the updating of approximate Hessians is done with the BFGS updating formula.

	Cartesian coordinates			Redundant internal coordinates		
	Level-shifted Newton	RF	Geometrical DIIS	Level-shifted Newton	RF	Geometrical DIIS
Exact Hessians	125	126	126	111	113	113
Initial Hessian exact	200	199	222	154	156	176
Initial model Hessian	311	307	321	213	191	208
Initial simple model Hessian	319	305	324	270	253	270

compared with an initial exact Hessian. These savings will be even larger for larger basis sets, demonstrating the usefulness of the model Hessian in the quasi-Newton scheme.

Another issue that warrants attention is the characterization of stationary points. It is important to note that there is no way to distinguish a minimum from a saddle point without calculating the Hessian. Indeed, five of the 30 test molecules end up at a saddle point rather than at a minimum. Mostly, this happens since the molecules are constrained by symmetry; the true minima and their correct symmetry can be found in the supplementary material.[24] In the quasi-Newton method, there is no way do detect such symmetry-broken solutions. By contrast, our implementation of Newton's method automatically detects such solutions, breaking the symmetry when necessary. For purposes of comparison with the quasi-Newton method, this ability to break symmetry was turned off during the Newton iterations.

C. Hessian updating

Various updating formulas have been tested extensively by others. In general, the BFGS formula has been found to be the best choice for minimizations. Nevertheless, we have here investigated their behavior briefly. In particular, we would like to explore our proposed BFGS/SR1 combination formula (fashioned after Bofill's update). Results for the different formulas are presented in Table V. Again, Cartesian and redundant internal coordinates are examined using three different initial Hessians (the exact Hessian, the model Hessian, and the simple model Hessian). With the exception of the Davidon-Fletcher-Powell (DFP) formula, which performed poorly with the simple model Hessian, all minimizations ended up at the correct stationary point.

As seen in previous tables, the use of an exact initial Hessian reduces the number of iterations significantly, albeit at the cost of a much more expensive first iteration. Again, the model Hessian performs better than the simple model. The performance of the PSB and Bofill updates are quite good, although they cannot compete with the BFGS and BFGS/SR1 updates. The two latter methods are similar but BFGS performs significantly better than BFGS/SR1 when using an initial exact Hessian in Cartesian coordinates; otherwise, the BFGS/SR1 update performs better than the BFGS update. In conclusion, there does not appear to be any strong reason for abandoning the BFGS formula in favor of the more complicated BFGS/SR1 combination.

D. Step control

The final factor we would like to investigate is the step control algorithm. For a comparison of the level-shifted Newton step, the RF step, and the GDIIS step, see Table VI. Again, both Cartesian and redundant internal coordinates are employed in combination with different Hessians.

We first note that methods perform almost equally well with exact Hessians, but that the variations are larger with approximate Hessians. In addition to being the most complicated method, the GDIIS approach performed the worst. In the following, therefore, we concentrate on the level-shifted Newton and RF methods. It is their performance with model Hessians that interests us the most. With such Hessians, the RF method outperforms the other methods. With exact Hessians, on the other hand, there is some indication that the level-shifted Newton method performs best.

With redundant internal coordinates and the model Hessian, the total number of iterations needed to optimize all 30

TABLE VII. Comparison of regular redundant internal coordinates and extra-redundant internal coordinates. The numbers quoted are the total number of iterations needed to minimize the 30 test cases in the STO-3G basis with Baker's convergence criteria. No line searches are performed and the updating of approximate Hessians is done with the BFGS updating formula. The RF method controls the step.

	Regular redundant internal coordinates		Extra-redundant internal coordinates	
	Level-shifted Newton	RF	Level-shifted Newton	RF
Exact Hessians	111	113	120	117
Initial Hessian exact	154	156	169	174
Initial model Hessian	214	192	202	185

TABLE VIII. Comparison of various dynamic and fixed trust radii, with and without a partial line search. The numbers quoted are the total number of iterations needed to minimize the 30 test cases in the STO-3G basis with Baker's convergence criteria. All optimizations are performed using the extra-redundant internal coordinates with the RF method for step control. The model Hessian is used for the initial Hessian and updating is performed with the BFGS formula.

Trust radius	Line search	0.2	0.3	0.4	0.5	0.6	0.7
Dynamic	None	198	188	185	185	185	186
Dynamic	Partial	198	189	185	185	187	187
Fixed	None	218	192	191	274[a]	275[a]	276[a]
Fixed	Partial	218	193	191	274[a]	274[a]	274[a]

[a]1.3.5-trisilacyclohexane was stopped after 100 iterations. The optimization was overshooting the minimum and convergence was extremely slow.

molecules is only 191 iterations—the best quasi-Newton result so far. From now on, we abandon the Cartesian coordinates (which are clearly outperformed), considering instead the proposed extra-redundant internal coordinates. To compare directly with the literature, we also turn off the step-rejection mechanism. Essentially, this means that a step resulting in a significant increase in the energy will be accepted, although the trust radius will be decreased accordingly.

Table VII shows the results for the regular and the extra-redundant coordinates, using the level-shifted Newton method and the RF method. With exact Hessians (initially or throughout the optimization), performance is slightly degraded by this extra redundancy. By contrast, for the quasi-Newton method with the initial model Hessian, convergence is nicely improved—the total number of iterations is now down to a respectable 185 iterations.

Most likely, this speed-up occurs since, with the additional internal coordinates, a more accurate model Hessian is initially constructed. The results in Table VII indicate that there is nothing to be gained by using the extra coordinates throughout the minimization. It would obviously be possible to set up the initial model Hessian in an extra-redundant coordinate system and then transform it to the regular redundant internal coordinates for the actual minimization. However, we have not tried this possibility, noting that it would require additional coordinate transformations and therefore more overhead. The use of a single internal coordinate system is simpler and more attractive.

The results reported so far have been calculated with a trust radius that is adjusted at each geometry iteration (based on how closely the second-order model resembles the true surface), with an initial trust radius of 0.5 a_0. Table VIII shows the effect of altering the initial trust radius and of using a fixed trust radius. Furthermore, the effect of employing a partial line search has been investigated. These tests were run with the extra-redundant internal coordinates, the initial model Hessian, the BFGS updating scheme, and the RF step control. The goal was to see if some tweaking of the method could improve convergence further.

The trust-region method is remarkably stable with respect to the initial trust radius. Reduction of the initial trust radius to 0.2 slows the minimizations down somewhat but the dynamic update scheme rapidly increases the trust radius, ensuring a fast convergence. With a fixed trust radius, by contrast, there is a more direct connection between the trust radius and performance, a small trust radius giving slower convergence. Unfortunately, when the fixed trust radius is increased to 0.5 or more, 1,3,5-trisilacyclohexane failed to converge within 100 iterations—if this molecule had converged "normally," this initial trust radius would have yielded the best result overall (182–184 iterations).

TABLE IX. Comparison of regular redundant internal coordinates and extra-redundant internal coordinates in the 6-31G* basis. The numbers quoted are the total number of iterations needed to minimize the 30 test cases with Baker's convergence criteria, with and without a partial line search. Hessian updating was performed with the BFGS formula.

	Regular redundant internal coordinates		Extra-redundant internal coordinates	
	Level-shifted Newton	RF	Level-shifted Newton	RF
None	217	213	207	198
Partial	220	208	205	198

Eckert *et al.* have reported good results with a method where the individual components of the RF step (rather than the whole step) are scaled.[3] This approach was tested but the results were virtually identical to the results with scaling of the whole step. We, therefore, prefer the simpler approach, with a scaling of the whole step. We also briefly revisited the BFGS/SR1 combination formula, obtaining the same result as before—that is, 185 iterations. From Table VIII, it is also apparent that adding a partial line search has no effect—if anything, the results are poorer. We therefore view the addition of a partial line-search mechanism as an unnecessary complication.

In conclusion, we use the dynamic trust-radius scheme with an initial trust radius of 0.5 a_0, the extra-redundant internal coordinates, the BFGS formula, and the RF method for quasi-Newton optimizations with the model Hessian. To verify that these choices are sensible, we carried out additional calculations in the 6-31G* basis. The results in Table IX confirm that the best performance is obtained with extra-redundant internal coordinates and RF step control. Partial line search has no effect for this method but gives a slight improvement for the other methods. As previously found, convergence is slightly slower with the 6-31G* basis than with the STO-3G basis. Nevertheless, it is reassuring to see the excellent performance—that is, 198 iterations for all 30 molecules—since this confirms that our choices and adjustments are not specific to the STO-3G basis.

E. Comparison with the literature

Several authors have tested methods for minimization in the STO-3G basis. Their results are summarized in Table X. The best performance was obtained by Eckert *et al.*, using natural internal coordinates, the model Hessian, and the RF step. All 30 molecules were minimized in 196 iterations, although one molecule (2-hydroxybicyclopentane) ended up at a higher stationary point. By contrast, our scheme requires 185 iterations, converging all molecules to the correct stationary point (in the given symmetry, as discussed previ-

TABLE X. The most efficient method of this work compared with results from the literature.

	STO-3G basis				6-31G* basis
Molecule	This work	Eckert *et al.* (Ref. 3)	Lindh *et al.* (Ref. 2)	Baker (Ref. 9)	This work
Water	4	4	4	6	5
Ammonia	5	6	5	6	5
Ethane	3	4	4	5	4
Acetylene	4	6	5	6	4
Allene	4	4	5	5	5
Hydroxysulphane	7	7	8	8	6
Benzene	3	3	3	4	3
Methylamine	4	5	5	6	5
Ethanol	4	5	5	6	5
Acetone	4	5	5	6	4
Disilyl ether	8	9	11	8	11
1,3,5-trisilacyclohexane	9	6	8	8	8
Benzaldehyde	4	5	5	6	5
1,3-difluorobenzene	4	5	5	5	5
1,3,5-trifluorobenzene	4	4	4	5	5
Neopentane	4	4	5	5	4
Furan	5	6	7	8	5
Naphtalene	5	6	6	5	5
1,5-difluoronaphtalene	5	6	6	6	5
2-hydroxybicyclopentane	9	9[a]	10	15	9
ACHTAR10	8	9	8	12	9
ACANIL01	7	8	8	8	6
Benzidine	9	7	10	9	9
Pterin	8	9	9	10	9
Difuropyrazine	6	7	7	9	7
Mesityl oxide	5	6	6	7	5
Histidine	16	14	20	19	16
Dimethylpentane	9	10	10	12	9
Caffeine	6	7	7	12	7
Menthone	12	10	14	13	13
Sum	185	196	215	240	198

[a]This molecule did not converge to the correct stationary point.

ously). The performance in the 6-31G* basis is also good, although a few more iterations (198) are needed for convergence. However, the second-order method also requires more iterations in this basis. The difference is therefore probably a consequence of different potential-energy surfaces rather than any preference of the method for the smaller basis.

F. Transition-state optimization

Results for Cartesian coordinates, redundant internal coordinates, and the extra-redundant internal coordinates are presented in Table XI. Note that only the level-shifted Newton method (by the use of an image function) and the RF (by partitioning) can be modified to cope with saddle point optimizations.

Calculations were carried out using a second-order method and a quasi-Newton method with the initial Hessian calculated exactly. The former method requires less iterations but, in terms of CPU time, it is more efficient to use updated Hessians. For transition states, we use Bofill's update formula.

Regular redundant internal coordinates give best results, followed closely by the extra-redundant ones. Apparently,

TABLE XI. Transition-state optimizations using various coordinate systems and step control algorithms. The numbers quoted are the total number of iterations needed to optimize the 15 test cases in the 3-21G basis with Baker's convergence criteria. Hessian updating was carried out with Bofill's formula.

	Cartesian coordinates		Regular redundant internal coordinates		Extra-redundant internal coordinates	
	Image	PRF	Image	PRF	Image	PRF
Exact Hessians	92	108	86	96	100[a]	102
Initial Hessian exact	188	201[b]	162	187[c]	162	198

[a]CH_2CHOH converged to a lower first-order saddle point with an energy of $-152.053\,44E_h$.
[b]s-tetrazine converged to a third-order saddle point.
[c]CH_2CHOH converged to a different saddle point of energy $-151.916\,44E_h$.

TABLE XII. Transition-state optimizations in Cartesian and redundant internal coordinates, using different updating formulas in the 3-21G basis. The initial Hessian is exact and Baker's convergence criteria are employed. The number of iterations is shown for each system. Dots indicate that the system did not converge within 50 iterations or that it converged to some other stationary point.

	Cartesian coordinates			Regular redundant internal coordinates		
	SR1	PSB	Bofill	SR1	PSB	Bofill
$HCN \rightleftharpoons HNC$	33	11	10	10	9	9
$HCCH \rightleftharpoons CCH_2$	11	8	8	8	7	7
$H_2CO \rightleftharpoons H_2+CO$	13	13	14	11	14	13
Parent Diels-Alder	17	18	17	⋯	17	15
s-tetrazine $\rightleftharpoons 2HCN+N_2$	⋯	19	20	9	10	9
$CH_3CH_3 \rightleftharpoons CH_2CH_2+H_2$	10	10	10	10	9	8
$CH_3CH_2F \rightleftharpoons CH_2CH_2+HF$	11	11	11	11	16	15
$CH_2CHOH \rightleftharpoons CH_3CHO$	19	18	16	⋯	20	19
Silylene insertion	⋯	14	17	13	15	12
$HNCCS \rightleftharpoons HNC+CS$	31	21	22	⋯	16	11
Rotational TS in acrolein	7	5	5	⋯	5	5
$HCONHOH \rightleftharpoons HCOHNHO$	10	8	7	5	5	5
$HNC+H_2 \rightleftharpoons H_2CNH$	10	8	8	14	8	8
$H_2CNH \rightleftharpoons HCNH_2$	19	18	13	23	25	18
$HCNH_2 \rightleftharpoons HCN+H_2$	11	10	10	8	8	8
Sum	⋯	192	188	⋯	184	162

the extra coordinates slow down convergence in these optimizations. Cartesians, as usual, perform worst, although the difference is less pronounced than for minimizations. Delocalized internal coordinates were briefly investigated but their performance is somewhat inferior to the redundant internal coordinates.

It is a general trend that the image method (the modified level-shifted Newton) performs better than the PRF method. More importantly, with the PRF method, some systems fail to converge to the correct stationary point. For saddle points, therefore, the image method is our first choice.

TABLE XIII. Transition-state optimizations in Cartesian and redundant internal coordinates, using different updating formulas in the 3-21G basis. The model Hessian is used as the initial Hessian and Baker's convergence criteria are used. The number of iterations is shown for each system. Dots indicate that the system did not converge within 50 iterations or that it converged to some other stationary point.

	Cartesian coordinates			Regular redundant internal coordinates		
	SR1	PSB	Bofill	SR1	PSB	Bofill
$HCN \rightleftharpoons HNC$	⋯	14	13	10	9	9
$HCCH \rightleftharpoons CCH_2$	23	⋯	27	13	10	9
$H_2CO \rightleftharpoons H_2+CO$	41	18	18	11	12	11
Parent Diels-Alder	⋯	⋯	⋯	⋯	⋯	⋯
s-tetrazine $\rightleftharpoons 2HCN+N_2$	⋯	⋯	⋯	14	18	16
$CH_3CH_3 \rightleftharpoons CH_2CH_2+H_2$	13	14	14	14	13	14
$CH_3CH_2F \rightleftharpoons CH_2CH_2+HF$	15	13	12	20	19	12
$CH_2CHOH \rightleftharpoons CH_3CHO$	⋯	⋯	19	⋯	⋯	⋯
Silylene insertion	38	25	21	⋯	15	19
$HNCCS \rightleftharpoons HNC+CS$	⋯	⋯	38	15	19	21
Rotational TS in acrolein	10	18	15	38	13	12
$HCONHOH \rightleftharpoons HCOHNHO$	⋯	26	29	9	10	7
$HNC+H_2 \rightleftharpoons H_2CNH$	18	13	23	27	19	23
$H_2CNH \rightleftharpoons HCNH_2$	⋯	30	23	⋯	21	24
$HCNH_2 \rightleftharpoons HCN+H_2$	16	13	16	⋯	15	15

TABLE XIV. Comparison of Cartesian coordinates and regular redundant internal coordinates in the 6-31G* basis. The numbers quoted are the total number of iterations needed to optimize the 15 test cases with Baker's convergence criteria. Hessian updating was done with Bofill's formula.

	Cartesian coordinates		Redundant internal coordinates	
	Image	PRF	Image	PRF
Exact	96	103	84	90
Exact initial	188	177[a]	155	157

[a]s-tetrazine converged to a fourth-order saddle point with an energy of $-294.224\,53E_h$.

Not all updating formulas are suitable for transition-state optimizations, so only the SR1 update, the PSB update, and Bofill's update were explored. Optimizations using these updates in Cartesian and regular redundant internal coordinates are found in Table XII, where the initial Hessian is calculated analytically. Since many optimizations failed to converge, the results for each system is listed separately—the total number of iterations is given only for those methods that converged all 15 tests. The SR1 update shows the most erratic behavior—for some systems, it works as well or even better than the other updates; for other systems, convergence is slow; and, in a few cases, it does not converge. The PSB and Bofill updates converge all systems to the correct stationary point, the latter being the fastest, particularly in redundant internal coordinates. These observations are in good agreement with what others have found.[17]

Encouraged by the success of the model Hessian for minimizations, we investigated its use for saddle points—see Table XIII. Unfortunately, no combination of coordinate system and update managed to converge all systems. With Bofill's update, however, only two systems failed to converge—for the remaining systems, convergence is fairly rapid

TABLE XV. The most efficient method of this work compared with results from the literature.

	3-21G basis		6-31G* basis
	This work	Baker and Chan (Ref. 4)	This work
$HCN \rightleftharpoons HNC$	9	9	10
$HCCH \rightleftharpoons CCH_2$	7	8	9
$H_2CO \rightleftharpoons H_2+CO$	13	13	11
Parent Diels-Alder	15	13	16
s-tetrazine $\rightleftharpoons 2HCN+N_2$	9	14	10
$CH_3CH_3 \rightleftharpoons CH_2CH_2+H_2$	8	12	10
$CH_3CH_2F \rightleftharpoons CH_2CH_2+HF$	15	11	12
$CH_2CHOH \rightleftharpoons CH_3CHO$	19	13	17
Silylene insertion	12	7	15
$HNCCS \rightleftharpoons HNC+CS$	11	10	7
Rotational TS in acrolein	5	4	5
$HCONHOH \rightleftharpoons HCOHNHO$	5	5	6
$HNC+H_2 \rightleftharpoons H_2CNH$	8	8	11
$H_2CNH \rightleftharpoons HCNH_2$	18	14	7
$HCNH_2 \rightleftharpoons HCN+H_2$	8	9	9
Sum	162	150	155

although slower than with exact initial Hessians. In fact, it is remarkable that we can perform the relatively difficult task of locating the transition state for most of the 15 systems without explicit calculation of second derivatives, in spite of the fact that the model Hessian was explicitly developed for minimizations.[2] The model Hessian thus opens up the possibility of efficiently locating transition states using high-level methods, where no analytical Hessian is available.

Finally, we have checked the performance of saddle-point methods in the 6-31G* basis. As seen from Table XIV, the conclusions based on the 3-21G basis still hold—that is, the redundant internal coordinates and the image method yields the best results. The PRF method performs worse, failing to converge one of the systems correctly. In general, the optimization in the larger basis requires fewer iterations than in the STO-3G basis in Table XI.

The only work suitable for comparison is that of Baker *et al.*[4]—see Table XV, where a comparison is made with our results. Our scheme using redundant internal coordinates is slightly slower than Baker's scheme using Z-matrices. However, whereas our coordinate system is set up automatically, the Z-matrix must be set up manually.

VI. CONCLUSIONS

We have designed schemes for the efficient localization of minima and transition states. Because of the different nature of these two types of optimization, the ideal methods differ slightly. For minimization, we recommend the use of a extra-redundant internal coordinate system (with extra bonds added to the regular redundant system) in combination with the model Hessian and BFGS update scheme. For step control, the RF method should be used together with a dynamic trust radius. For transition-state optimizations, we recommend calculating the initial Hessian analytically to ensure robustness. Subsequent updates should be performed using Bofill's formula. The optimization is efficiently carried out in the regular redundant (rather than extra-redundant) internal coordinates, using the level-shifted Newton method with an image function for step control.

ACKNOWLEDGMENT

This work has received support through grants of computer time from the Research Council of Norway (Grant No. NN1118K).

[1] C. Peng, P. Y. Ayala, H. B. Schlegel, and M. J. Frisch, J. Comput. Chem. **17**, 49 (1996).
[2] R. Lindh, A. Bernhardsson, G. Karlström, and P.-Å. Malmqvist, Chem. Phys. Lett. **241**, 423 (1995).
[3] F. Eckert, P. Pulay, and H.-J. Werner, J. Comput. Chem. **18**, 1473 (1997).
[4] J. Baker and F. Chan, J. Comput. Chem. **17**, 888 (1996).
[5] Ö. Farkas and H. B. Schlegel, J. Chem. Phys. **109**, 7100 (1998).
[6] Ö. Farkas and H. B. Schlegel, J. Chem. Phys. **111**, 10806 (1999).
[7] W. J. Hehre, L. Radom, P. R. v. Schleyer, and J. A. Pople, *Ab initio Molecular Orbital Theory* (Wiley, New York, 1986).
[8] J. Baker and J. Hehre, J. Comput. Chem. **12**, 606 (1991).
[9] J. Baker, J. Comput. Chem. **14**, 1085 (1993).
[10] P. Pulay, G. Fogarasi, F. Pang, and J. E. Boggs, J. Am. Chem. Soc. **101**, 2550 (1979).
[11] G. Fogarasi, X. F. Zhou, P. W. Taylor, and P. Pulay, J. Am. Chem. Soc. **114**, 8191 (1992).
[12] J. Baker, A. Kessi, and B. Delley, J. Chem. Phys. **105**, 192 (1996).
[13] E. B. W. Jr., J. C. Decius, and P. C. Cross, *Molecular Vibrations—The Theory of Infrared and Raman Vibrational Spectra* (Dover, New York, 1980).
[14] R. Fletcher, *Practical Methods of Optimization Vol.1—Unconstrained Optimization* (Wiley, New York, 1981).
[15] T. Helgaker, in *Lecture Notes in Quantum Chemistry*, edited by B. O. Roos (Springer-Verlag, Berlin, 1992).
[16] H. B. Schlegel, in *Modern Electronic Structure Theory Part 1*, edited by D. R. Yarkony (World Scientific, Singapore, 1995).
[17] J. M. Bofill, J. Comput. Chem. **15**, 1 (1994).
[18] H. J. A. Jensen, P. Jørgensen, and T. Helgaker, J. Chem. Phys. **85**, 3917 (1986).
[19] A. Banerjee, N. Adams, J. Simons, and R. Shephard, J. Phys. Chem. **89**, 52 (1985).
[20] J. Baker, J. Comput. Chem. **7**, 385 (1986).
[21] P. Csàszàr and P. Pulay, J. Mol. Struct. **114**, 31 (1984).
[22] H. B. Schlegel, J. Comput. Chem. **3**, 214 (1982).
[23] T. Helgaker, Chem. Phys. Lett. **182**, 503 (1991).
[24] See EPAPS Document No. E-JCPSA6-117-303244 for geometries and energies of all test molecules. A direct link to this document may be found in the online article's HTML reference section. The document may also be reached via the EPAPS homepage (http://www.aip.org/pubservs/epaps.html) or from ftp.aip.org in the directory /epaps/. See the EPAPS homepage for more information.