JCTC Journal of Chemical Theory and Computation

Letter

pubs.acs.org/JCTC

Reaction Path Optimization without NEB Springs or Interpolation Algorithms

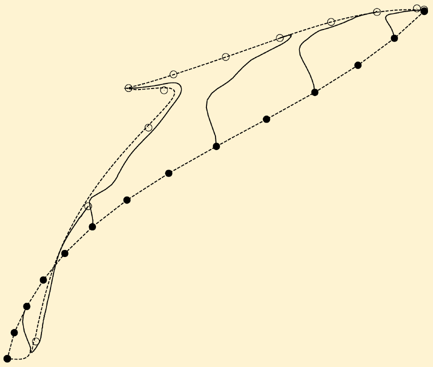
P. Plessow*

BASF SE, GVM/M - B009, D-67056 Ludwigshafen, Germany

CaRLa (Catalysis Research Laboratory), Im Neuenheimer Feld 584, D-69120 Heidelberg, Germany

S *Supporting Information*

ABSTRACT: This letter describes a chain-of-states method that optimizes reaction paths under the sole constraint of equally spaced structures. In contrast to NEB and string methods, it requires no spring forces, interpolation algorithms, or other heuristics to control structure distribution. Rigorous use of a quadratic PES allows calculation of an optimization step with a predefined distribution in Cartesian space. The method is a formal extension of single-structure quasi-Newton methods. An initial guess can be evolved, as in the growing string method.



Locating transition states is one of the most challenging geometry optimization problems in theoretical chemistry. Local search algorithms such as the Trust Region Image Minimization (TRIM)[1] typically work very well if they are started from a good approximation to the transitions state. Obtaining this approximation is the goal of self-consistent optimization of the reaction path (RP). Methods that use reactant and product structure to compute the RP are often referred to as "double-ended" methods or, if the RP is discretized, "chain-of-states" methods.[2−4]

The RP connects reactant and product, its highest point being the transition state. It is a steepest descent path, which means that its tangent **t** is always parallel to the gradient **g**.[5] The RP is in actual calculations discretized by a finite number of structures n. The tangents are parallel to the gradients for all structures $i = 1, ..., n$. Assuming normalized tangents $\mathbf{t}_i^{\mathrm{T}}\mathbf{t}_i = 1$, this can be written as

$$a_i\mathbf{t}_i = \mathbf{g}_i \tag{1}$$

$$0 = \mathbf{g}_i^{\perp} = (1 - \mathbf{t}_i\mathbf{t}_i^{\mathrm{T}})\mathbf{g}_i \tag{2}$$

Here, a_i is a constant and $\mathbf{g}_i^{\perp}$ is the projected gradient. Several approximations for the tangents $\mathbf{t}_i$ exist,[4,6] usually using finite difference schemes. Due to the approximate tangents, the discretized RP can in general only converge to an approximate RP.

Solving eqs 1 and 2 for a finite number of structures on an approximate path gives rise to two main problems:

1. computing an optimization step, $\delta\mathbf{x}_i$, that minimizes $\mathbf{g}_i^{\perp}$
2. restricting the structures to have a certain distribution along the path

The second point is important because minimization of $\mathbf{g}_i^{\perp}$ can only be meaningful if the structures are a reasonable representation of the path. The most common methods, the Nudged Elastic Band (NEB)[4,6] and String Method (SM),[7] are characterized by the way they address this problem and prevent the structures from "sliding" down the RP toward products and reactants during the optimization. The NEB achieves equal spacing by manipulating the forces acting on the structures. Artificial spring forces between adjacent structures are introduced that act along the tangents and together with $-\mathbf{g}_i^{\perp}$ give a total force $\mathbf{f}_i$. Upon convergence ($\mathbf{f}_i = \mathbf{g}_i^{\perp} = \mathbf{0}$), the structures solve eq 2 and are equally spaced. The SM combines an optimization step with subsequent redistribution of the structures on a path obtained by interpolation, for example, with cubic splines. This requires and introduces additional approximations to the path between the structures. Convergence of NEB and SM depend upon their tools introduced to restrict the structures to a certain distribution. The choice of spring forces or interpolation algorithm therefore determines the performance of these methods and is an active field of research.[8,9]

The main problem in optimizing a discretized RP is the tangents which couple the coordinates of different structures and change during the optimization. The usual approach in NEB and SM calculations is to compute the tangents at the beginning of each optimization cycle for the current geometry. The projected gradient and optimization step is then calculated on the basis of these tangents. This procedure does not take into account that the tangents of the current geometry and the

Received: October 31, 2012
Published: February 25, 2013

dx.doi.org/10.1021/ct300951j | *J. Chem. Theory Comput.* 2013, 9, 1305−1310

tangents of the geometry after the optimization step will in general be different. Tangents obtained within this approximation will in the following be referred to as "fixed tangents." In contrast to time-demanding electronic-structure calculations, the tangents are usually analytically known as functions of the coordinates and are readily calculated. It is therefore desirable to calculate an optimization step that is based on the tangents of the structures at the optimized geometries to include the effect of tangent change. As the optimization step depends on the tangent and vice versa, these tangents will be referred to as "self-consistent tangents." Self-consistent tangents have already been used for NEB[10] and string methods.[11] Choosing fixed or self-consistent tangents can only affect the performance of the optimization, while the final result is entirely determined by the used tangent formula.

Depending on the employed scheme, the tangents $\mathbf{t}_i$ will depend on the Cartesian coordinates of structure i, $\mathbf{x}_i$, and at least on one additional structure, or even on energies.[6] In the following, the tangents will therefore be assumed to be a function of all structures:[12]

$$\mathbf{t}_i = \mathbf{t}_i(\mathbf{x}_1, ..., \mathbf{x}_n) \tag{3}$$

While $\mathbf{g}_i$ is the gradient of the Potential Energy Surface (PES), $V(\mathbf{x}_i)$, and therefore is a function only of $\mathbf{x}_i$, the projected gradient, $\mathbf{g}_i^{\perp}$, will, like $\mathbf{t}_i$, be a function of all structures. It can therefore not be a gradient of a potential function of $\mathbf{x}_i$, $\tilde{V}(\mathbf{x}_i)$:

$$\mathbf{g}_i(\mathbf{x}_i) = \nabla V(\mathbf{x}_i) \tag{4}$$

$$\mathbf{g}_i^{\perp}(\mathbf{x}_1, ..., \mathbf{x}_n) \neq \nabla \tilde{V}(\mathbf{x}_i) \tag{5}$$

This is why straightforward quasi-Newton optimization of the single structures does not work well[13] without line search.[14] Quasi-Newton methods can be used efficiently by treating the entire system of n structures $\mathbf{X}^{\mathrm{T}} = (\mathbf{x}_1^{\mathrm{T}}, ..., \mathbf{x}_n^{\mathrm{T}})$ as one optimization problem: $\mathbf{G}^{\perp\mathrm{T}} = (\mathbf{g}_i^{\perp\mathrm{T}}, ..., \mathbf{g}_n^{\perp\mathrm{T}})$ is minimized as a function of $\mathbf{X}$. This demands a Hessian matrix n^2 times larger than Hessians in single structure optimizations, which is why the limited memory Broyden−Fletcher−Goldfarb−Shanno update (BFGS) is used.[15] Together with the fact that one Hessian is used "globally," for all structures, the optimizer was labeled GL-BFGS.[14] Other common techniques, like the Conjugate Gradient Method (CG),[16] can also be used "globally," minimizing $\mathbf{G}^{\perp}$ as a function of $\mathbf{X}$.

Another approach is to use a quadratic description of the PES, which means approximating $V(\mathbf{x}_i)$ by a quadratic function. This is not only physically meaningful but also provides a connection to the multitude of methods developed for single structure optimization. This idea has been pursued in different ways for SM[17,18] and NEB.[10] A local quadratic potential has been used by Schlegel and Ayala[19] to optimize reactants, transition states, and RP. The method does not require either springs or interpolation algorithms. The transition state is optimized using quasi-Newton/eigenvector following, and the RP is optimized using Gonzalez−Schlegel path following.[20,21] Separately considering $\mathbf{g}_i$ and its projection $\mathbf{g}_i^{\perp}$ makes it also possible to include the change of the tangents in the calculation.[10]

This work describes a simple, efficient, and robust algorithm to optimize a RP as in NEB and SM and to evolve it as in the GSM. It is based only on a local quadratic potential and on constraints concerning the spacing of the structures. It uses arbitrary tangents, for example, obtained from approximations common in NEB or SM calculations.[14] The two challenges stated in the beginning can be summarized as a constrained optimization problem: minimizing $\mathbf{g}^{\perp}$ under the constraint of a predefined distribution of the structures. Let d_i be the distances between two adjacent optimized structures, i and $i-1$:

$$d_i = |\mathbf{x}_i + \delta\mathbf{x}_i - \mathbf{x}_{i-1} - \delta\mathbf{x}_{i-1}| \tag{6}$$

Here, $\mathbf{x}_i$ is the current geometry, $\delta\mathbf{x}_i$ is the optimization step, and $\mathbf{x}_i + \delta\mathbf{x}_i$ are the coordinates of the optimized structure. Reactants and products will now be assumed to be fixed, which means that only structures $i = 2, ..., n-1$ are optimized, e.g., $\delta\mathbf{x}_1 = \delta\mathbf{x}_n = \mathbf{0}$. The structures are now required to be spaced such that the $n-1$ distances, d_i, have a predefined distribution, f_i, along the RP of length L:

$$d_i = f_i \cdot L$$
$$\text{with } i = 2, ..., n \text{ and } f_i > 0 \ \forall \ i \text{ and } \sum_{i=2}^{i=n} f_i = 1 \tag{7}$$

With a constant distribution, $f_i = 1/(n-1)$, all distances are equal and sum up to the path length as is usually the aim in NEB and SM. The deviation of the distances between the structures from the desired spacing can be expressed as an error vector, $\mathbf{R}$:

$$R_i = d_i - f_i \cdot L \tag{8}$$

Minimization of the norm of $\mathbf{R}$ is therefore the constraint of the optimization. With a Lagrangian multiplier λ, the optimization problem can be written as a minimization of the function

$$\mathcal{L} = \sum_{i=2}^{n-1} |(1 - \mathbf{t}_i\mathbf{t}_i^T)\mathbf{g}_i| + \lambda|\mathbf{R}| \tag{9}$$

with respect to the optimization steps $\delta\mathbf{x}_i$. Here $\mathbf{t}_i$, $\mathbf{g}_i$, and $\mathbf{R}$ depend on the steps, $\delta\mathbf{x}_i$. The function $\mathcal{L}$ is always positive, and $\mathcal{L}$ equals zero if the points lie on the RP.

The minimization of $\mathcal{L}$ will now be described under the approximation of a quadratic PES and fixed tangents, for example, those of the current approximation to the path $\mathbf{t}_i = \mathbf{t}_i(\mathbf{x}_1, ..., \mathbf{x}_n)$. The quadratic description of the PES, $V(\mathbf{x})$, of molecules using Hessian matrices is the workhorse of geometry optimization of minima and transition states:

$$V(\mathbf{x} + \delta\mathbf{x}) \approx V(\mathbf{x}) + \delta\mathbf{x}^T[\nabla V(\mathbf{x})] + \frac{1}{2}\delta\mathbf{x}^T[\nabla\nabla^T V(\mathbf{x})]\delta\mathbf{x} \tag{10}$$

$$= V(\mathbf{x}) + \delta\mathbf{x}^T\mathbf{g}(\mathbf{x}) + \frac{1}{2}\delta\mathbf{x}^T\mathbf{h}(\mathbf{x})\delta\mathbf{x} \tag{11}$$

This defines the gradient $\mathbf{g}(\mathbf{x})$ and the Hessian matrix $\mathbf{h}(\mathbf{x})$. The gradient change on the quadratic PES is given by

$$\mathbf{g}(\mathbf{x} + \delta\mathbf{x}) \approx \mathbf{g}(\mathbf{x}) + \mathbf{h}(\mathbf{x})\delta\mathbf{x} \tag{12}$$

The Newton step leading to a stationary point is obtained by requiring the gradient to be zero and solving for $\delta\mathbf{x}$:

$$0 = \mathbf{g}(\mathbf{x}) + \mathbf{h}(\mathbf{x})\delta\mathbf{x} \tag{13}$$

$$\delta\mathbf{x} = -\mathbf{h}^{-1}(\mathbf{x})\mathbf{g}(\mathbf{x}) \tag{14}$$

Dropping the dependence on $\mathbf{x}$ in the following, the optimization step to a point on the RP is obtained by inserting eq 12 into eq 1:

$$a_i\mathbf{t}_i = \mathbf{g}_i + \mathbf{h}_i\delta\mathbf{x}_i \tag{15}$$

$$\delta\mathbf{x}_i = -\mathbf{h}_i^{-1}\mathbf{g}_i + a_i\mathbf{h}_i^{-1}\mathbf{t}_i \tag{16}$$

The optimization step therefore precisely leads to coordinates with $|\mathbf{g}_i^{\perp}| = 0$. This resembles the Newton step that yields coordinates with $|\mathbf{g}_i| = 0$. The main difference is that the Newton step leads to a stationary point, and care must be taken if the number of negative eigenvalues of the Hessian is not the same as that of the desired stationary point. A point on the RP on the other hand is only characterized by its gradient being parallel to the tangent. The number of negative Hessian-eigenvalues is in general not known.

The first part of the problem, obtaining a step that gives $|\mathbf{g}^{\perp}| = 0$, is solved. Obviously, for a given tangent, $\mathbf{t}_i$, there are infinite solutions due to the arbitrary choice of a_i. The coefficients a_i now need to be determined to satisfy the constraint of predefined spacing.

The possible solutions are vectors as defined by eq 16. Computing an optimization step with specific spacing is therefore identical to placing points on these vectors so that they have a certain distance. As noted above, reactant and product structures are assumed to be fixed: $\delta\mathbf{x}_1 = \delta\mathbf{x}_n = \mathbf{0}$ and $a_1 = a_n = 0$. The distances between two adjacent optimized structures, i and $i - 1$, are readily evaluated as a function of a_i by inserting eq 16 into eq 6:

$$d_i = |\underbrace{\mathbf{x}_i - \mathbf{x}_{i-1} - \mathbf{h}_i^{-1}\mathbf{g}_i + \mathbf{h}_{i-1}^{-1}\mathbf{g}_{i-1}}_{\mathbf{Q}_i} + a_i \underbrace{\mathbf{h}_i^{-1}\mathbf{t}_i}_{\mathbf{T}_i} - a_{i-1} \underbrace{\mathbf{h}_{i-1}^{-1}\mathbf{t}_{i-1}}_{\mathbf{T}_{i-1}}| \tag{17}$$

$$d_i^2 = 2a_i\mathbf{Q}_i^T\mathbf{T}_i + a_i^2|\mathbf{T}_i|^2 + |\mathbf{Q}_i|^2 - 2a_ia_{i-1}\mathbf{T}_i^T\mathbf{T}_{i-1} - 2a_{i-1}\mathbf{Q}_i^T\mathbf{T}_{i-1} + a_{i-1}^2|\mathbf{T}_{i-1}|^2 \tag{18}$$

The path length, L, of the discretized RP is now defined as the sum of Cartesian distances between the optimized structures, d_i:

$$L(a_2, ..., a_{n-1}) = \sum_{i=2}^{i=n} d_i(a_i, a_{i-1}) \tag{19}$$

The error vector $\mathbf{R}$ can now be expressed and minimized as a function of the coefficients a_i. The dimension of the optimization problem is reduced to the number of structures to be optimized and is independent of the molecular degrees of freedom. First partial derivatives of the error vector, $J_{ij} = \partial R_i/\partial a_j$, are readily obtained from eq 18, which is at most quadratic in the coefficients. Approximating $\mathbf{R}$ to first order in a_i gives

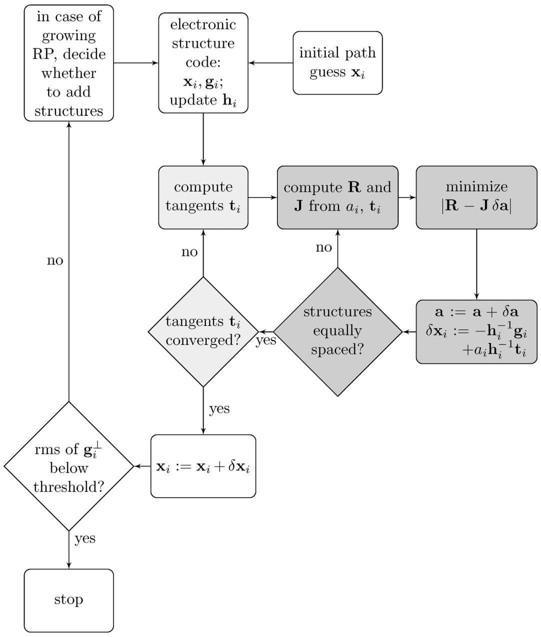
$$R_i(a_2, ..., a_{n-1}) \approx R_i(\tilde{a}_2, ..., \tilde{a}_{n-1}) + \sum_{j=2}^{n-1} J_{ij}(\tilde{a}_2, ..., \tilde{a}_{n-1})\cdot(a_j - \tilde{a}_j) \tag{20}$$

Refined coefficients $\mathbf{a} = \tilde{\mathbf{a}} + \delta\mathbf{a}$ can be obtained from a current guess, $\tilde{\mathbf{a}}$, the corresponding error vector, $\mathbf{R} = \mathbf{R}(\tilde{a}_2, ..., \tilde{a}_{n-1})$, and its first derivatives, $\mathbf{J} = \mathbf{J}(\tilde{a}_2, ..., \tilde{a}_{n-1})$, by minimizing

$$|\mathbf{R} - \mathbf{J}\delta\mathbf{a}| \tag{21}$$

using singular value decomposition (SVD).[1] Note that $\mathbf{R}$ is of dimension $n - 1$, $\mathbf{a}$ is of dimension $n - 2$, and $\mathbf{J}$ is therefore rectangular.[23]. The vector $\mathbf{R}$ and its derivatives $\mathbf{J}$ are now recalculated for $\mathbf{a} = \tilde{\mathbf{a}} + \delta\mathbf{a}$. This procedure is iterated until the path length is converged and the structures are equally spaced. It typically converges in less than 10 iterations. The threshold for the deviation of d_i from the desired value is 10^{-6} a_0 and for the change of the path length, L, 10^{-10} a_0. An overview over the complete course of the optimization of a RP is given in Scheme 1.

Scheme 1. A Simplified Scheme Illustrating the Course of a Self-Consistent RP Optimization[a]


in case of growing RP, decide whether to add structures
electronic structure code: $\mathbf{x}_i, \mathbf{g}_i$; update $\mathbf{h}_i$
initial path guess $\mathbf{x}_i$
compute tangents $\mathbf{t}_i$
compute $\mathbf{R}$ and $\mathbf{J}$ from a_i, $\mathbf{t}_i$
minimize $|\mathbf{R} - \mathbf{J}\,\delta\mathbf{a}|$
no
no
no
tangents $\mathbf{t}_i$ converged?
yes
structures equally spaced?
$\mathbf{a} := \mathbf{a} + \delta\mathbf{a}$ $\delta\mathbf{x}_i := -\mathbf{h}_i^{-1}\mathbf{g}_i + a_i\mathbf{h}_i^{-1}\mathbf{t}_i$
yes
rms of $\mathbf{g}_i^{\perp}$ below threshold?
$\mathbf{x}_i := \mathbf{x}_i + \delta\mathbf{x}_i$
yes
stop

[a] The white part is the framework of the driving program and is the same as for NEB, SM, GSM, or any geometry optimization. The gray parts constitute the optimization algorithm. The basic algorithm computing the optimization step, for given tangents, $\mathbf{t}_i$, is shown in dark gray. The optimization step, $\delta\mathbf{x}_i$, is defined by the coefficients a_i. Optimization of the coefficients, a_i, is finished when they are converged so that the resulting optimized coordinates are equally spaced (threshold 10^{-6} a_0) and the pathlength is converged (threshold 10^{-10} a_0). The tangents can be updated (pale gray boxes) until the optimization step is converged (threshold 10^{-6} a_0) also with respect to tangent change. Initialization of $\delta\mathbf{x}_i$ and a_i is done as in eq 23 and has been omitted for clarity.

This algorithm is computationally not demanding, because the scalar products in eq 18 need only be computed once. The other matrix operations including SVD are fast because the dimension is given by the number of structures (typically less than 30). Since the algorithm described above can only optimize existing a_i, one needs to provide an initial guess. A reasonable choice for this guess is the optimization step that is orthogonal to the current tangent $\mathbf{t}_i$:

$$0 = \mathbf{t}_i^T[-\mathbf{h}_i^{-1}\mathbf{g}_i + \tilde{a}_i\mathbf{h}_i^{-1}\mathbf{t}_i] \tag{22}$$

$$\tilde{a}_i = \frac{\mathbf{t}_i^T\mathbf{h}_i^{-1}\mathbf{g}_i}{\mathbf{t}_i^T\mathbf{h}_i^{-1}\mathbf{t}_i} \tag{23}$$

If the recipe described above is followed, one obtains optimized coordinates $\mathbf{x}_i + \delta\mathbf{x}_i$ that (1) have the predefined, for example equally spaced, distribution, f_i, in Cartesian space and (2) lie on the RP according to the quadratic PES and the old tangents $\mathbf{t}_i = \mathbf{t}_i(\mathbf{x}_1,..., \mathbf{x}_n)$. An optimization step that also takes into account the tangent change can in most cases be computed by simply updating the tangents for the step obtained with old tangents, $\mathbf{t}_i$

= $\mathbf{t}_i(\mathbf{x}_1 + \delta\mathbf{x}_1, ..., \mathbf{x}_n + \delta\mathbf{x}_n)$, and recalculating the coefficients a_i until self-consistency. In some of the examples studied below, this straightforward approach did not work, and a DIIS-scheme[24] has been used to converge the tangents. As updating the tangents turned out to give no significant improvement (see below), this DIIS scheme is described in the Supporting Information. In practice, the tangents are updated until the root mean square (RMS)[25] of the resulting change of $\delta\mathbf{x}_i$ is less than 10^{-6} a_0. In ref 19, Ayala and Schlegel use tangents as defined in Gonzalez−Schlegel path-following,[20,21] which means they depend on the optimization step. A step based on these tangents is computed, and the structures are afterward redistributed to be equally spaced on the RP. This procedure is iterated until convergence. The algorithm described here uses a complementary approach: An equally spaced optimization step is first calculated for fixed tangents. Self-consistence can be achieved by repeatedly recomputing the tangents according to the new coordinates.

The Hessian matrices are as usual in geometry optimization most efficiently obtained as approximate Hessians. They are initialized as unity matrices or as simple model Hessians[26,27] and updated. Step-size control is applied using the trust region method: All Hessian matrices are diagonal-shifted until the steps of the initial guesses (see eq 23) are inside the trust radii. The actual steps are then usually also inside the trust radii. The details of the Hessian update and trust region method are of course not unique, and details of the procedure used here can be found in the Supporting Information.

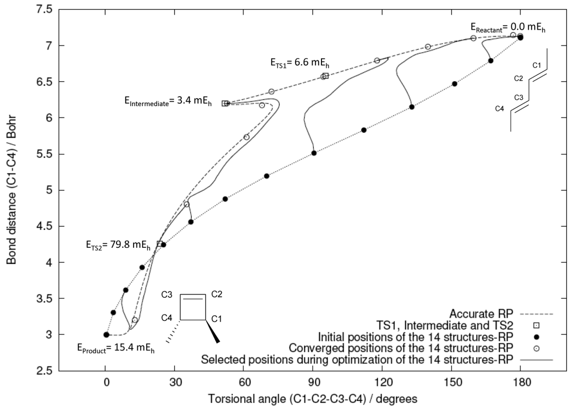
The GSM introduced the concept of evolving the RP from both ends. Structures are added sequentially if the growing parts of the RP are sufficiently converged. When the two fragments of the RP meet, the merged RP can be optimized as usual in string methods. An initial guess is therefore not needed, which is beneficial for complicated reactions, where such a guess is hard to obtain. A RP can be evolved within the discussed method by treating the two growing parts as one RP with increased distance between the two frontier structures. This means using a distribution of the structures with two different values. If n is the final number of structures of the full grown RP, n_1 is the current number and j and $j - 1$ are the frontier structures, then the distribution is

$$f_i = \begin{cases} \dfrac{1}{n - 1} & \text{for } i \neq j \\ \dfrac{n + 1 - n_1}{n - 1} & \text{for } i = j \end{cases} \tag{24}$$

The tangents of the loose ends of the growing RP are described by vectors pointing to the adjacent inner structure.

The cyclization of *trans*-hexadiene (see Scheme 2) was introduced as a test reaction for the freezing string method.[5] The reaction has a high barrier for cyclization (79.8 mE_h) and a much lower rotational barrier (6.6 mE_h). If reactant and product structure are oriented so that carbon atoms C1 to C4 lie in one plane (see Scheme 2), it is indeed a challenging reaction in Cartesian coordinates. In a linear interpolation, these carbon atoms stay in that plane so that several atoms crash or are pushed through bonds. In the actual RP, two of the carbon atoms in the plane have to be bent out of the plane and rotated back in afterward. The RP was started with two structures per fraction of the RP and has been grown to a total number of 20 structures. New structures were added when the RMS of the projected gradient, $\mathbf{g}_i^{\perp}$, of the frontier structures dropped below 0.01 E_h/a_0. Using these convergence thresholds, the RP evolves nonlinearly and includes only reasonable structures. The RP is closed after 64 iterations (65 iterations for fixed tangents) if the Hessians are initialized as unity matrices. If the Hessian matrices are initialized as model Hessians, the RP is closed after 57 iterations (61 for fixed tangents). All electronic structure calculations described in this letter were carried out with TURBOMOLE at the BP86/def2-SV(P) level of density functional theory.[29−37]

Scheme 2. Illustration of the RP of the Cyclization of Hexadiene in Terms of the Bond Distance Corresponding to Bond Formation and the Torsional Angle Corresponding to the Formed Ring[a]



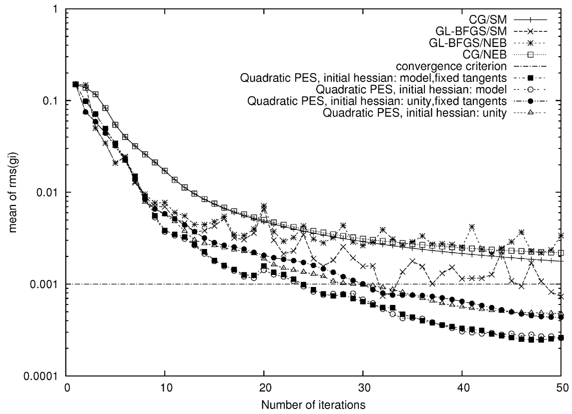
[a]The structures of intermediate and transition states have been obtained by usual quasi-Newton methods. The accurate RP has been obtain by two separate RP optimizations.[38] The initial and the converged path are shown for a calculation with 14 structures. For clarity, the path taken during the optimization from initial to final position is shown only for a few structures.

The challenge of this reaction is of course artificial and due to the choice of coordinate system and starting geometry. Using internal coordinates or a different orientation (see below), the problem could be solved more easily. The example demonstrates, however, that even when an optimization cannot be started from linear interpolation, a nonlinear RP can be evolved in the same coordinate system. Evolution of the RP as an alternative to linear interpolation is the basic idea of the GSM.

If the Cartesian distance between reactant and product structure of the discussed reaction is minimized by rotation and translation, linear interpolation gives a very good initial guess. This guess is used to test the efficiency of the method by comparing it against the string method (SM)[39] and Nudged Elastic Band method (NEB) as implemented in the Dl-FIND optimizer.[40] SM and NEB were used together with the conjugate gradients (CG)[14] and GL-BFGS Optimizer[14] mentioned in the beginning. All methods were used with the tangents as described in ref 6 and employed 14 structures to discretize the RP.

In this example, the methods from the literature are more sensitive to optimizer (CG/GL-BFGS) than to method (SM/NEB), as noted in ref 14. The GL-BFGS optimizer converges faster but is less systematic than CG. From Scheme 3, it can be seen that the proposed method performs slightly better and is more robust than the best literature optimizer "GL-BFGS" that uses curvature information. The tangent update does not improve the performance. The model Hessian as a better initial

Scheme 3. Comparison of the Performance of Different Methods in the RP Optimization of the Cyclization of Hexadiene[a]



[a]The graph shows the mean value of RMS($\mathbf{g}_i^{\perp}$) in E_h/a_0 over all i = 1, ..., n structures as a function of the number of iterations. In the case of NEB calculations, the RMS of the NEB force has been added to the projected gradient.[25]

Table 1. Short Description of Test Reactions and Number of Iterations to Reach the Threshold, Mean of RMS($\mathbf{g}_i^{\perp}$) < 10^{-3} E_h/a_0[a]

method used	string		NEB		quadratic PES	
details (optimizer/initial Hessian)	CG	GL-BFGS	CG	GL-BFGS	model	unity
cyclization of hexatriene[28]	-	32	-	-	24	31
Diels−Alder reaction	59	24	-	33	16	23
alanine−dipeptide rearrang.	29	13	32	17	13	14
H_2O + *iso*-butene → *tert*-butanol[9]	52	23	-	-	16	22
acrylate formation, Ni-cat.	57	30	-	-	21	26
CO_2−ethylene coupling, Ni-cat.	41	19	48	24	13	16
amine + alcohol → hemiaminal	25	12	28	12	10	12
hemiaminal → imine, catalyzed	60	25	-	45	19	21
hemiaminal → imine	-	26	-	37	17	23
polymerization of ethylene, Ni-cat.	51	17	-	25	17	20
alkylation of formaldehyde by MeI	29	21	33	41	11	12

[a]The described method has been run with a model Hessian and unity matrix as initial Hessians and with fixed tangents. In the case of NEB calculations, the RMS of the NEB force has been added to the projected gradient.[25] Optimizations that did not reach the threshold within 60 iterations were assigned a "-". The number of energy/gradient calls is given by 2 + 12 × [number of iterations].

Hessian clearly improves the performance of the method. While Scheme 3 shows how the projected gradient converges, the geometrical difference to exact transition states and RP is illustrated in Scheme 2 following the presentation in ref 28. The discretized RP converges fast initially in terms of the projected gradient and then slows down. This can be explained in terms of the PES: The high-energy TS is close to the product, while the potential is very flat for the longest part of the RP, and there even exists a shallow minimum that is off a direct path connecting product and reactant. A good geometrical and energetic estimate of the high-energy part of the RP is obtained fast. Convergence in terms of the structure in the low-energy part is much slower. The discretized RP converges to the intermediate only if it is optimized to a very low projected gradient (mean of RMS($\mathbf{g}_i^{\perp}$) = 5.2 × 10^{-6} E_h/a_0).

More extended tests have been carried out for the reactions listed in Table 1. As above, initial paths have been obtained by interpolation in Cartesian coordinates, and 14 structures have been used to discretize the RP. The trends observed above are, in general, confirmed: For the methods from the literature, the choice of optimizer is most important, and GL-BFGS performs much better than CG. The SM perfoms consistently somewhat better than NEB, so that the best method/optimizer combination from the literature is SM/GL-BFGS. As mentioned previously, GL-BFGS uses curvature information based on the projected gradients and is initialized with a unity matrix Hessian. This explains the observed performance: Depending on Hessian initialization, the method described here improves on average by two (unity matrix) or six iterations (model Hessian) on SM/GL-BFGS. Using a structure of the converged RP as a TS guess, it was for all reactions listed in Table 1 possible to obtain the exact TS by refinement with the trust region image method[1] with analytical Hessian calculation as implemented in TURBOMOLE. The barrier was estimated with an accuracy of better than 5 mE_h for seven out of the 11 test reactions.

A method has been described to self-consistently compute reaction paths based on a quadratic PES as in single structure quasi-Newton methods. An optimization step to the reaction path within the quadratic approximation is well-defined and can be restricted to have predefined spacing between the structures. It is not necessary to invoke heuristics such as spring forces or interpolation algorithms: The only assumption is a quadratic PES. This is an improvement over existing methods because it reduces the dependency on additional approximations. In practice, however, an additional interpolation algorithm is still useful to obtain a better initial guess or to improve the addition of structures to the growing RP. The method has been shown to perform better than the tested methods from the literature. Complicated reactions can be provided with an initial guess as in the growing string method.

■ ASSOCIATED CONTENT

Ⓢ Supporting Information

Details of the derivatives of the error vector with respect to a_i, the Hessian update, step-size control, and Cartesian input coordinates of the considered reactions are provided in the Supporting Information. This material is available free of charge via the Internet at http://pubs.acs.org.

■ AUTHOR INFORMATION

Corresponding Author

*E-mail: philipp-nikolaus.plessow@basf.com.

Notes

The authors declare no competing financial interest.

■ ACKNOWLEDGMENTS

P.P. works at CaRLa of Heidelberg University, being cofinanced by Heidelberg University, the state of Baden-Wuerttemberg, and BASF SE. Support from these institutions is gratefully acknowledged. P.P. thanks Dr. Robert Send and Dr. Ansgar Schaefer for helpful comments on the manuscript.

REFERENCES

(1) Helgaker, T. *Chem. Phys. Lett.* **1991**, *182*, 503−510.

(2) Halgren, T.; Lipscomb, W. *Chem. Phys. Lett.* **1977**, *49*, 225−232.

(3) Elber, R.; Karplus, M. *Chem. Phys. Lett.* **1987**, *139*, 375−380.

(4) G, M.; Jonsson, H. *Phys. Rev. Lett.* **1994**, *72*, 1124−1127.

(5) Fukui, K. *J. Phys. Chem.* **1970**, *74*, 4161.

(6) Henkelman, G.; Jonsson, H. *J. Chem. Phys.* **2000**, *113*, 9978−9985.

(7) Weinan, E.; Ren, W.; Vanden-Eijnden, E. *Phys. Rev. B* **2002**, *66*, 052301.

(8) Behn, A.; Zimmerman, P. M.; Bell, A. T.; Head-Gordon, M. *J. Chem. Theory Comput.* **2011**, *7*, 4019−4025.

(9) Chaffey-Millar, H.; Nikodem, A.; Matveev, A. V.; Krueger, S.; Roesch, N. *J. Chem. Theory Comput.* **2012**, *8*, 777−786.

(10) Chu, J.; Trout, B.; Brooks, B. *J. Chem. Phys.* **2003**, *119*, 12708−12717.

(11) Burger, S. K.; Yang, W. *J. Chem. Phys.* **2007**, 127.

(12) This formally includes energy dependence, as energy is a function of structure. In the case of energy-dependent tangents, evaluation of the tangents would always require energy calculations. For most tangents, as defined in ref 6, only the relative order of the magnitude of the energies is important. This as well as the small influence of tangent updates justifies the assumption of constant energies within one optimization cycle.

(13) Trygubenko, S.; Wales, D. *J. Chem. Phys.* **2004**, *120*, 2082−2094.

(14) Sheppard, D.; Terrell, R.; Henkelman, G. *J. Chem. Phys.* **2008**, *128*, 134106.

(15) Nocedal, J. *Math. Comput.* **1980**, *35*, 773−782.

(16) Polak, E.; Ribiere, G. *Revue Francaise d'Informatique De Recherche Operationnelle*; University of Michigan: Ann Arbor, MI, 1969; Vol. 3.

(17) Peters, B.; Heyden, A.; Bell, A.; Chakraborty, A. J. *Chem. Phys.* **2004**, *120*, 7877−7886.

(18) Burger, S.; Yang, W. *J. Chem. Phys.* **2006**, *124*, 054109.

(19) Ayala, P.; Schlegel, H. *J. Chem. Phys.* **1997**, *107*, 375−384.

(20) Gonzalez, C.; Schlegel, H. *J. Chem. Phys.* **1989**, *90*, 2154−2161.

(21) Gonzalez, C.; Schlegel, H. *J. Phys. Chem.* **1990**, *94*, 5523−5527.

(22) Anderson, E.; Bai, Z.; Bischof, C.; Blackford, S.; Demmel, J.; Dongarra, J.; Du Croz, J.; Greenbaum, A.; Hammarling, S.; McKenney, A.; Sorensen, D. *LAPACK Users' Guide*, 3rd ed.; Society for Industrial and Applied Mathematics: Philadelphia, PA, 1999.

(23) The overdetermined formulation of the equation system can be avoided. If the path length is in all components $R_i = d_i - f_i L$ substituted by $L = d_1/f_1$, the first component, R_1, is eliminated, and $\mathbf{R}$ has the dimension $n - 2$.

(24) Pulay, P. *Chem. Phys. Lett.* **1980**, *73*, 393−3982.

(25) The RMS of a vector $\mathbf{v}$ with n_v entries is $\mathrm{RMS}(\mathbf{v}) = (\mathbf{v}^{\mathrm{T}}\mathbf{v}/n_v)^{1/2}$. The projected gradient of a RP is analyzed as the mean value over all $i = 1, ..., n$ structures of $\mathrm{RMS}(\mathbf{g}_i^{\perp})$. In the case of NEB calculations, the RMS of the NEB force has been added, because only its vanishing assures equal spacing and therefore comparability to the string method and the discussed method, which always enforce equally spaced structures.

(26) Lindh, R.; Bernhardsson, A.; Karlstrom, G.; Malmqvist, P. *Chem. Phys. Lett.* **1995**, *241*, 423−428.

(27) The Hessian was intialized in the first iteration, reinitialized, and updated in the second iteration and in and after the third iteration only updated. The model Hessian has been obtained by numerical differentiation of the harmonic model function proposed by Lindh et al. The parameters of the third row given in ref 26 have simply been extended to the lower rows of the periodic table.

(28) Behn, A.; Zimmerman, P. M.; Bell, A. T.; Head-Gordon, M. *J. Chem. Phys.* **2011**, *135*, 224108.

(29) *TURBOMOLE*, V6.3; University of Karlsruhe and Forschungszentrum Karlsruhe GmbH: Karlsruhe, Germany, 1989−2007; TURBOMOLE GmbH: Karlsruhe, Germany, 2007. Available from http://www.turbomole.com (accessed 1/21/2013).

(30) Weigend, F.; Ahlrichs, R. *Phys. Chem. Chem. Phys.* **2005**, *7*, 3297.

(31) Weigend, F. *Phys. Chem. Chem. Phys.* **2006**, *8*, 2057.

(32) Becke, A. *Phys. Rev. A* **1988**, *38*, 3098.

(33) Dirac, P. A. M. *Proc. R. Soc. London, Ser. A* **1929**, *123*, 714.

(34) Slater, J. C. *Phys. Rev.* **1951**, *81*, 385.

(35) Vosko, S. H.; Wilk, L.; Nusair, M. *Can. J. Phys.* **1980**, *58*, 1200.

(36) Perdew, J. P.; Wang, Y. *Phys. Rev. B* **1992**, *45*, 13244.

(37) Perdew, J. P. *Phys. Rev. B* **1986**, *33*, 8822.

(38) The two optimizations have been started from the three minimum structures with 100 structures each and have been converged to a mean of $\mathrm{RMS}(\mathbf{g}_i^{\perp}) \approx 10^{-4}$ $\mathrm{E_h/a_0}$. The estimated energy barriers from these optimizations differ by $\approx 10^{-4}$ $\mathrm{E_h}$ from the precise values of the TS optimization.

(39) The string method as implemented in DL-FIND has not been published and needs to be activated in the open-source code in which it is deactivated by default.

(40) Kaestner, J.; Carr, J. M.; Keal, T. W.; Thiel, W.; Wander, A.; Sherwood, P. *J. Chem. Phys.* **2009**, *43*, 11856−11865.