This document is the Accepted Manuscript version of a Published Work that appeared in final form in:
Cvitaš, M. T. (2018). Quadratic string method for locating instantons in tunneling splitting calculations. Journal of Chemical Theory and Computation, 14(3), 1487–1500. https://doi.org/10.1021/acs.jctc.7b00881

Quadratic string method for locating instantons in tunneling splitting calculations

Marko T. Cvitaš*

Ruđer Bošković Institute, Department of Physical Chemistry, Bijenička Cesta 54, 10000 Zagreb, Croatia.

E-mail: mcvitas@irb.hr

December 31, 2017

Abstract

The ring-polymer instanton (RPI) method is an efficient technique for calculating approximate tunneling splittings in high-dimensional molecular systems. In the RPI method, tunneling splitting is evaluated from the properties of the minimum action path (MAP) connecting the symmetric wells, whereby the extensive sampling of the full potential energy surface of the exact quantum-dynamics methods is avoided. Nevertheless, the search for the MAP is usually the most time-consuming step in the standard numerical procedures. Recently, nudged elastic band (NEB) and string methods, originaly developed for locating minimum energy paths (MEPs), were adapted for the purpose of MAP finding with great efficiency gains [*J. Chem. Theory Comput.* **2016**, 141, 024101]. In this work, we develop a new *quadratic* string method for locating instantons. The Euclidean action is minimized by propagating the initial guess (a path connecting two wells) over the quadratic potential energy surface approximated by means of updated hessians. This allows the algorithm to take many minimization steps between the potential/gradient calls with further reductions in the computational effort, exploiting the

*To whom correspondence should be addressed

smoothness of potential energy surface. The approach is general, as it uses Cartesian coordinates, and widely applicable, with computational effort of finding the instanton usually lower than that of determining the MEP. It can be combined with expensive potential energy surfaces or on-the-fly electronic-structure methods to explore a wide variety of molecular systems.

1 INTRODUCTION

Tunneling splitting in symmetric well systems manifests itself in the energy difference of the otherwise degenerate ground state level.[1] The magnitude of the splittings in molecular systems varies over many orders of magnitude and it is sensitive to the non-equilibrium regions of potential energy surface (PES).[2] Numerical determination of splittings can thus be used to learn about interactions and dynamics, and to interpret experimental spectra, with the study of rearrangement dynamics in water clusters, as a recent notable example.[3–5]

Tunneling spittings can be obtained numerically by a direct solution of the Schrödinger equation. However, this is a formidable computational task even for small gas-phase systems[6–8] and scales exponentially with the number of degrees of freedom (DOF). For large systems, either the diffusion Monte Carlo can be used,[9–11] which is possible if one can locate the nodal diving surface of the first excited state (by symmetry arguments or otherwise), or the recent method based on the path integral techniques.[12] Those methods scale roughly linearly with the system size and give the exact result. All exact numerical approaches for calculating tunneling splittings probe the PES at a vast number of molecular geometries that are accessible to the nuclear wave function and are therefore computationally expensive. A number of approximate semiclassical methods have therefore developed,[13–18] which often give qualitative agreement with the experimental splittings. The instanton method belongs to the class of approximate semiclassical methods. The tunneling splitting is obtained from a single imaginary-time trajectory, which carries the dominant contribution in the path integral representation, and its harmonic vicinity. The accuracy of the tunneling splittings

increases as the symmetric wells get better separated and as the barrier between them rises. In this limit, when the tunneling splittings are extremely small, instanton method can even rival the accuracy of the exact quantum methods based on basis set representation, as it becomes increasingly difficult to resolve the levels with a reasonable basis set size.

The instanton method has first been introduced in the rate theory[19,20] and subsequently extended to treat tunneling splittings by Vainshtein *et al.*[21] Mil'nikov and Nakamura extended the theory to treat multidimensional systems and developed a numerical approach for calculating tunneling splittings,[22] which they applied to a number of molecules.[23] They used the internal molecular coordinates in their approach, which means that it was not trivial to adapt it to treat new molecular geometries. The method has been rederived as the ring-polymer instanton (RPI)[24,25] method, where discrete mathematics and Cartesian coordinates were employed to simplify algebra. This formulation led to a general computational tool that is readily applicable to systems of arbitrary geometries.[25] The approach was also extended to treat any number of symmetric wells[26] and low vibrationally excited states.[27]

Instantons have recently been a topic of active investigation. In rate theory context, a derivation from first principles was proposed[28] (that avoids the Im F premise), as well as the extensions to treat certain non-adiabatic systems,[29–31] approaches to calculate microcanonical rates[32,33] and the rates above the cross-over temperature.[34–36]

At low temperatures, investigative work concentrated on improving efficiency. Rommel *et al*[38] used unequal imaginary time steps in order to distribute ring polymer beads along the path more uniformly and prevent the clustering of beads near turning points. Einarsdóttir *et al*[39] tackled the same problem using nudged elastic band method (NEB)[40] to locate the instanton path by minimizing the abbreviated action. Kryvohuz developed an instanton search algorithm based on the Fourier representation of the path,[37] which is independent of the path length. Kawatsu and Miura[41] found a way to take the zero-temperature limit analytically, which is a powerful tool in high accuracy calculations. Smedarchina *et al* developed a modified instanton approach[18,42,43] based on the potential and hessian at the stationary

points only, without the explicit evaluation of the instanton trajectory. In a recent paper,[44] the chain-of-states-based minimum-energy path (MEP) finding methods, such as the nudged elastic band (NEB)[40] and string methods,[45,46] were used to minimize the abbreviated action at zero temperature in order to calculate tunneling splittings, which resulted in orders of magnitude efficiency savings, measured in terms of the number of potential evaluations needed.

In the present paper, we develop a way to apply another MEP-finding tool, the *quadratic* string method (QSM),[47] to further reduce the computational work in finding instantons in calculations of tunneling splittings in molecules. We measure the required work for locating instantons in terms of the number of calls to the potential/gradient subroutine, *i.e.*, the number of geometries at which potential and gradient are evaluated. The assumption is that the time overhead of the specific numerical method employed is negligible compared to the *ab initio* potential/gradient evaluations, with the future on-the-fly applications in view. The minimization methods that rely on the exact hessian evaluation at every iteration have not been considered.

In the quadratic string method,[47] the abbreviated action is minimized by integration in the direction perpendicular to the path. The minimum action path (MAP) is represented by a string of beads and the trajectory of every bead is integrated on an approximate quadratic surface obtained by a local Taylor expansion that uses updated hessians. Updated hessians are approximate hessians calculated using the standard formulae[48] that are based on the gradient information from previous bead positions. The integration range and the quality of the approximate PES are controlled using a trust region algorithm. In the MEP finding applications, the method proved to be more efficient than the standard NEB with the global LBFGS minimizer for certain benchmark systems.[47] The number of steps taken, *i.e.*, the number of local integrations performed, during a minimization depends on how 'quadratic' the surface is and on how much the properties of the PES change as beads travel over the potential energy landscape in course of a minimization. The abbreviated action surface

parametrized by Cartesian coordinates of bead positions is a much more complicated and rugged surface than the PES. The action gradients are composed of two forces: one tries to minimize the potential, whereas the other wants to minimize the path length and couples the beads. So the idea behind the QSM is to gain the computational savings by using a search that is guided by the simplicity of the PES instead of the perpendicular-to-the-path-projected gradients of the abbreviated action surface, which is contained in the integration scheme on the approximated potential.

After the MAP has been determined using the QSM in the first step, the numerical evaluation of tunneling splittings proceeds by (ii) evaluating hessians along the MAP connecting the minima, and, (iii) evaluating the contribution of neighboring paths in harmonic approximation using hessians from step (ii). In the RPI method, the harmonic contributions in step (iii) are calculated using the eigenvalues of the discretized action hessian (see Section 2), which, at low temperatures, involves a diagonalization of a large banded matrix. Instead, we use the zero-temperature limit formula by Mil'nikov and Nakamura,[22] which is a multi-dimensional extension of the original expression by Vainsthein *et al*,[21] and evaluate it, for the first time, in Cartesian coordinates (Section 5). In this way, all numerical effort has been transfered into the MAP search using the QSM and the subsequent hessian evaluation along the MAP.

The RPI method for calculating tunneling splittings is summarized in Sec. 2, together with the standard and the chain-of-states-based numerical methods for evaluating them. It is shown that the chain-of-states methods based on minimizing the abbreviated action are equivalent to those that are based on the full action minimization with optimally adjusting time steps[44] (at every iteration in course of a minimization). In Sec. 3, we develop the quadratic string method for instanton search. Sec. 4 summarizes the evaluation of harmonic contributions using the Mil'nikov-Nakamura formula in Cartesian coordinates. In Sec. 5, the quadratic string method is applied to calculate the tunneling splittings in malonaldehyde, ammonia and hydronium and the efficacy of the method is compared to the string method

(with LBFGS optimizer) for finding instantons.[44] Concluding remarks are given in Sec. 6. Atomic units are used throughout, with $\hbar = 1$.

2 RING POLYMER INSTANTON METHOD

In the formulation of statistical Feynman path integrals, the instanton represents the path with the dominant contribution to the quantum partition function. Tunneling splittings can be obtained from the ratio of quantum partition functions, whereby the contribution from the dominant path is calculated and that of the harmonic fluctuations about it is evaluated and added analytically using potential hessians along the path. The above approximation thus avoids the need to sample the PES extensively.

The tunneling splitting, Δ, between two symmetric wells can be obtained from the ratio of partition functions,[1,21]

$$\lim_{\beta\to\infty} \frac{Q(\beta)}{Q_0(\beta)} = \cosh\left(\frac{\beta\Delta}{2}\right), \tag{1}$$

where $Q(\beta)$ is the partition function of the full system and $Q_0(\beta)$ is the partition function of the system in the absence of tunneling. The expression in Eq. (1) is evaluated in the zero-temperature limit, with $\beta = 1/kT$, where k is the Boltzmann constant. Quantum partition functions can be expressed in the discretized path-integral (ring polymer) formalism by splitting paths into N infinitesimal, and not necessarily equal,[38] imaginary time steps d_i, where $i = 1, ..., N$ and $\sum_{i=1}^{N} d_i = \beta$,

$$Q(\beta) = \mathrm{Tr}\left[\mathrm{e}^{-\beta\hat{H}}\right] = \lim_{N\to\infty} \prod_i \frac{1}{\sqrt{2\pi d_i}} \int \mathrm{d}\mathbf{x} \mathrm{e}^{-S(\mathbf{x})}, \tag{2}$$

where S is recognized as the Euclidean action discretized on the imaginary time grid d_i as

$$S(\mathbf{x}) = \sum_{i=1}^{N} \left[\frac{1}{2} \sum_{j=1}^{f} \frac{(x_{i+1,j} - x_{i,j})^2}{d_i} + V(x_{i,1}, ..., x_{i,f}) \frac{d_i + d_{i+1}}{2} \right]. \tag{3}$$

The molecular system follows the path discretized using mass-scaled Cartesian coordinates (the ring-polymer coordinates) $x_{i,j}$ ($x_{i,j} = \sqrt{m_j} \times$ position of atom j), where i denotes the imaginary time index or the bead number, and j denotes the degree of freedom, and $f = 3 \times$ (number of atoms) is the dimensionality of the system.

The dominant contribution to the partition function comes from path that minimizes the action S in the exponent of Eq. (2) and its vicinity. According to the least action principle, the minimum of Euclidean action corresponds to the classical trajectory on the inverted potential energy surface.[1] In the RPI method,[25] the ratio in Eq. (1) is evaluated approximately by expanding the Euclidean action S to second order in the time- and mass-scaled Cartesian coordinates $x_{i,j}\sqrt{(d_{i+1}+d_i)/2}$[38] about the classical periodic orbit connecting the two symmetric minima, the instanton. The instanton orbit is of infinite period, $\beta \to \infty$, *i.e.*, zero energy (where zero is defined at the PES minimum), and the contributions of arbitrary number of passages between the minima are included.[21] The final result for the tunneling splitting Δ is expressed in terms of the single kink action $S_{\rm kink}$, which is the action of a single passage from one minimum to another, and the prefactor Φ as[25]

$$\Delta = \frac{2}{\Phi}\sqrt{\frac{S_{\rm kink}}{2\pi}}\mathrm{e}^{-S_{\rm kink}}. \tag{4}$$

The prefactor Φ is the ratio of the non-zero eigenvalues λ_l of the hessian of the action $S_{\rm kink}$,[38]

$$\mathbf{H}_{i,j;i',j'} = \frac{\partial^2 S_{\rm kink}}{\partial x_{i,j}\partial x_{i',j'}}\frac{2}{\sqrt{d_{i+1}+d_i}\sqrt{d_{i'+1}+d_{i'}}}, \tag{5}$$

in time- and mass-scaled coordinates, and the action hessian eigenvalues λ_l^0 of the corresponding non-tunneling system as

$$\Phi = \left[\frac{\prod_l \lambda_l}{\prod_{l'} \lambda_{l'}^0}\right]^{1/2}. \tag{6}$$

The matrix elements of the action hessian in Eq. (5) are given in Appendix I of Ref. 44. The

excluded zero eigenvalues of the action hessian come from the invariance under imaginary-time translation of the kink[21,25] and, if the system considered is free in space, from overall ring polymer (path) translation and rotation. The RPI theory neglects the overall rotational motion of the molecule and the anharmonicity perpendicular to the instanton tunneling path.

Numerical implementation of the RPI method consists of finding the MAP by minimizing the kink action in Eq. (3), and, subsequently, the evaluation of hessians along the MAP and the calculation of the prefactor Φ.

The prefactor Φ can be calculated in several ways. In a direct approach, one can determine the eigenvalues of the S-hessian by explicit diagonalization of the banded matrix.[25] In the approach adopted here and developed by Mil'nikov and Nakamura,[22] a semiclassical formula, obtained in the $\beta \to \infty$ limit and based on the integration of Jacobi fields, is used. Other approaches exist, but have so far only been applied to the rate calculations.[19,30]

3 LOCATING INSTANTONS

Before proceeding to develop the QSM, we overview below the current methodology of locating insantons numerically. We describe the standard RPI approach[25] and the recent chain-of-states methods,[44] based on either the string or NEB representation, which offer superior performance. We also show that the two chain-of-state methods, the Euclidean action minimization using variable time steps and the string method, both developed in Ref. 44, are equivalent.

In the standard RPI approach for calculating tunneling splittings in molecules, the instanton orbit is obtained numerically by Euclidean action minimization in Eq. (3). The single kink path is represented by N beads along the path in Cartesian coordinates $x_{i,j}$, which serve as minimization parameters. Each bead is a molecular state of dimensionality f, a snapshot of the imaginary-time trajectory that connects the two symmetric minima. We need to specify the starting guess (initial values for $x_{i,j}$), set the time duration of the kink, β,

and the imaginary-time step sizes d_i in Eq. (3). The Nf-dimensional minimization of action is then performed, most commonly using the limited-memory Broyden-Fletcher-Goldfarb-Shano (LBFGS) algorithm,[49] to iteratively provide improved estimates of the instanton orbit. The minimization algorithm is guided by the action gradients,

$$\frac{\partial S}{\partial x_{i,j}} = \frac{x_{i,j} - x_{i-1,j}}{d_{i-1}} + \frac{x_{i,j} - x_{i+1,j}}{d_i} + \frac{\partial V(\mathbf{x}_i)}{\partial x_{i,j}} \frac{d_i + d_{i-1}}{2}. \tag{7}$$

The potential gradients are evaluated at each bead at every iteration to provide the action gradients for the next-step generation through Eq. (7). The convergence criterion is based on the vanishing gradients. By equating the right-hand-side of Eq. (7) with zero, discretized Newton's equations of motion are obtained,[24] indicating that the trajectory evolves indeed classically in the imaginary time over the inverted PES.

In the standard RPI calculations, the minimization run needs to be repeated many times in order to converge the instanton (kink) orbit with respect to the kink duration β and the number of beads N. The kink duration has to be long enough to allow the trajectory to connect the symmetric minima within the convergence criterion of the minimization algorithm. The number of beads needs to be high enough to allow a faithful description of the path for the calculation of the prefactor Φ. If the imaginary time step is uniform, $d_i = \beta/N$, a larger duration β requires a larger number of beads N, with most of them clustering at turning points, where the speed of the system (in the upside-down potential) is low. In order to represent the shape of the orbit with fewer beads, unequal time steps d_i have been proposed,[38]

$$d_i = r_i/p_i, \tag{8}$$

where r_i is the length of vector $\mathbf{r}_i = \mathbf{x}_{i+1} - \mathbf{x}_i$, *i.e.*, the Euclidean length (in the mass-scaled coordinate system) of the path segment between bead i and bead $i+1$, and p_i is the absolute value of the mass-scaled velocity of the system in segment i ($\mathbf{p} = \dot{\mathbf{x}}$). The energy of the orbit E is zero (with system at rest at the PES minima), so speed in segment i can be determined

from energy conservation using the mid-segment potential,

$$p_i = \sqrt{2\left(\frac{V_i + V_{i+1}}{2} - E\right)} = \sqrt{V_i + V_{i+1}}, \tag{9}$$

where V_i is the potential at bead i. It was shown that the equidistant beads along the path, $r_i = r$, facilitate the instanton finding, whereas the prefactor calculation requires dense grid near turning points.[38,44] The standard RPI approach often works best using a compromise between equal time steps and equal distances.[38,44]

In the chain-of-states-based methods for finding instantons, the shape of a classical trajectory is determined by exploiting the Jacobi version of the least action principle,[50] in which the time variable is factored out. The abbreviated (Jacobi) action $S_A = \int \mathbf{p} d\mathbf{x}$ at the constant energy $E = 0$ is discretized as,

$$S_A = \sum_{i=1}^{N-1} p_i r_i = \sum_{i=1}^{N-1} \sqrt{V_i + V_{i+1}} |\mathbf{x}_{i+1} - \mathbf{x}_i|, \tag{10}$$

which, we note, is slightly different to earlier expressions.[39,44] The abbreviated action in Eq. (10) is minimized using either the string or NEB method. In the full action minimization above, the beads were kept separated by keeping the time steps d_i fixed and finite. To prevent beads falling on top of each other in the course of the abbreviated action minimization, the beads are redistributed equidistantly after each iteration by interpolation, in the string method, or are kept separated by the introduction of the artificial spring forces that act along the path, in the NEB method. The gradient of abbreviated action in Eq. (10),

$$\begin{aligned} \frac{\partial S_A}{\partial x_{i,j}} &= \frac{r_{i-1}}{2\sqrt{V_{i-1} + V_i}} \frac{\partial V(\mathbf{x}_i)}{\partial x_{i,j}} + \frac{r_i}{2\sqrt{V_i + V_{i+1}}} \frac{\partial V(\mathbf{x}_i)}{\partial x_{i,j}} \\ &+ p_{i-1} \frac{x_{i,j} - x_{i-1,j}}{r_{i-1}} + p_i \frac{x_{i,j} - x_{i+1,j}}{r_i}, \end{aligned} \tag{11}$$

is projected at every iteration to act perpendicular to the path. The so-called upwinding tangents[51] are used to define the path direction for the projection of action gradients, which

facilitate stability of chain-of-states based algorithms.

The above projection scheme eliminates entirely the force on the path ends. We, therefore, fixed the end beads at the minimum geometries, which are determined in advance, and orient them so as to minimize their contribution to the abbreviated action in Eq. (10) following every move of the optimizer.[44] The reorientation of end beads reduces to the least squares problem, e.g., $|\mathbf{x}_2 - \mathbf{x}_1| = \min$, which has an analytical solution for the orientation of $\mathbf{x}_1$[52] in terms of quaternion algebra.

It was demonstrated that the chain-of-states-based methods reduce the number of potential evaluations by two orders of magnitude compared to the standard RPI method for locating instantons.[44] The savings mainly come from the small number of beads N used to represent the instanton path using an equidistant grid in the (mass-scaled) configuration space.

However, the calculation of the prefactor Φ requires many geometries along the path, specifically near the turning points. The imaginary-time dependence of the closely spaced geometries cannot be obtained by the interpolation in imaginary time or by using the velocity-Verlet algorithm with sufficient accuracy. Instead, another standard RPI optimization of the full imaginary-time-dependent action with a much higher number of beads is undertaken in one dimension along the interpolated potential at the converged instanton to determine the geometries at a prespecified dense time grid. Hessian matrix elements at the dense grid of geometries are then determined by interpolation in terms of the path length from the set of hessian matrices evaluated at beads, and the tunneling splittings determined using Eq. (4) without the additional potential/gradient evaluations.

The Euclidean action and abbreviated action at the converged minimum have the same value at zero energy. It can be noted that the Euclidean action gradient, Eq. (7), and the abbreviated action gradient, Eq. (11), are also identical, if one chooses the time steps in Eq. (7) according to the prescription given by Eqs. (8) and (9). Ref. 44 proposed an efficient chain-of-states method based on Euclidean action minimization in which the time steps d_i

were recalculated at every iteration so as to zero the component of action gradient $\nabla_i S$ along the path,

$$\hat{\boldsymbol{\tau}}_i \nabla_i S = 0, \tag{12}$$

where $\hat{\boldsymbol{\tau}}_i$ is the unit tangent vector to the path at bead i, in order that the action gradient does not act along the path to distort the uniform bead distribution. This condition leads to the quadratic equation for d_{i+1} in terms of d_i, with $d_1 = r_1/p_1$. The beads were redistributed equidistantly at every step by interpolation. In fact, Eq. (12) is the least action principle applied to the motion along the current guess for the instanton path. We can think of the current shape of the instanton guess as a holonomic constraint for the motion defined by Eq. (12). The forces of the constraint are perpendicular to the motion and hence do no work. The solution to Eq. (12) is therefore the classical orbit along the current path, the energy of which was determined by setting d_1. The remaining d_i can then be calculated from $E = 0$ condition analytically, as in Eqs. (8) and (9), which is, within the discretization accuracy, equivalent to the numerical solution of Eq. (12), proposed in Ref. 44. Having set the d_i, the perpendicular constraint force is then effectively equal to the perpendicular component of the abbreviated action gradient in Eq. (11) and is used to propagate the string. The string is propagated until the force of the constraint becomes zero and the constraint can be removed. This demonstrates the equivalence (to within the discretization accuracy) of the two chain-of-states methods proposed in Ref. 44, *i.e.*, the full action minimization with the variable time steps and the abbreviated action minimization.

All chain-of-states-based methods, the NEB and the string method using either the full action or the abbreviated action as the object function, proved to be similar in the efficiency of locating instantons.[44] Below, we use the string method with the LBFGS optimizer for the efficiency comparisons with the quadratic string method.

4 QUADRATIC STRING METHOD

Quadratic string method[47] is an extension of the string method, introduced to find minimum energy paths, which uses local quadratic estimates of the PES to propagate the string towards the MEP. This enables us to evaluate the potential less frequently and take larger steps in course of a minimization. In the spirit of Ref. 44 of using the MEP methods for finding instantons, we adapt the QSM method for MEP finding for the instanton search below.

A comparative study[53] found that in the standard string/NEB method for finding MEPs, it is most efficient to use the global LBFGS (without line minimization) optimizer[55] to propagate a chain of beads. All beads are moved simultaneously with a single instance of the optimizer. The study considered optimizers based on the potential/gradient knowledge only. It was noted that the convergence properties of the string/NEB methods are degraded by the non-conservative forces and the non-hermitian character of the hessian matrix, a consequence of the use of the projected forces.[53,54] The strategy behind using the QSM for locating MEPs is to wrap the effect of using non-conservative forces into a local propagation scheme that uses an approximate potential and does not require the evaluation of potential/gradient (*ab initio* or otherwise provided). Instead, the convergence of QSM is governed by the range of validity of the locally approximated potentials, which is controlled by a trust region method.[48]

The local potentials are quadratic expansions at every bead and their performance is reliant on the approximate hessians of the potential. The symmetric hessian matrices are constructed using gradients evaluated at beads along the trajectory they traverse in course of the minimization. In this way, the undesirable properties of the action hessian are sidestepped in terms of the convergence properties, and the convergence depends on the properties of the potential along the transit of beads. The number of potential/gradient evaluations needed for the convergence of QSM depends on the form of the potential that needs to be traversed by the beads and the quality of the local quadratic surface representation in terms of the approximate symmetric hessians. The effect of coupling terms between beads in Eq. (11) on convergence is, along with the effect of projection forces, wrapped into the local propagation,

which requires no additional potential evaluations.

The outline of the QSM is given below, with further details described in the following sections:

(I) The geometries of the two symmetric minima are assumed to be known in Cartesian coordinates (or determined using an f-dimensional LBFGS minimisation). The initial path with N beads is constructed with end beads fixed in two symmetric potential minima. The intermediate beads are distributed by interpolation (see Section 4.4) using either the end points only, or the end points supplemented by a set of points aimed at guiding the path away from high-potential regions, or the path obtained in a previous calculation.

(II) Evaluate the potential and gradients $\mathbf{g}_i$ at all $N-2$ intermediate geometries i. Potential at end beads is defined as zero.

(III) Construct the action gradient using Eq. (11) and stop if a chosen convergence criterion based on the perpendicular components of the action gradient is satisfied.

(IV) Update each local potential hessian $\mathbf{H}_i$ at bead i using the new potential and its gradient.

(V) Update each trust radius Δ_i.

(VI) Integrate $\mathbf{x}_i$ along the steepest descent of action surface over the quadratic potentials given by $\mathbf{H}_i$ in order to minimize the abbreviated action, stopping when one bead reaches the trust radius, negative potential region, or when the MAP is found.

(VII) Redistribute beads along the path by interpolation (if required). Go back to (II).

4.1 Hessian update

In order to move the path towards the minimum of action without evaluating potential/gradients, a local quadratic surface $m_i(\mathbf{x}_i)$ is constructed at each intermediate bead i along the path as the second order Taylor series,

$$m_i(\mathbf{x}_i) = V(\mathbf{x}_i^k) + \nabla_i V(\mathbf{x}_i^k)\delta\mathbf{x}_i + \frac{1}{2}\delta\mathbf{x}_i^{\mathrm{T}}\mathbf{H}_i^k\delta\mathbf{x}_i, \tag{13}$$

where $V(\mathbf{x}_i^k)$ and $\nabla_i V(\mathbf{x}_i^k)$ are the potential and gradient evaluated (*ab initio* or otherwise) at $\mathbf{x}_i^k$, the position of bead i at k^{th} iteration, $\delta\mathbf{x}_i = \mathbf{x}_i - \mathbf{x}_i^k$ and $\mathbf{H}_i^k$ is an approximate hessian, $(\mathbf{H}_i^k)_{j'j} = \partial^2 V(\mathbf{x}_i^k)/\partial x_{ij'}^k \partial x_{ij}$. The approximate hessian is updated at every iteration based on gradient knowledge according to one of the standard quasi-Newton updating formulae.

Quasi-Newton updating formulae maintain the symmetry of the hessian matrix and satisfy the quasi-Newton condition (with more details given in Appendix). The following schemes are used in the present study.

BFGS (Broyden-Fletcher-Goldfarb-Shanno) update[56] is the most prevalent updating scheme in global minimizations. It preserves the positive definiteness of the hessian throughout if certain conditions are met (see Appendix). The path connecting the symmetric minima has to pass through a maximum, where at least one of the hessian eigenvalues is negative. BFGS can nevertheless be useful, as the negative eigenvalue usually lies along the path and the hessian of the potential is usually positive definite in the hypersurface perpendicular to the path, within which the search is performed. The wrong curvature of the potential tangential to the path is projected out during the minimization. Alternatively, one can implement the right curvature along the path into the hessian matrix explicitly, as shown below.

SR1[57] is the simplest symmetric rank-1 updating scheme. A safeguard needs to be implemented and the update skipped, if conditions for a breakdown are met. More details are given in Appendix. SR1 does not guarantee positive definiteness, so it is appropriate to describe hessians with negative eigenvalues along instantons.

PSB (Powell-Symmetric-Broyden)[58] update is a non-positive-definite rank-2 update that does not fail in the same way as SR1. More details are given in Appendix.

Bofill update[59] is a combination of SR1 and PSB updates. It switches to PSB update whenever the conditions for failure of SR1 are approached. Bofill's method is the method of choice for transition state searches, when one or more hessian eigenvalues are expected to be negative.

In similar fashion, Ayala and Schlegel update[60] uses a combination of BFGS and PSB updates. The BFGS update is predominantly used for displacements perpendicular to the path, whereas the PSB dominates for moves nearly parallel to the path or along negative eigenvalue directions.

If beads represent the path faithfully, we can determine the curvature of the potential along the path with relative accuracy and use this information to improve the quality of approximate hessians. We parametrize the path in terms of the arc length s and determine tangents $\hat{\boldsymbol{\tau}}_i = d\mathbf{x}_i/ds$ at each bead, as described in Section 4.4. The potential $V(\mathbf{x}_i)$ and its derivative along the path, $dV(\mathbf{x}_i)/ds = \hat{\boldsymbol{\tau}}_i \nabla V(\mathbf{x}_i)$, at $\mathbf{x}_{i-1}$, $\mathbf{x}_i$ and $\mathbf{x}_{i+1}$ can be used to construct a quintic polynomial interpolant of the potential along the path, which then yields the second derivative of the potential along the path at bead i,

$$\frac{d^2V(\mathbf{x}_i)}{ds^2} = \frac{d}{ds}\left(\nabla V(\mathbf{x}_i)\hat{\boldsymbol{\tau}}_i\right) = \hat{\boldsymbol{\tau}}_i \mathbf{H}_i \hat{\boldsymbol{\tau}}_i + \nabla V(\mathbf{x}_i)\frac{d\hat{\boldsymbol{\tau}}_i}{ds}, \tag{14}$$

where $(d/ds)\hat{\boldsymbol{\tau}}_i = \kappa \hat{\boldsymbol{N}}_i$, κ is curvature and $\hat{\boldsymbol{N}}_i$ is unit normal to the path. The procedure was used in Ref. 60 for MEP searches, but the last term in Eq. (14) was missing. This term may become important when search is nearing the converged instanton. In MEP applications, it is zero for the converged MEP, because the potential gradient and the path normal are perpendicular to each other. The updated hessian at $\mathbf{x}_i$ can be supplemented by the d^2V_i/ds^2 knowledge using

$$\mathbf{H}_i^{\text{updated}} = (1 - \hat{\boldsymbol{\tau}}_i\hat{\boldsymbol{\tau}}_i^{\mathrm{T}})\mathbf{H}_i(1 - \hat{\boldsymbol{\tau}}_i\hat{\boldsymbol{\tau}}_i^{\mathrm{T}}) + \left(\frac{d^2V(\mathbf{x}_i)}{ds^2} - \nabla V(\mathbf{x}_i)\frac{d\hat{\boldsymbol{\tau}}_i}{ds}\right)\hat{\boldsymbol{\tau}}_i\hat{\boldsymbol{\tau}}_i^{\mathrm{T}}, \tag{15}$$

where use has been made of Eq. (14). The path normal at bead i can be obtained by a cubic spline interpolation of bead positions in terms of arc length, $\boldsymbol{N}_i = d^2\mathbf{x}_i/ds^2$. Alternatively, it can be determined by finite differences using coordinates of neighboring beads, $\mathbf{x}_{i-1}$ and $\mathbf{x}_{i+1}$.

We also considered using more accurate updating formulae by Wu *et al.*[61] Previous ap-

plications in MD simulations[62] included occasional evaluation of exact hessians. If one starts without the knowledge of initial hessian and does not evaluate it in course of a minimization, we found that the use of those formulae is unsuitable.

4.2 Trust radius update

The trust radius Δ_i^k defines the region in which the quadratic approximation to the potential around the bead i in Eq. (13) is considered valid. Initial trust radius is chosen as the Euclidean length of the path segment (distance to the nearest-neighbor bead), unless it is user specified. Its value is updated when the next position of the bead is reached and potential evaluated at $\mathbf{x}_i^{k+1} = \mathbf{x}_i^k + \delta\mathbf{x}_i^k$, where k counts iterations. Potential $V(\mathbf{x}_i)$ is used as a merit function and m_i in Eq. (13) provides the expected value. The parameter ρ_i is determined using

$$\rho_i = \frac{V(\mathbf{x}_i^{k+1}) - V(\mathbf{x}_i^k)}{m_i(\delta\mathbf{x}_i^k) - m_i(0)}. \tag{16}$$

Based on the parameter ρ_i, which measures how close the merit function is to the expected value at bead i, the trust radius is updated as

$$\Delta_i^{k+1} = \begin{cases} \sqrt{2}\Delta_i^k & \text{if} \quad 0.75 < \rho_i < 1.25 \quad \text{and} \quad 1.25|\delta\mathbf{x}_i^k| > \Delta_i^k \\ 0.25\Delta_i^k & \text{if} \quad \rho_i > 1.75 \quad \text{or} \quad \rho_i < 0.25 \\ \Delta_i^k & \text{otherwise} \end{cases} . \tag{17}$$

The prescription in Eq. (17) is slightly more conservative than in the original QSM for MEP search,[47] and it is similar to the prescriptions for transition state search.[59] The performance of the method should only weakly depend on the details of the formulation.

4.3 Integration along the steepest descent

After a local quadratic approximation to the PES has been constructed, we integrate the positions of beads along the steepest descent direction on the action surface, perpendicular

to the path. The differential equation that governs the motion of beads is

$$\frac{d\mathbf{x}_i}{dt} = (1 - \hat{\boldsymbol{\tau}}_i \hat{\boldsymbol{\tau}}_i^{\mathrm{T}}) \frac{dS_A^{\mathrm{quad}}}{d\mathbf{x}_i}, \tag{18}$$

where t is the parameter of integration. The action gradient $dS_A^{\mathrm{quad}}/d\mathbf{x}_i$ is obtained from Eq. (11), where the quadratic approximation of the potenial m_i in Eq. (13) is used in place of V_i and the potential gradients replaced by the approximate expression

$$\nabla_i m(\mathbf{x}_i) = \nabla_i V(\mathbf{x}_i^k) + \mathbf{H}_i^k \delta\mathbf{x}_i. \tag{19}$$

The integration of Eq. (18) does not require further (*ab initio* or subroutine-provided) potential/gradient evaluations.

Original QSM formulation[47] uses Runge-Kutta adaptive step-size method[63] for integration of Eq. (18). The method adjusts the step size Δt based on the accuracy needed ϵ_0. It determines the error of a given step by comparing to the lower-order (RK4) to higher-order method and updates the step size as described in Ref. 63. The desired accuracy of integration is set to $\epsilon_0 = \max(\max(|dS_A^{\perp}/d\mathbf{x}_i|/10, \epsilon)/(10||d^2 S_A^{\perp}/d\mathbf{x}_i^2||))$, in analogy with Ref. 47. Here, $\epsilon = \max(\nabla_i S^{\perp})$ is the preset accuracy of the instanton path based on the vanishing of the perpendicular action gradients, and a rough estimate of $||d^2 S_A^{\perp}/d\mathbf{x}_i^2||$ is obtained by adding norms of the derivatives of each term on r.h.s. in Eq. (11), neglecting terms that couple different beads and using perpendicular projection of the local updated hessian.

Alternatively, we use a damped molecular dynamics (DMD) scheme, similar to the original NEB optimizer[40] or FIRE[64] without the step variation. The force (r.h.s. of Eq. (18)) acts perpendicular to the path and accelerates every bead. At every step, a portion of the velocity of each bead is kept and a portion is redirected to act along the force at that bead. Velocities of all beads are reset to zero when the Nf-dimensional force becomes antiparallel to the full Nf-dimensional velocity.

Using one of the above integrators, the string of beads is propagated until a minimum of

action on the approximate potential is reached or a bead reached its trust radius or an area of negative potential (on approximate surface) is entered. The integration is then stopped for potential/gradient evaluation, followed by hessian and trust-radius updates. The updates can either be performed simultaneosly or for each bead independently, when one of the criteria for stopping is reached (negative potential or trust radius limit).

4.4 Redistribution of beads

In the string method, beads need to be regularly redistributed to keep the bead spacings at a desired level. We keep beads approximately equally spaced and achieve this by redistributing them after every local integration step or, in a more relaxed way, after bead spacings have become uneven. The criterion used for respacing the beads is

$$\max_i ||\mathbf{x}_{i+1} - \mathbf{x}_i| - |\mathbf{x}_i - \mathbf{x}_{i-1}|| > \frac{\sum_i |\mathbf{x}_{i+1} - \mathbf{x}_i|}{10(N-1)}. \tag{20}$$

Both approaches worked equally well in all our tests.

Beads are redistributed using interpolation of Cartesian coordinates of beads with arc length as the parametric variable. Linear interpolation and use of upwinding tangents[51] is a stable way to propagate the string using Eq. (18).

Natural cubic splines can be used when higher accuracy is needed. We interpolate bead coordinates using splines with distances between beads (in mass-scaled coordinates) as arc lengths. We use a dense grid on the interpolated path to obtain arc lengths more accurately and subsequently reinterpolate bead coordinates using improved arc lengths. Natural cubic splines provide tangents as $\hat{\boldsymbol{\tau}}_i = d\mathbf{x}_i/ds$ and normals as $\boldsymbol{N}_i = d^2\mathbf{x}_i/ds^2$ at beads, which can be used for hessian updates in Eq. (15) and force projection scheme in Eq. (18).

5 MIL'NIKOV-NAKAMURA FORMULA

After we have obtained the minimum action path, the hessians of the potential need to be evaluated at beads along the path in order to calculate the contributions of the neighboring paths to the tunneling splitting contained in the prefactor Φ in Eq. (6). In the standard RPI approach, the prefactor Φ is evaluated as the ratio of the action hessian eigenvalues, obtained by the diagonalization of the Nf-dimensional banded matrix of the discretized action hessian, which is a time-consuming step.

Alternatively, the prefactor can be obtained by exploiting the relationship between the ratio of action determinants and that of Jacobi fields[65] and taking the zero-temperature limit.[1,21] Mil'nikov and Nakamura generalized the original one-dimensional treatment[22] and obtained, after a lengthy and non-trivial derivation, the expression

$$\frac{1}{\Phi} = \sqrt{\frac{2}{S_{\rm kink}}}\sqrt{\frac{\det A(-\infty)}{\det A^{\perp}(0)}}\exp\left[\int_{-\infty}^{0} d\tau \left[{\rm Tr}A(-\infty) - {\rm Tr}A(\tau)\right]\right], \tag{21}$$

where $A(\tau)$ is the log-derivative of a Jacobi field, which satisfies the Riccati equation,

$$\frac{d}{d\tau}A(\tau) = \mathbf{H}(\tau) - A(\tau)^2, \tag{22}$$

where τ is the imaginary time with $\tau = \pm\infty$ at the symmetric minima, $\mathbf{H}(\mathbf{x}(\tau))$ is the hessian of the potential at the geometry $\mathbf{x}$, where the system is situated at the imaginary time τ,

$$\mathbf{H}_{j'j}(\tau) = \partial^2 V(\mathbf{x}(\tau))/\partial x_{j'}\partial x_j. \tag{23}$$

$A^{\perp}(0)$ in Eq. (21) is the matrix A projected to the $(f-1)$-dimensional configuration space perpendicular to the instanton path at the imaginary time $\tau = 0$, which corresponds to the symmetrical mid point of the path. The initial condition for the desired solution of Eq. (22) is

$$A^2(-\infty) = \mathbf{H}(-\infty). \tag{24}$$

We discretized the integrand in Eq. (21) on an equally-spaced time grid τ_i, specified by the imaginary time duration β and the size of the grid N. The system geometries on the time grid $\mathbf{x}(\tau_i)$ where obtained in a 1D full action minimisation on the interpolated potential along the instanton. The hessian matrices at $\mathbf{x}(\tau_i)$ are obtained from the hessian matrices at the N beads by cubic spline interpolation. The integration of Eq. (22) in time was performed by the log derivative algorithm using modified Simpson's rule and a constant reference potential.[66] The integration of Eq. (22) was started at a finite imaginary time τ_k that corresponds to the geometry $\mathbf{x}(\tau_k)$ on the grid that is closest to a prespecified offset distance away from the minimum of the potential along the instanton to avoid the singularity in the integral in Eq. (21). The initial $A(\tau_k)$ was obtained by linearization procedure as described in Ref. 22 and the integral in Eq. (21) was evaluated using trapesium rule.

6 TUNNELING SPLITTING CALCULATIONS

In numerical tests below, the QSM is tested in tunneling splitting calculations on the hydrogen transfer in malonaldehyde and on umbrella motion in the inversion of ammonia and hydronium. The efficiency of QSM is measured in terms of the number of potential/gradient calls to the external routine that provides them, and it is compared to the string method with the global LBFGS optimizer.[44] Numerical overhead of the QSM is considered negligible. The initial guess is the linear path with equidistant beads in mass-scaled coordinates connecting the two symmetric minima. The end bead geometries are fixed at potential minima. The orientation of end beads and half of the intermediate beads are optimized, whereas the other half is obtained by symmetry. The convergence parameter is $\epsilon = \max(\nabla_i S^{\perp})$, the maximum absolute value of the perpendicular-to-the-path action gradient.

6.1 MALONALDEHYDE

The potential energy surface of malonaldehyde by Sewell *et al*[67] was used for testing of the string method and other gradient-based search methods for finding instantons,[22,25,44] so it provides an ideal benchmark system for testing the QSM. The malonaldehyde molecule has 27 degrees of freedom and the surface is full-dimensional and analytic (with analytic gradients). It is, however, inaccurate, with energy barrier of 3497 cm^{-1} along MEP and tunneling splitting of ≈ 51 cm^{-1} [22,44] obtained by the instanton method. The experimental value is 21.6 cm^{-1}.[68]

Ref. 44 provides the details of the tunneling splitting calculations on this model system using the string method. There is a slight modification in the discretization of the action integral in Eq. (10) and correspondingly in the gradient, Eq. (11) (to be compared with Eq. (15) and (17) in Ref. 44), so the iteration count obtained using the string method with LBFGS optimizer here may not be identical to that obained in Ref. 44. The tunneling splittings were determined sufficiently accurately to within 3 % (1 %) using 18 (27) equally spaced beads. With 2 beads fixed in potential minima and half of the path obtained by symmetry, optimization was 8×27 (13×27)-dimensional.

The QSM and string method optimize the same object, a string of equidistant beads, so they require the same number of beads N for a given accuracy in the tunneling splitting Δ. The difference comes in the number of iterations required for a given N and convergence parameter ϵ. In QSM method, we have used the DMD (damped molecular dynamics) optimizer described in Subsection 4.3, with propagator parameters fixed to values given in Ref. 64, apart from the 'time step' that was fixed to 1.5 a.u. Local potential hessians at each bead were initialized to diagonal matrices with $|\nabla_i V(\mathbf{x}_i)|/\Delta_i^0$ on diagonal, where Δ_i^0 is the initial trust radius. We fixed Δ_i^0 to 5 $m_e^{1/2} a_0$ in our first set of calculations. Updates were performed simultaneously on all beads, whenever one of the beads met criteria for updating (partial convergence, trust radius violation or negative potential encounter). First update on all beads was performed if all beads have moved by 0.5 $m_e^{1/2} a_0$, to guarantee that the first

update takes place relatively soon. We also place a limit on the minimum trust radius of $0.1 m_e^{1/2} a_0$ to prevent updating too often near zero potential and in the non-quadratic regions. At each update k, the convergence criterion ϵ^k is set to $\max(\max(|\nabla_i S^\perp|/10), \epsilon)$ to serve as a criterion for the partial convergence in order to prevent many integrator steps away from the true minimum of action (but near the minimum on the approximate potential). Hessian updates were performed using Bofill formula, Eqs. (33) and (34). Upwinding tangents[51] were used to define tangents for the gradient projections and beads were redistributed using linear interpolation at every step of the integrator. We have also tried using Eq. (20) with no noticable gains in efficiency.

Comparison between the string method and QSM is shown in Fig. 1. The number of iterations in the string method grows approximately lineraly with the number of beads N. With 18 (27) beads, the number of gradient evaluations for converging the action gradient to tolerance $\epsilon = 10^{-4}$ a.u. is 568 (1261). This can be compared to 216 (299) gradient evaluations with QSM. With 40 beads, the number of gradient evaluations for the string method / QSM is 3496 / 456. This ratio does not change appreciably when tolerance ϵ is reduced.

Fig. 1 shows that the number of iterations required by QSM remains approximatelly constant with N; it does not depend strongly on the number of beads used to represent the path. As we shall see later (inset of Fig. 3), the number of steps of the DMD integrator does grow with N. The number of iterations of QSM, on the other hand, depends on how far the string has travelled and properties of the potential, on how quadratic the surface is. It shows that the bead-coupling forces in Eq. (11) have successfully been absorbed in the DMD integrator, which requires no additional gradient calls.

Fig. 2 shows the influence of different updating techniques in QSM. Results are shown for damped BFGS, SR1, PSB, Bofill and Ayala-Schlegel updates. The formulae are summarized in Appendix. Initial trust radius was set to 5 $m_e^{1/2} a_0$. It is immediately obvious that no method has a clear advantage over the rest. Ayala-Schlegel updating gives best results by a narrow margin. The next best is the QSM with Bofill updating. SR1 updating gives worst

results. BFGS updating resulted in 5 failures (over 36 calculations with $N = 5 - 40$). When we added the tangent updating (dashed gray line) using Eq. (15), we managed to avoid the failures of convergence using BFGS updates. The onset of failure of the damped BFGS was always (in all our tests, not only those shown in the figure) associated with the moves that break the curvature condition in Eq. (27) (which implies a skip of update in the damped BFGS). Additional tangent updates in general did not bring a performance advantage, but improved the stability of calculations that use the damped BFGS updates.

The damped BFGS update provides a positive definite hessian by construction, whereas the instanton has at least one negative hessian eigenvalue along its path. BFGS usually works because tangential moves are projected out and the remaining $(f-1)$-dimensional hessian is positive definite. Alternatively, the negative eigenvalue can be supplied using Eq. (15). Nevertheless, through numerous tests (using different initial paths and time steps in DMD), we found that BFGS and SR1 are less stable than the other updating schemes tested above.

In Fig. 3, we again study the QSM with Bofill updates. This time we set the initial trust radius to the average distance between beads at the start of the calculations, in an attempt to avoid having to make a choice about its value at start. The solid black line shows the number of iterations as a function of the number of beads. The number of iterations slowly grows with the number of beads N. The solid red line in Fig. 3 shows the dependence with Δ_i^0 set to 5 $m_e^{1/2} a_0$ for comparison. The curve is flattened and shows no growth with N. The starting path length is 71 $m_e^{1/2} a_0$, which means that for $N > 14$, the initial trust radius is shorter than 5 $m_e^{1/2} a_0$. So the larger trust radius and less updating leads to a better performance at large N. In this test system, more frequent updating is unnecessary for large N. The trust region mechanism has not equilibrated over a short propagation ($20 - 40$ updates) and efficiency depends on the initial trust radius choice. The initial trust radius set at the average bead distance (at the start of the calculation) provides a good initial guess for the converged tunneling splittings (within 3%[44]) with $N = 18$.

In Fig. 3, we have also tried using the QSM with Runge-Kutta (RK) integrator, described

in Subsection 4.3, and compared it to the QSM with DMD integrator. When RK integrator is used, we implemented a safeguard to prevent integration without progress. Updating is triggered if the number of RK steps exceeds 500 or if the RK step becomes negligible. The total number of integrator steps in an instanton optimization is much larger with the RK integrator (dashed line) in comparison with the DMD (solid line), as shown in the inset in Fig. 3, but the performance in terms of the number of QSM iterations is similar (initial trust radius set to the average bead distance is used).

We also tried the QSM with the individual updating instead of the simultaneous updating used in the calculations above. In the individual updating, we update only the beads that reached the trust radius or the area of negative potential (symmetrically). For the first 5 updates and also if partial convergence was accomplished, all beads are updated simultaneously. Solid blue line in Fig. 3 shows the 'effective' number of iterations as a function of N. Each update of a bead accounts for $1/(N-2)$ iterations for comparison with the simultaneous updates. The figure shows that the performance of the individual and simultaneous updating is similar. The dotted blue line shows the number of string geometries at which the updating takes place showing that the individual updates do often take place after 5 simultaneous updates at the start (which are forced). When nearing the full convergence, the simultaneous updates again take over as partial convergences are reached.

Fig. 4 shows the rate of convergence of the string method using QSM (Bofill updates) and LBFGS (with the global optimizer) using 18 beads (for which the tunneling splittings are converged to within 3 %) (red line) and 40 beads (black line). The initial trust radius was set to 5 $m_e^{1/2} a_0$. It can be seen that the QSM is capable of superlinear convergence. It is clear that the superior performance of the QSM holds over orders of magnitude of accuracy. The approximate N-independence of QSM persists for all accuracies. The QSM requires about 3 times less potential/gradient evaluations than the string method/LBFGS for a converged tunneling splitting calculation ($\approx 1\%$) in this system and the ratio grows with the added beads and accuracy.

Next, we compare the instanton search with the MEP search. The convergence criterion based on the gradient of action and potential, respectively, needs to be changed (as respective gradients have different units). We measured the average root-mean-square distance to the converged result with the same number of beads N and set $\epsilon = 0.1\ m_e^{1/2} a_0$ (which corresponds to the gradient-based $\epsilon \approx 10^{-4}$ a.u.). We again compare the efficiency of the string method and the QSM for locating instantons (red line) and MEPs (black line) in Fig. 5. QSM is used with $\Delta_i^0 = 5\ m_e^{1/2} a_0$, Bofill updates and the DMD integrator. The DMD time step was set to 1.5 a.u. in the instanton search, as above, and to 25 a.u. in the MEP search.

Fig. 5 shows again that the QSM performance is superior to the string method. The efficiency of instanton search is comparable to the MEP search for the string method. The average r.m.s. distance of the starting path to instanton and MEP are 19 and 69 $m_e^{1/2} a_0$, respectively. The longer distance to the solution is compensated by the less complicated force scheme (no inter-bead couplings) in the MEP search. In QSM, the number of iterations remains independent of the number of beads for both, the instanton and the MEP search, which proves that the distance to the solution is the main factor affecting the performance. The instanton search clearly overperforms MEP search, because the burden of the more complicated force scheme is carried by the DMD integrator, and the distance to the solution is smaller for the instanton.

In Fig. 6, we plot the average r.m.s. distance of the converged path with N beads to the fully converged path ($N \to \infty$) against the number of beads N. It can be seen that for a given accuracy of the path, MEP requires a much larger number of beads. The inset in Fig. 6 shows that the MEP is 2.5 times longer than the instanton path. Both paths have end points in PES minima. The difference in length is due to the corner-cutting effect of the minimum action path, which is very pronounced in this system. The conclusion is that in the model system of malonaldehyde the full tunneling splitting calculation is computationally less expensive than finding the MEP.

6.2 AMMONIA

For further validation of the QSM, we tested the method on inversion tunneling in ammonia. The potential energy surface by Marquardt *et al*[69] was used, which is again analytical and full dimensional. It is obtained by a fit to *ab initio* data on a combination of the CCSD(T) and MRCI levels of theory with aug-cc-pVQZ basis. The surface has inversion barrier of 1774 cm^{-1}.

We optimized the instanton using the string method with LBFGS and QSM. The number of iterations as a function of number of beads is shown in Fig. 7. The starting path length was the linear interpolation between minima and is 95.76 $m_e^{1/2}a_0$ long. The converged instanton ($N \to \infty$) is 97.47 $m_e^{1/2}a_0$, with the average r.m.s. deviation from the starting path of 5.72 $m_e^{1/2}a_0$. We note here that the length of MEP is 97.61 $m_e^{1/2}a_0$ with the average r.m.s. deviation from the starting path of 5.98 $m_e^{1/2}a_0$ and 0.259 $m_e^{1/2}a_0$ from the instanton path. This makes the calculation trivial, but realistic in terms of the accuracy of the potential. The trust radius was set to the average bead distance at the starting path. We used all hessian updating formulae listed in Appendix. Only results obtained using Bofill and Ayala-Schlegel updates are shown. The other updates gave almost identical results with a maxium ± 1 iterations difference in each calculation. The advantage of QSM over the string method using LBFGS begins to show for $N > 10$ beads and grows to a factor of 3 advantage at $N = 40$.

The converged instanton was interpolated using splines and a 1D full action minimization with $\beta = 4000$ a.u. and $N = 401$ using equal imaginary time steps was performed in order to obtain bead positions on an imaginary time grid. The prefactor was calculated using Eq. (21) with the integrand obtained by a log-derivative propagation of Eq. (22) on the precalculated imaginary time grid, with the offset of 0.1 $m_e^{1/2}a_0$ along the instanton away from the minimum. The splittings Δ were obtained using Eq. (4). The convergence of the splittings is shown in the inset in Fig. 7. The results with 4 beads and above are already within 15 % of the fully converged result. The splitting Δ is converged within 3 % using

12 beads and within 1 % using 20 beads. Using 20 beads, QSM requires 5 iterations for convergence, which means $9 \times 5 = 45$ potential/gradient evaluations. String method requires 11 iterations and 99 potential/gradient calls.

The comparison of tunneling splittings and the associate actions for the isotopically substituted ammonia molecules obtained using the RPI method is given in Table I, together with accurate quantum mechanical results of Ref. 69 and the experiment.[70–73] Results deviate up to 40 %, which is in line with the expectancies of the instanton theory.

Table 1: Tunneling splittings in ammonia obtained using RPI-QSM. Comparison to accurate quantum results by Marquardt *et al*[69] and experiment.[70–73]

Δ / cm^{-1}	instanton	Marquardt *et al*[69]	experiment	Action / a.u.
NH_3	0.476	0.75	0.79[70]	8.085
NDH_2	0.250	0.38	0.406[71]	8.707
NHD_2	0.108	0.16	0.171[72]	9.499
ND_3	0.0344	0.05	0.053[73]	10.55

6.3 HYDRONIUM

Another realistic example is the inversion tunneling in hydronium ion. We use the potential by Huang *et al*,[74] a fit to *ab initio* energies obtained using CCSD(T) and aug-cc-pVQZ basis set. The inversion barrier is 690 cm^{-1}.

We again optimized instantons using the string method and QSM and show the number of iterations vs. number of beads in Fig. 8. Linear starting path is 72.59 $m_e^{1/2} a_0$ long, whereas the instanton and MEP are 73.42 and 73.46 $m_e^{1/2} a_0$ long, respectively. Average r.m.s. distance of the instanton to the starting path and MEP are 3.46 and 0.0915 $m_e^{1/2} a_0$. This is another example where the solution is pratically within the trust radius, initially set to the average bead separation at the starting path, but it is also a realistic example. We again tried using all hessian updating formulae listed in Appendix, with only ± 1 iterations difference between them when converged. Only results obtained using Bofill and Ayala-Schlegel updates are

shown in Fig. 8. For $N > 10$ beads, QSM again dominates the string method. At 40 beads, efficiency ratio between the string and QSM is 2.6.

The converged instanton was interpolated using cubic splines and a 1D action minimization with $\beta = 5000$ a.u. and $N = 201$ using equal imaginary time steps was performed in order to obtain the grid for the prefactor calculation. Eq. (22) was solved using a log derivative propagation on the grid with the starting offset of 0.1 $m_e^{1/2} a_0$ along the path, and the splittings obtained using Eqs. (21) and (4). Convergence of the splittings is shown in the inset in Fig. 8.

The converged instanton was interpolated using splines and a 1D full action minimization with $\beta = 4000$ a.u. and $N = 401$ using equal imaginary time steps was performed in order to obtain bead positions on an imaginary time grid. The prefactor was calculated using Eq. (21) with the integrand obtained by a log-derivative propagation of Eq. (22) on the precalculated imaginary time grid, with the offset of 0.1 $m_e^{1/2} a_0$ along the instanton away from the minimum. The splittings Δ were obtained using Eq. (4).

The results with 4 beads and above are already within 16 % of fully converged result. Δ is converged within 3 % using 11 beads and within 1 % using 19 beads. Using 19 beads, QSM requires 4 iterations, which means $9 \times 4 = 36$ potential/gradient evaluations. String method requires 9 iterations and 81 potential/gradient calls.

The comparison of tunneling splittings and the associate actions for the isotopically substituted hydronium ions obtained using RPI-QSM is given in Table II, together with accurate quantum mechanical results of Ref. 74 and the experiment.[75,76] Instanton results deviate up to 50 % from the quantum results, which is, again, in line with the expectancies of the instanton theory.

The instanton results in hydronium and ammonia above show similar or better agreement to quantum results than the competing reduced dimensional quantum theories.[77] The full power of instantons lies in the applications in systems that are beyond reach of quantum calculations and the full power of QSM lies in the systems with demanding potentials. The

above tests on simple systems give us the reassurance that the tunneling splittings in large systems can be obtained using RPI-QSM with signicant computational savings.

Table 2: Tunneling splittings in hydronium obtained using instantons and compared to accurate quantum results by Huang *et al*[74] and experiment.[75,76]

Δ / cm^{-1}	instanton	Huang *et al*[74]	experiment	Action / a.u.
H_3O^+	67.0	46	55.35[75]	3.679
DH_2O^+	48.4	33	–	3.968
HD_2O^+	31.6	22	–	4.339
D_3O^+	17.5	12	15.35[76]	4.836

6.4 COMPUTATIONAL CONSIDERATIONS

The search for the instanton and the evaluation of hessians along the instanton both require a significant computational effort. The above results show that the QSM brings considerable efficiency savings to the instanton search. We examine below how does the share of the effort of the instanton search scale up in larger systems.

The instanton search is based on the potential and gradient evaluations. For illustrative purposes, we assume that gradients are obtained using a 3-point stencil, which requires $2f$ potential calls. For a complete instanton search, the number of potential evaluations needed is $N_{\text{iter}} \times (N-2) \times (2f+1)$, where N_{iter} is the number of iterations in a converged calculation, whereby potential and gradient are evaluated at $N-2$ beads ($N-1$, if the symmetry of the path is included explicitly) at every iteration.

If analytical hessians are not available, the hessian at a bead can be obtained with sufficient accuracy from the potential evaluations in the neighborhood of the bead using finite differences based on a 3-point stencil (using formulae 25.3.23 and 25.3.27 from Ref. 78). The hessian evaluation based on 3-point stencil requires f^2+f+1 potential evaluations, where $2f+1$ of them are already known from the instanton search. The added effort is f^2-f potential evaluations (or similar, if using other schemes of comparable accuracy are used).

Hessians need to be evaluated at all beads along the MAP resulting in $\approx N \times f^2$ potential evaluations. Evaluating hessians at a denser or sparser grid along the instanton did not bring any accuracy gains or efficiency savings.

Therefore, the instanton search is more demanding than the subsequent hessian evaluation if $N_{\mathrm{iter}} > f/2$, approximately, which is satisfied for all but the simplest systems.

The number of iterations N_{iter} in the string method based on the global LBFGS optimizer in a converged calculation scales itself proportionaly to the number of degrees of freedom being optimized, $N_{\mathrm{iter}} \sim Nf$. This means that in the high-accuracy calculations, or for the more demanding paths that require a large number of beads N for a faithful description, the instanton search will dominate the hessian evaluation more drastically.

The number of iterations N_{iter} in the QSM is proportional to the number of degrees of freedom f, where the remaining effort proportional to Nf is picked up by the local integrator, as can be seen in Figs. 1 and 3. The consequence of the above is that the share of the effort of the instanton search using the QSM in the tunneling splitting calculations is expected to remain independent of the system size. It will however depend on how far the initial guess is from the solution and on the complexity of the potential and dynamics involved. In the malonaldehyde test case, with a a pronounced skeletal rearrangement and a relatively large corner-cutting effect, the effort of instanton search using the QSM with 18, 27 and 40 beads is $\approx$ 1.88, 1.67 and 1.79 times more demanding, respectively, than the subsequent hessian evaluation, in terms of the number of potential evaluations under the assumptions given above. In the trivial examples of ammonia (and hydronium) with $N = 20$ and 40, the effort ratio is 0.85 and 0.90 (and 0.68 and 0.90), meaning that the instanton search takes less time than the hessian evaluation (as the initial guess is already very close to the instanton in those systems). The numbers are illustrative as the above calculations use analytical gradients, which are expected to aid the convergence. The numerical effort of the prefactor calculation using the Mil'nikov-Nakamura formula is negligible in comparison.

7 CONCLUSIONS

We have proposed the quadratic string method for locating instantons in the calculations of tunneling splittings in molecular systems, as an effective way of reducing the number of *ab initio* potential/gradient evaluations compared to other state-of-the-art methods. Instanton path is represented by a string of equidistant beads in Cartesian coordinates, where each bead represents the molecular geometry along the tunneling path. The end beads remain fixed at potential minima. The quadratic string method, first developed for locating MEPs, was modified to minimize the abbreviated action with bead coordinates as parameters. The path is iteratively improved upon by minimizing abbreviated action in the local quadratic potentials centered at beads, constructed using updated hessians. The method requires potential/gradients along the path at every iteration and is thus trivially parallelizable.

The QSM was tested on a model potential of malonaldehyde, and on inversion tunneling in ammonia molecule and hydronium ion, where the instanton tunneling splittings were obtained for all isotopically substituted systems. The computational cost of locating instantons using the QSM is about three times lower than the cost of using the string method with LBFGS optimizer, measured in terms of the number of potential/gradient evaluations. The method is general, as it uses Cartesian coordinates, and widely applicable, as locating instantons takes less effort than locating MEPs. The instanton search using the QSM takes a comparable amount of time to the hessian evaluation along the path. The prefactor was obtained from hessians at beads using the Mil'nikov-Nakamura formula in Cartesian coordinates with little extra effort. The QSM relies on the techniques for updating hessians and a trust region algorithm that have both been extensively tested in the MEP search applications. With this added reassurance, having in mind that the tunneling paths are usually simpler than MEPs, the QSM is expected to provide a simple and powerful way of calculating tunneling splittings using instanton theory in large systems of systems with more complex motions and in combination with expensive potential energy surfaces or on-the-fly electronic structure calculations.

Acknowledgement

This work has been fully supported by Croatian Science Foundation under the project IP-2014-09-7540.

APPENDIX: Hessian updating formulae

The updating formulae are used to obtain the approximation to hessian $\mathbf{H}^{k+1}$ at iteration $k+1$ from the hessian at iteration k, $\mathbf{H}^k$. The restriction on the form of the updating formulae are symmetry (hessian must remain a symmetric matrix) and the quasi-Newton condition, defined as

$$\mathbf{H}^{k+1}\delta\mathbf{x}^k = \delta\mathbf{g}^k, \tag{25}$$

where $\delta\mathbf{x}^k = \mathbf{x}^{k+1} - \mathbf{x}^k$, $\delta\mathbf{g}^k = \mathbf{g}^{k+1} - \mathbf{g}^k$ and $\mathbf{g}^k = \nabla V(\mathbf{x}^k)$, which ensures that the quadratic expansion at $\mathbf{x}^{k+1}$ with the updated hessian $\mathbf{H}^{k+1}$ reproduces $\mathbf{g}^k$ at $\mathbf{x}^k$. We have made use of the following updating formulae.

Broyden-Fletcher-Goldfarb-Shanno (BFGS) update[56]

$$\mathbf{H}^{k+1} = \mathbf{H}^k - \frac{\mathbf{H}^k\delta\mathbf{x}^k(\mathbf{H}^k\delta\mathbf{x}^k)^{\mathrm{T}}}{(\delta\mathbf{x}^k)^{\mathrm{T}}\mathbf{H}^k\delta\mathbf{x}^k} + \frac{\delta\mathbf{g}^k(\delta\mathbf{g}^k)^{\mathrm{T}}}{(\delta\mathbf{g}^k)^{\mathrm{T}}\delta\mathbf{x}^k} \tag{26}$$

The BFGS update is positive definite if curvature condition,

$$(\delta\mathbf{x}^k)^{\mathrm{T}}\delta\mathbf{g}^k > 0, \tag{27}$$

is satisfied at every move. In case a step is performed that does not satisfy Eq. (27), a damped BFGS update[48] can be performed,

$$\delta\mathbf{g}^k_{\mathrm{damped}} = \theta^k\delta\mathbf{g}^k + (1-\theta^k)\mathbf{H}^k\delta\mathbf{x}^k, \tag{28}$$

where

$$\boldsymbol{\theta}^k = \begin{cases} 1 & \text{if} \quad (\delta\mathbf{x}^k)^{\mathrm{T}}\delta\mathbf{g}^k \geq 0.2(\delta\mathbf{x}^k)^{\mathrm{T}}\mathbf{H}^k\delta\mathbf{x}^k \\ \frac{0.8(\delta\mathbf{x}^k)^{\mathrm{T}}\mathbf{H}^k\delta\mathbf{x}^k}{(\delta\mathbf{x}^k)^{\mathrm{T}}\mathbf{H}^k\delta\mathbf{x}^k-(\delta\mathbf{x}^k)^{\mathrm{T}}\delta\mathbf{g}^k} & \text{otherwise} \end{cases}, \tag{29}$$

which ensures that the curvature condition is always satisfied, $(\delta\mathbf{x}^k)^{\mathrm{T}}\delta\mathbf{g}^k_{\mathrm{damped}} = 0.2(\delta\mathbf{x}^k)^{\mathrm{T}}\mathbf{H}^k\delta\mathbf{x}^k > 0$, and hessian remains positive-definite. The damped BFGS interpolates between BFGS ($\theta = 1$) and no-update, $\mathbf{H}^{k+1} = \mathbf{H}^k$ ($\theta = 0$).

Symmetric rank 1 (SR1) update[57]

$$\mathbf{H}^{k+1} = \mathbf{H}^k + \frac{(\delta\mathbf{g}^k - \mathbf{H}^k\delta\mathbf{x}^k)(\delta\mathbf{g}^k - \mathbf{H}^k\delta\mathbf{x}^k)^{\mathrm{T}}}{(\delta\mathbf{g}^k - \mathbf{H}^k\delta\mathbf{x}^k)^{\mathrm{T}}\delta\mathbf{x}^k} \tag{30}$$

SR1 update does not guarantee positive-definiteness of hessian. A safeguard needs to be introduced against vanishing denominator in Eq. (30). We skip the update if

$$|(\delta\mathbf{x}^k)^{\mathrm{T}}(\delta\mathbf{g}^k - \mathbf{H}^k\delta\mathbf{x}^k)| \geq r|\delta\mathbf{x}^k||\delta\mathbf{g}^k - \mathbf{H}^k\delta\mathbf{x}^k|, \tag{31}$$

where we take $r = 10^{-8}$.[48]

Powell-symmetric-Broyden (PSB) update[58]

$$\begin{aligned} \mathbf{H}^{k+1} = \mathbf{H}^k \quad &+ \quad \frac{(\delta\mathbf{g}^k - \mathbf{H}^k\delta\mathbf{x}^k)(\mathbf{x}^k)^{\mathrm{T}} + \mathbf{x}^k(\delta\mathbf{g}^k - \mathbf{H}^k\delta\mathbf{x}^k)^{\mathrm{T}}}{(\delta\mathbf{x}^k)^{\mathrm{T}}\delta\mathbf{x}^k} \\ &- \quad \frac{\left[(\delta\mathbf{x}^k)^{\mathrm{T}}(\delta\mathbf{g}^k - \mathbf{H}^k\delta\mathbf{x}^k)\right]\delta\mathbf{x}^k(\delta\mathbf{x}^k)^{\mathrm{T}}}{((\delta\mathbf{x}^k)^{\mathrm{T}}\delta\mathbf{x}^k)^2} \end{aligned} \tag{32}$$

PSB update does not guarantee a positive-definite hessian and fails at different circumstances to SR1 update. It is used as part of Bofill and Ayala-Schlegel update.

Bofill update[59]

$$\mathbf{H}_{\text{Bofill}}^{k+1} = (1-\phi^k)\mathbf{H}_{\text{SR1}}^{k+1} + \phi^k \mathbf{H}_{\text{PSB}}^{k+1}, \tag{33}$$

where

$$\phi^k = 1 - \frac{\left[(\delta\mathbf{x}^k)^{\text{T}}(\delta\mathbf{g}^k - \mathbf{H}^k\delta\mathbf{x}^k)\right]^2}{\left[(\delta\mathbf{x}^k)^{\text{T}}\delta\mathbf{x}^k\right]\left[(\delta\mathbf{g}^k - \mathbf{H}^k\delta\mathbf{x}^k)^{\text{T}}(\delta\mathbf{g}^k - \mathbf{H}^k\delta\mathbf{x}^k)\right]}. \tag{34}$$

Bofill update uses SR1 updating formula unless it is close to its validity limit, Eq. (31), when it uses the PSB updating formula instead.

Ayala-Schlegel update[60]

$$\mathbf{H}_{\text{AS}}^{k+1} = (1-\phi^k)\mathbf{H}_{\text{BFGS}}^{k+1} + \phi^k \mathbf{H}_{\text{PSB}}^{k+1}, \tag{35}$$

where

$$\boldsymbol{\phi}^k = \begin{cases} 1 & \text{if} \quad (\delta\mathbf{g}^k)^{\text{T}}\delta\mathbf{x}^k < 0 \quad \text{or} \quad (\delta\mathbf{x}^k)^{\text{T}}\mathbf{H}^k\delta\mathbf{x}^k < 0 \\ \left(\frac{(\mathbf{g}^{k+1})^{\text{T}}\delta\mathbf{x}^k}{|\mathbf{g}^{k+1}||\delta\mathbf{x}^k|}\right)^2 & \text{otherwise} \end{cases}. \tag{36}$$

The Ayala-Schlegel update uses BFGS updating formula, unless curvature condition in Eq. (27) breaks down or a region of negative curvature along the path is encountered, when it uses the PSB updating formula.

References

(1) Benderskii, V. A.; Makarov, D. E.; Wight, C. A. Chemical Dynamics at Low Temperatures. *Adv. Chem. Phys.* **1994**, 88, 1-358.

(2) DeVault, D. Quantum Mechanical Tunneling in Biological Systems. *Q. Rev. Biophys.* **1980**, 13, 387-564.

(3) Keutsch, F. N.; Saykally, R. J. *Proc. Natl. Acad. Sci. U.S.A.* **2001**, 98, 10533-10540.

(4) Richardson, J. O.; Wales, D. J.; Althorpe, S. C.; McLaughlin, R. P.; Viant, M. R.; Shih, O.;, Saykally, R. J. *J. Phys. Chem. A* **2013**, 117, 6960-6966.

(5) Richardson, J. O.; Pérez, C.; Lobsiger, S.; Reid, A. A.; Temelso, B.; Shields, G. C:; Kisiel, Z.; Wales, D. J.; Pate, B. H.; Althorpe, S. C. *Science* **2016**, 351, 1310-1313.

(6) Hammer, T.; Coutinho-Neto, M. D.; Viel, A; Manthe, U. *J. Chem. Phys.* **2009**, 131, 224109.

(7) Neff, M.; Rauhut G. *Spectrochim. Acta A* **2014**, 119, 100-106.

(8) Marquardt, R.; Sagui, K.; Zheng, J.; Thiel, W.; Luckhaus, D.; Yurchenko, S.; Mariotii, F.; Quack, M. *J. Phys. Chem. A* **2013**, 117, 7502-7522.

(9) Coker, D. F.; Watts, R. O. *J. Phys. Chem.* **1987**, 91, 2513-2518.

(10) Wang, Y.; Braams, B. J.; Bowman, J. M.; Carter, S.; Tew, D. P. *J. Chem. Phys.* **2008**, 128, 224314.

(11) Ceperley, D. M.; Jacucci, G. *Phys. Rev. Lett.* **1987**, 58, 1648-1651.

(12) Mátyus, E.; Wales, D.J.; Althorpe, S. C. *J. Chem. Phys.* **2016**, 144, 114108.

(13) Miller, W. H. *J. Phys. Chem.* **1979**, 83, 960-963.

(14) Makri, M.; Miller, W. H. *J. Chem. Phys.* **1989**, 91, 4026-4036.

(15) Ben-Nun, M.; Martinez, T. J. *J. Phys. Chem. A* **1999**, 103, 6055-6059.

(16) Wales, D. J. *J. Am. Chem. Soc.* **1993**, 115, 11191-11201.

(17) Takahashi, M.; Watanabe, Y.; Taketsugu, T.; Wales, D. J.; *J. Chem. Phys.* **2005**, 123, 044302.

(18) Smedarchina, Z.; Siebrand, W.; Fernandez-Ramos A. *J. Chem. Phys.* **2012**, 137, 224105.

(19) Miller, W. H. *J. Chem. Phys.* **1975**, 62, 1899-1906.

(20) Callan, C. G.; Coleman, S. *Phys. Rev. D* **1977**, 16, 1762-1768.

(21) Vainshtein, A. I.; Zakharov, V. I.; Novikov, V. A.; Shifman, M. A. *Phys. Usp.* **1982**, 25, 195-215.

(22) Mil'nikov, G. V.; Nakamura, H. *J. Chem. Phys.* **2001**, 115, 6881-6897.

(23) Nakamura, H.; Mil'nikov, G. V. *Quantum Mechanical Tunneling in Chemical Physics*; CRC Press Inc, 2013.

(24) Richardson, J. O.; Althorpe, S. C. *J. Chem. Phys.* **2009**, 131, 214106.

(25) Richardson, J. O.; Althorpe, S. C. *J. Chem. Phys.* **2011**, 134, 054109.

(26) Richardson, J. O.; Althorpe, S. C.; Wales, D. J. *J. Chem. Phys.* **2011**, 135, 124109.

(27) Mil'nikov, G.V.; Nakamura, H. *J. Chem. Phys.* **2005**, 122, 124311.

(28) Richardson, J. O. *J. Chem. Phys.* **2016**, 144, 114106.

(29) Richardson, J. O.; Bauer, R.; Thoss, M. *J. Chem. Phys.* **2015**, 143, 134115.

(30) Richardson, J. O. *J. Chem. Phys.* **2015**, 143, 134116.

(31) Cao, J.; Minichino, C.; Voth, G.A. *J. Chem. Phys.* **1995**, 103, 1391-1399.

(32) Richardson, J. O. *Faraday Discuss.* **2016**, 195, 49-67.

(33) McConnell, S.; Löhle, A.; Kästner, J. *J. Chem. Phys.* **2017**, 146, 074105.

(34) Zhang, Y.; Rommel, J.B.; Cvitaš, M. T.; Althorpe, S. C. *J. Chem. Phys.* **2014**, 16, 24292-24300.

(35) McConnell, S.; Kästner, J. *J. Comput. Chem.* **2017**, 38, 2570-2580.

(36) Kryvohuz, M. *J. Chem. Phys.* **2011**, 134, 114103.

(37) Kryvohuz, M. *J. Chem. Phys.* **2012**, 137, 234304.

(38) Rommel, J. B.; Kästner, J. *J. Chem. Phys.* **2011**, 134, 184107.

(39) Einarsdóttir, D. M.; Arnaldsson, A; Óskarsson, F.; Jónsson, H. Path optimization with application to tunneling. *Lecture Notes in Computer Science* **2012**, 7134, 45-55.

(40) Jónsson, H.; Mills, G.; Jacobsen, K. W. Nudged elastic band method for finding minimum energy paths of transitions. *Classical and Quantum Dynamics in Condensed Phase Simulations*; World Scientific, 1998; pp 385-404.

(41) Kawatsu, T.; Miura S. *J. Chem. Phys.* **2014**, 141, 024101.

(42) Smedarchina, Z.; Siebrand, W.; Zgierski, M.Z.; Fernandez-Ramos, A. *J. Chem. Phys.* **1999**, 18, 5-41.

(43) Smedarchina, Z.; Siebrand, W.; Fernandez-Ramos, A. *J. Chem. Phys.* **2013**, 117, 11086-11100.

(44) Cvitaš, M. T.; Althorpe, S. C. *J. Chem. Theory Comput.* **2016**, 141, 024101.

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

(46) E, W.; Ren, W.; Vanden-Eijnden, E. *Phys. Rev. B* **2007**, 126, 164103.

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

(48) Nocedal, J.; Wright, S. J. *Numerical Optimization*, Springer-Verlag New York, 1999.

(49) Zhu, C.; Byrd, R. H.; Lu, P.; Nocedal, J. *ACM. Trans. Math. Softw.* **1997**, 23, 550-560.

(50) Goldstein, H. *Classical Mechanics*, Addison-Wesley, 1950; pp 365-371.

(51) Henkelman, G.; Jónsson, H. *J. Chem. Phys.* **2000**, 113, 9978-9985.

(52) Karney, C. F. F. *J. Mol. Graph. Model.* **2007**, 5, 595-604.

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

(54) Bohner, M. U.; Meisner, J.; Kästner, J. *J. Chem. Theory Comput.* **2013**, 9, 3498-3504.

(55) Trygubenko, S. A.; Wales, D. J. *J. Chem. Phys.* **2004**, 120, 2082-2094.

(56) Broyden, C.G. *IMA J. Appl. Math.* **1970**, 6, 76-90. Fletcher, R. *Comput. J.* **1970**, 13, 317-322. Goldfarb, D. *Math. Comput.* **1970**, 24, 23-26. Shanno, D.F. *Math. Comput.* **1970**, 24, 647-656.

(57) Murtagh, B.A.; Sargent R.W.H. *Comput. J.* **1972**, 13, 185-194.

(58) Powell, M.J.D. *Math. Program.* **1971**, 1, 26-57.

(59) Bofill, J. M. *J. Comput. Chem.* **1994**, 15, 1-11.

(60) Ayala, P.Y.; Schlegel, H.B. *J. Chem. Phys.* **1997**, 107, 375-384.

(61) Wu, H.; Rahman, M.; Wang, J.; Louderaj, U.; Hase, W.L.; Zhuang, Y. *J. Chem. Phys.* **2010**, 133, 074101.

(62) Zhuang, Y.; Siebert, M.R.; Hase, W.L.; Kay, K.G.; Ceotto, M. *J. Chem. Theory Comput.* **2013**, 9, 54-64.

(63) Press, W.H.; Teukolsky, S.A:; Vetterling, W.T.; Flannery, B.P. *Numerical Recipes*, Cambridge University Press, New York, 2007.

(64) Bitzek, E.; Koskinen, P.; Gähler, F.; Moseler, M.; Gumbsch, P. *Phys. Rev. Lett.* **2006**, 97, 170201.

(65) Van Vleck, J.H. *Proc. Natl. Acad. Sci. U.S.A.* **1928**, 14, 178-188.

(66) Manolopoulos, D.E. *J. Chem. Phys.* **1986**, 85, 6425-6429.

(67) Sewell, T. D.; Guo, Y.; Thompson, D. L. *J. Chem. Phys.* **1995**, 103, 8557-8565.

(68) Baba, T.; Tanaka, T.; Morino, I.; Yamada, K. M. T.; Tanaka, K. *J. Chem. Phys.* **1999**, 110, 4131-4133.

(69) Marquardt, R.; Sagui, K.; Zheng, J.; Thiel, W.; Luckhaus, D.; Yurchenko, S.; Mariotti, F.; Quack, M. *J. Phys. Chem. A* **2013**, 117, 7502-7522. 109, 8439-8451.

(70) Urban, Š.; Špirko, V.; Papoušek, D.; Kauppinen, J.; Belov, S.B.; Gershtein, L.I.; Krupnov, A.F. *J. Mol. Spetrosc.* **1981**, 88, 274-292.

(71) Coudert, L.; Valentin, A.; Henry, L. *J. Mol. Spetrosc.* **1986**, 120, 185-204.

(72) Kartha, S.B.; Singh, K.; Job, V.A.; Kartha, V.B. *J. Mol. Spetrosc.* **1988**, 129, 86-98.

(73) Fusina, L.; Di Leonardo, G.; Johns, J.W.C. *J. Mol. Spetrosc.* **1986**, 118, 397-423.

(74) Huang, X.; Carter, S.; Bowman, J. *J. Chem. Phys.* **2003**, 118, 5431-5441.

(75) Liu, D.; Haese, N.N.; Oka, T. *J. Chem. Phys.* **1985**, 82, 5368-5372.

(76) Sears, T.J,; Bunker, P.R.; Davies, P.B.; Johnson, S.A.; Špirko, V. *J. Chem. Phys.* **1985**, 83, 2676-2685.

(77) Kamarchik, E.; Wang, Y.; Bowman, J. *J. Chem. Phys.* **2009**, 113, 7556-7562.

(78) Abramowitz, M.; Stegun I.A. *Handbook of Mathematical Functions*, National Bureau of Standards, 1972; pp 884.

Figure Captions

Figure 1 Number of iterations needed to converge the instanton tunneling path in malonaldehyde to tolerance ϵ (colors denote the tolerance levels as indicated in the legend) as a function of the number of beads. Growing dependencies are obtained using the string method with LBFGS optimizer (denoted as LBFGS), flat dependencies are obtained with the QSM with Bofill updates.

Figure 2 Number of iterations needed to converge instanton path in malonaldehyde ($\epsilon = 10^{-4}$ a.u.) using the QSM as a function of the number of beads. Different colors refer to different hessian updating formulae used, as shown in legend. Dashed black line is obtained using BFGS and tangent updating according to Eq. (15). Missing black circles for certain bead numbers point to the convergence failures.

Figure 3 Number of iterations needed to converge the instanton path in malonaldehyde ($\epsilon = 10^{-4}$ a.u.) using the QSM method with Bofill updates as a function of the number of beads. Initial trust radius Δ_0 is set to the average bead distance at the start of the calculation or fixed at 5 $m_{\mathrm{e}}^{1/2} a_0$, as indicated in the legend. Blue line refers to the calculation in which hessian updates are performed individually (more details in the text). Dotted blue line shows the number of different path configurations at which a hessian update is performed. Inset shows the number of local propagator iterations. Solid line refers to the DMD integrator and dashed line to RK4.

Figure 4 Tolerance (maximum value of the perpendicular action gradient at a bead) as a function of the number of iterations for the instanton calculations in malonaldehyde. Comparison is made between the QSM method with Bofill updates (marked QSM) and the string method with LBFGS optimizer (marked LBFGS) for the number of beads $N = 18$ (red line) and 40 (black line).

Figure 5 Number of iterations needed to converge the instanton path (black line) and MEP (red line) using the string method with LBFGS optimizer and QSM with Bofill updates in malonaldehyde as a function of the number of beads. Convergence criterion ϵ is the average

r.m.s. Euclidean distance in the mass-scaled coordinates to the converged path with the given number of beads, $\epsilon = 0.1\ m_e^{1/2} a_0$.

Figure 6 Average r.m.s. deviation of the converged path with N beads to the fully-converged path ($N \to \infty$) for instanton (black line) and MEP (red line) in malonaldehyde. Inset shows the potential energy along the instanton path and MEP as a function of the mass-scaled distance travelled along the path.

Figure 7 Number of iterations needed to converge instanton tunneling path in ammonia to tolerance ϵ (colors denote tolerance levels as indicated in the legend) as a function of the number of beads. Growing dependencies are obtained using string method with LBFGS optimizer (denoted as LBFGS), flat dependencies are obtained with QSM, where hessians were updated according to Bofill (line) and Ayala-Schlegel (circles). Inset shows the convergence of tunnesling splittings Δ with the number of beads.

Figure 8 Number of iterations needed to converge instanton tunneling path in hydronium to tolerance ϵ (colors denote tolerance levels as indicated in the legend) as a function of the number of beads. Growing dependencies are obtained using string method with LBFGS optimizer (denoted as LBFGS), flat dependencies are obtained with QSM, where hessians were updated according to Bofill (line) and Ayala-Schlegel (circles). Inset shows the convergence of tunnesling splittings Δ with the number of beads.

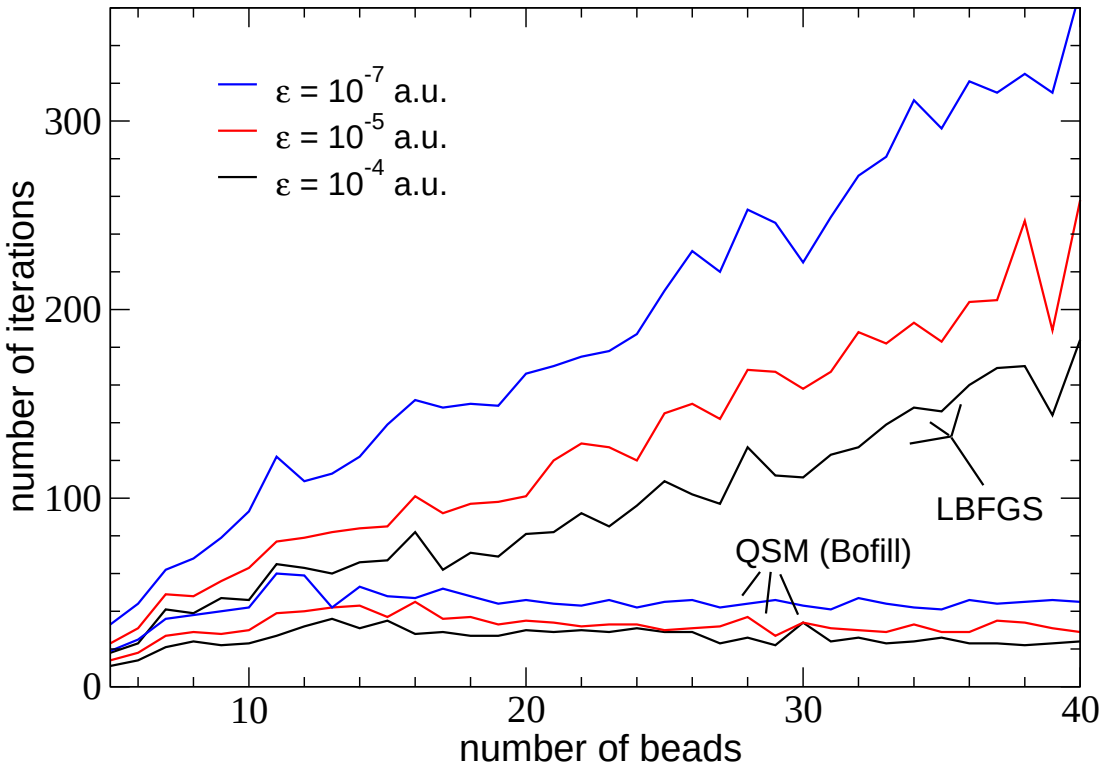
ε = 10^-7 a.u.
ε = 10^-5 a.u.
ε = 10^-4 a.u.
LBFGS
QSM (Bofill)
number of iterations
number of beads
0
100
200
300
10
20
30
40

Figure 1

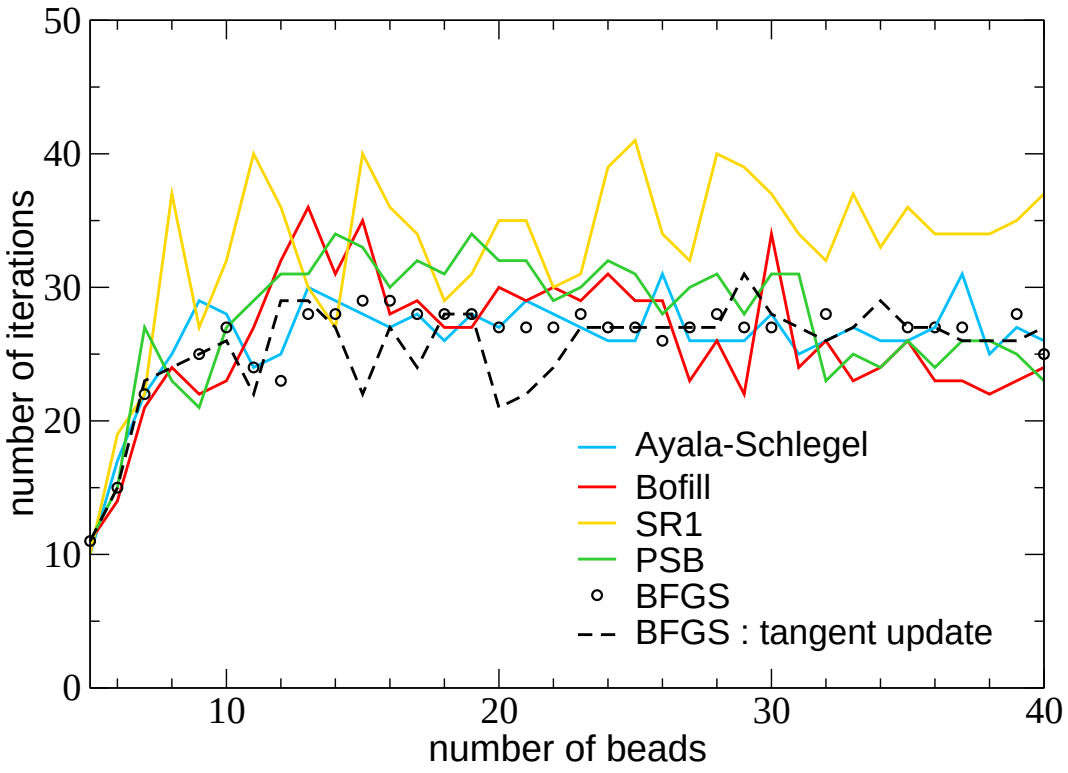
number of iterations
number of beads
0
10
20
30
40
50
10
20
30
40
Ayala-Schlegel
Bofill
SR1
PSB
BFGS
BFGS : tangent update

Figure 2

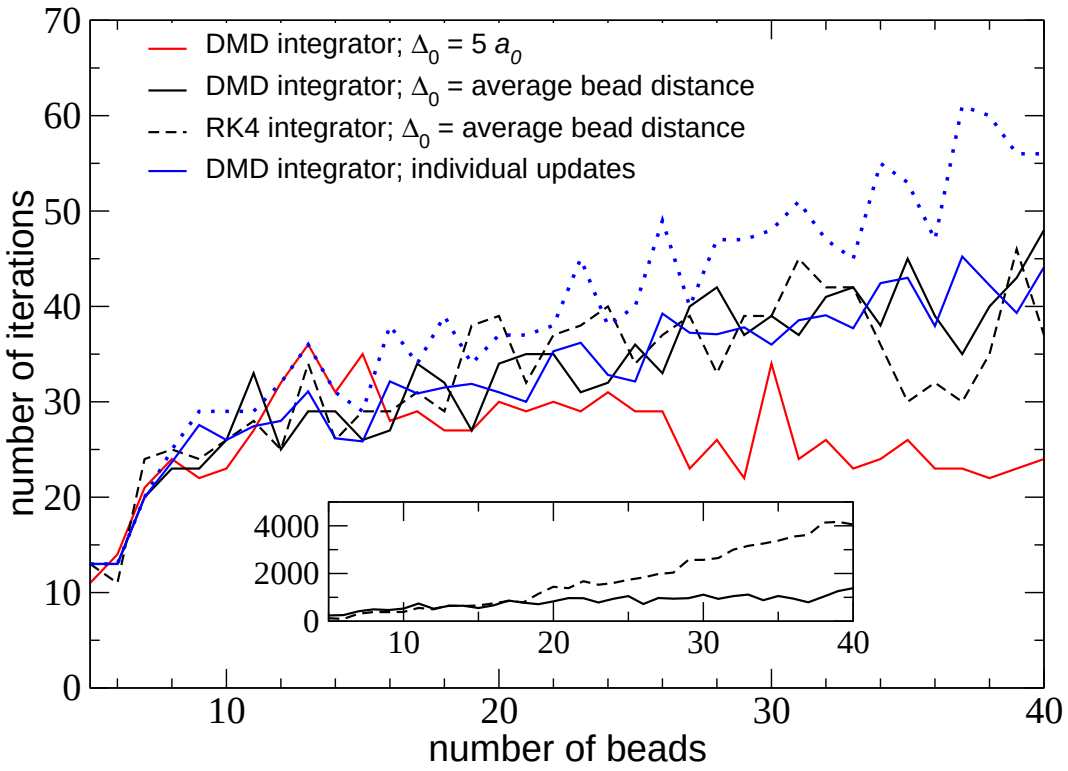
DMD integrator; Δ_0 = 5 a_0
DMD integrator; Δ_0 = average bead distance
RK4 integrator; Δ_0 = average bead distance
DMD integrator; individual updates
number of iterations
number of beads
0
10
20
30
40
50
60
70
4000
2000

Figure 3

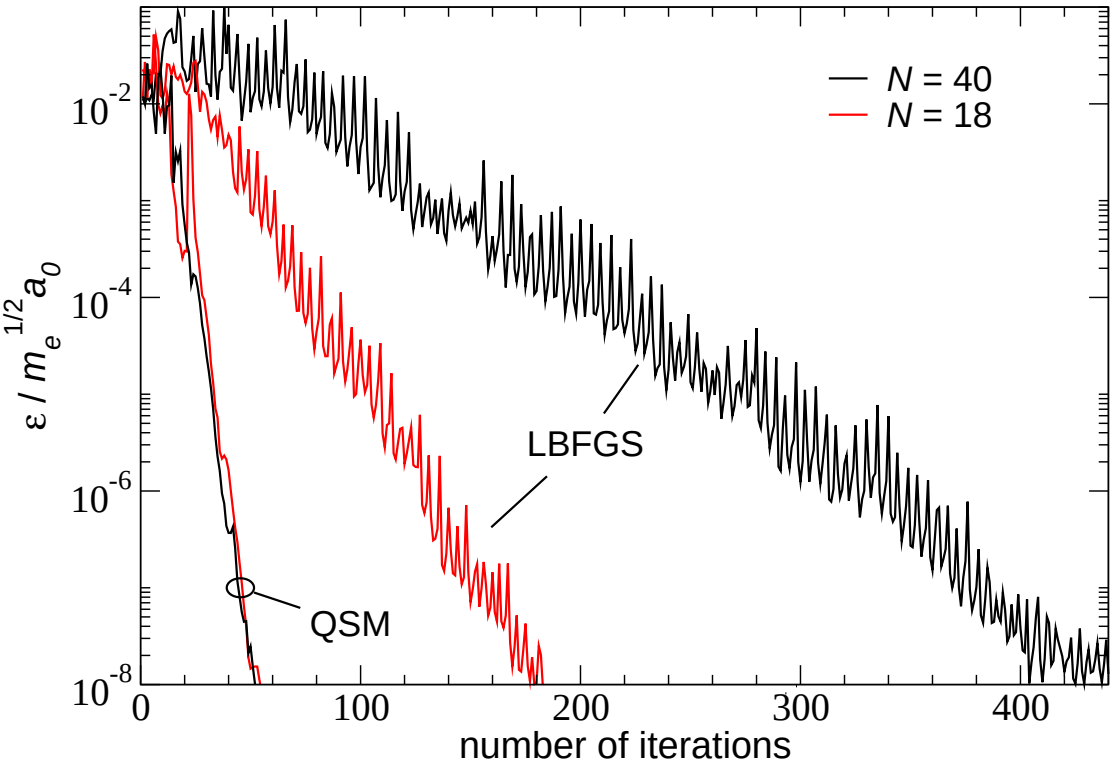
N = 40
N = 18
LBFGS
QSM
ε / m_e^{1/2} a_0
10^-2
10^-4
10^-6
10^-8
0
100
200
300
400
number of iterations

Figure 4

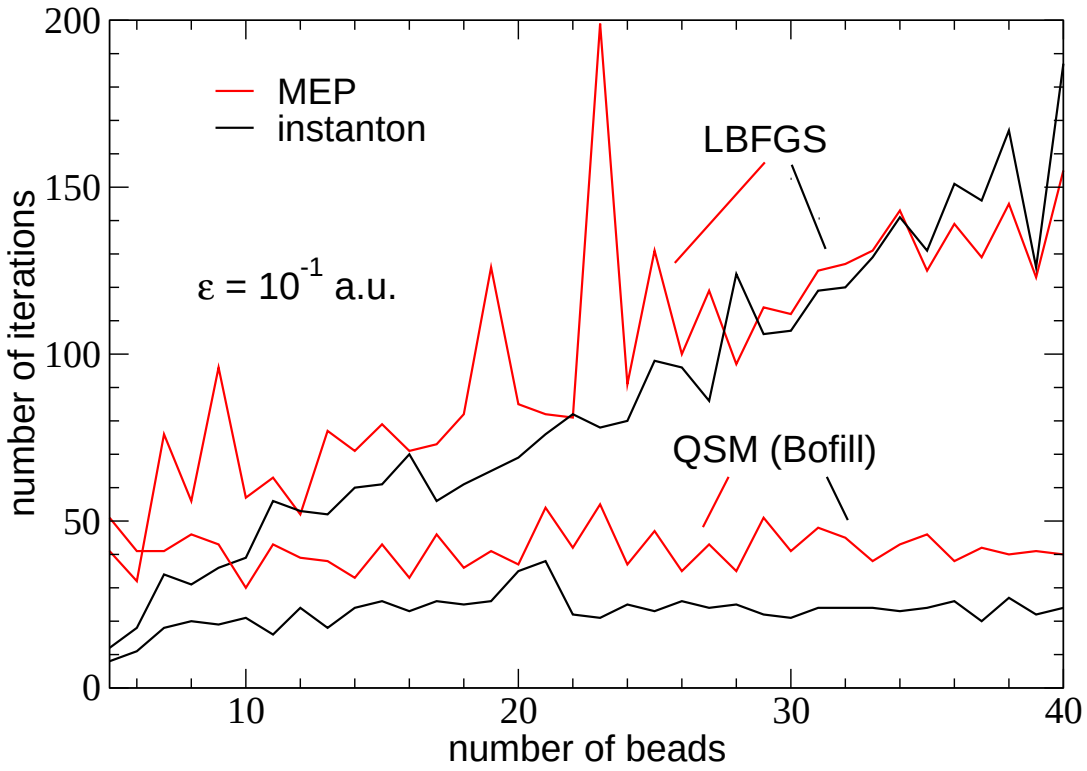
MEP
instanton
LBFGS
QSM (Bofill)
ε = 10^-1 a.u.
number of iterations
number of beads
0
50
100
150
200
10
20
30
40

Figure 5

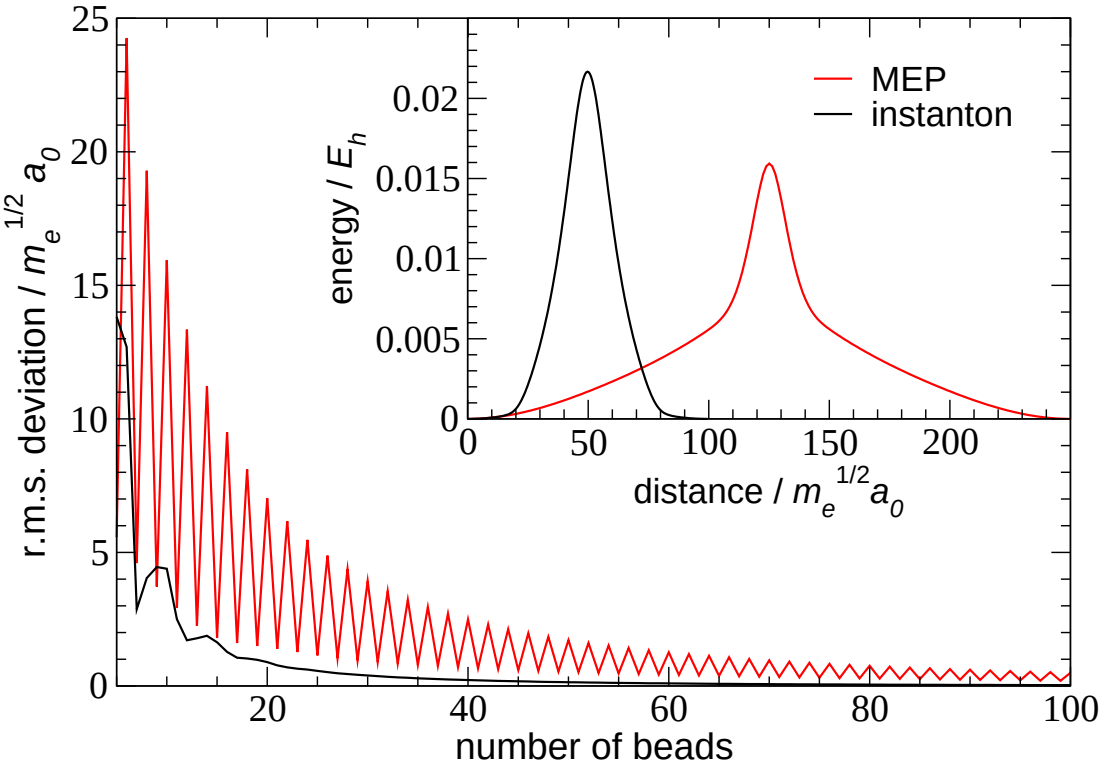
r.m.s. deviation / $m_e^{1/2} a_0$
number of beads
energy / E_h
distance / $m_e^{1/2} a_0$
MEP
instanton

Figure 6

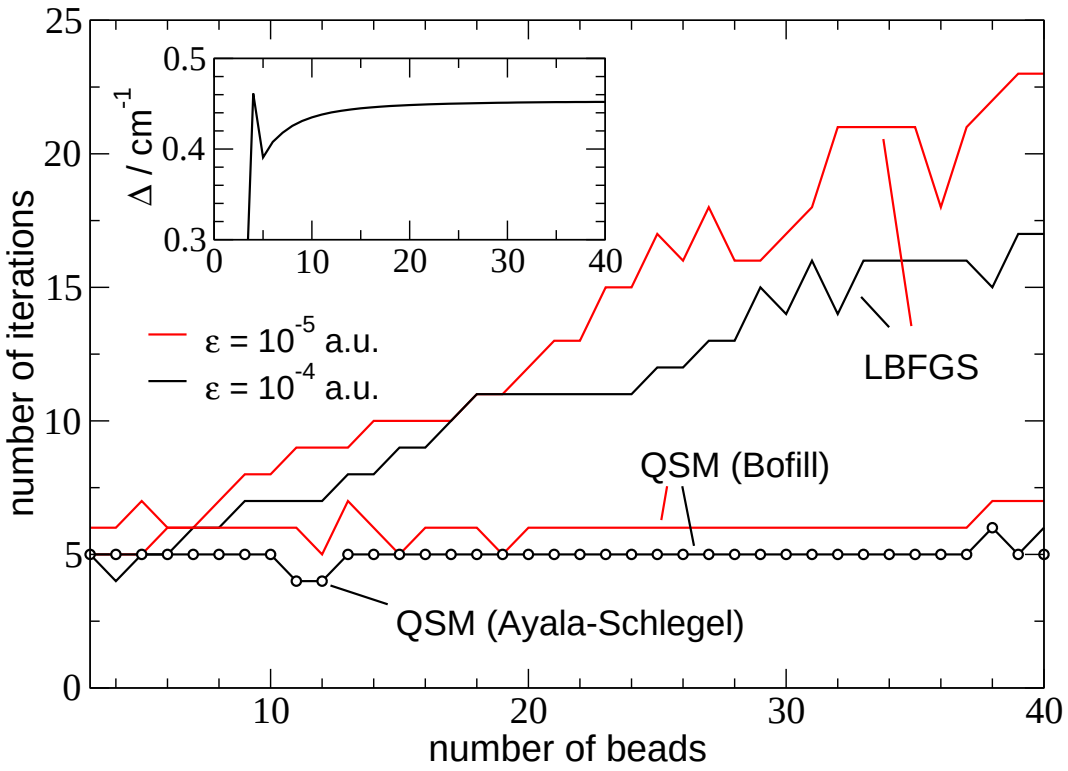
number of iterations
number of beads
Δ / cm-1
ε = 10-5 a.u.
ε = 10-4 a.u.
LBFGS
QSM (Bofill)
QSM (Ayala-Schlegel)

Figure 7

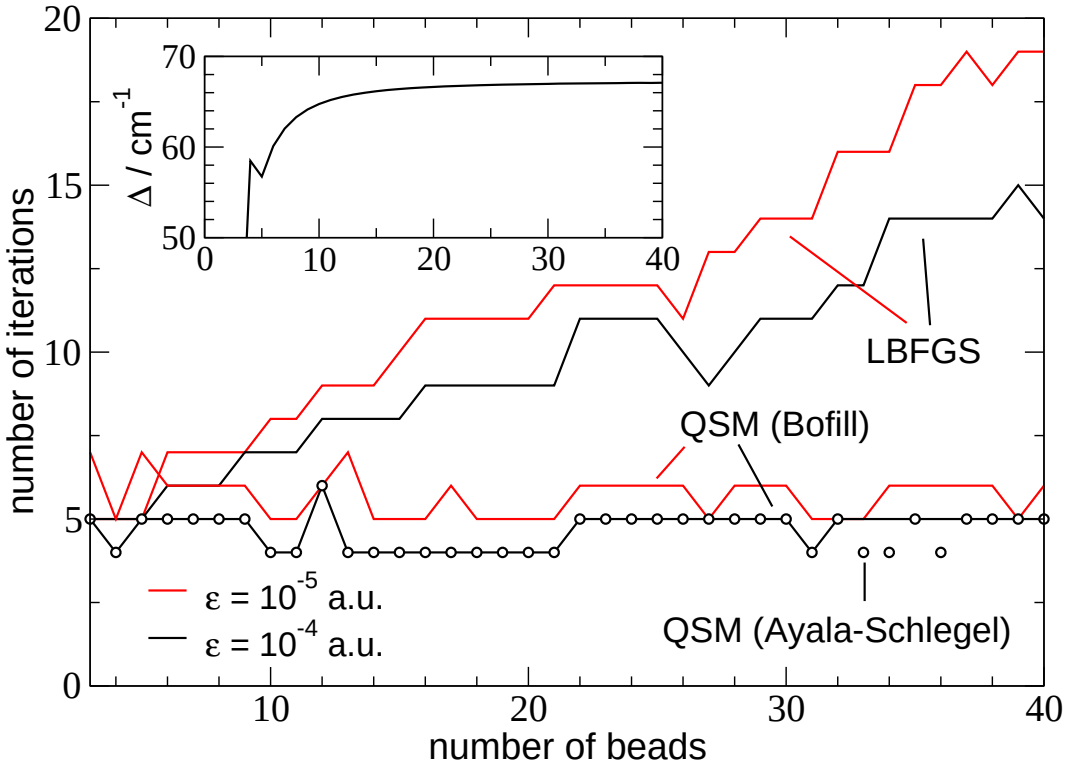
number of iterations
number of beads
Δ / cm-1
LBFGS
QSM (Bofill)
QSM (Ayala-Schlegel)
ε = 10-5 a.u.
ε = 10-4 a.u.

Figure 8