Self-Consistent Variational Transition State Theory with Multidimensional Tunneling Calculations in an Embarrassingly Parallel Scheme

ARTICLE *in* JOURNAL- CHINESE CHEMICAL SOCIETY TAIPEI · APRIL 2007

Impact Factor: 0.65 · DOI: 10.1002/jccs.200700043

2 AUTHORS:

Chien-Pin Chou

National Chiao Tung University

22 PUBLICATIONS **109** CITATIONS

SEE PROFILE

Yao-Yuan Chuang

National University of Kaohsiung

42 PUBLICATIONS **1,329** CITATIONS

SEE PROFILE

Self-Consistent Variational Transition State Theory with Multidimensional Tunneling Calculations in an Embarrassingly Parallel Scheme

Chien-Pin Chou (周建斌) and Yao-Yuan Chuang* (莊曜遠)
Department of Applied Chemistry, National University of Kaohsiung, Kaohsiung 811, Taiwan, R.O.C.

Variational Transition State Theory with Multidimensional Tunneling (VTST/MT) has been successfully used for calculating rate constants of reactions in gas and condensed phases. The current software implementation of VTST/MT is, however, based on the assumption of a fast, serial evaluation of the energetic information of a given molecular structure. We propose a simple and effective parallel method for performing VTST/MT calculations utilizing a cost effective Linux based PC cluster. Five different parallel computing schemes for choosing structures and computing their Hessians along a pre-defined Minimum Energy Path were developed. We found that the Energy Block and Asymmetric Cyclic Execution (EBACE) scheme, which is also most physically intuitive, results in converged rate constants with the least number of Hessians computed. We believe that carrying out the VTST/MT calculation in parallel makes it more attractive for calculating the rate constants of complex chemical systems.

Keywords: Variational Transition State Theory; Parallel computing.

1. INTRODUCTION

Variational Transition State Theory with Multidimensional Tunneling (VTST/MT)[1] has been applied to calculate the rate constants of various types of reactions, for example, gaseous reactions,[2] reactions in condensed phase,[3] metal surface catalytic reactions,[4] and even enzyme catalytic reactions.[5] Due to increases in computer processor speed and the easy availability of large memory systems, applying VTST/MT to calculate rate constants for a complex system is no longer a problem. The software implementation of VTST/MT, specifically, the software package POLYRATE,[6] is a serial program and is based on the assumption of a fast evaluation of the energetic information from a pre-fitted potential energy surface (PES).[7] Truhlar et al. have extended VTST/MT calculations with various single-level[8] and dual-level[9] interpolation methods that reduced computational costs with *ab initio*[10] or semi-empirical[11] methods in the direct dynamics scheme[12] with reasonable computational costs. However, we can speed up the VTST/MT calculation by simply adopting the parallel computing method. Therefore, the goal of this paper is to develop an affordable method which utilizes the advantage of the parallel computing environment while carrying out VTST/MT calculations.

When applying POLYRATE to compute the rate constants, one first needs to optimize the geometries and calculate the vibrational frequencies of the reactants, products, and the transition state of a chemical reaction. The transition state is the maximum point on the minimum energy path (MEP)[13] and it is characterized by an imaginary frequency that corresponds to the unbound motion of the atoms. The MEP can be determined by stepping downhill from the transition state toward the reactant and product regions. After the MEP has been found, free energies along the MEP are calculated by adding the vibrational zero point energy (ZPE) and the contribution due to thermal excitations at various temperatures based on the partition functions using the Harmonic Oscillator and Rigid Rotator (HORR) approximations,[14] or anharmonicity[15] can be included for specific vibrational modes. Due to the re-crossing of the reactive trajectories to the dividing surface, the rate constants calculated using Canonical Variational Theory (CVT)[16] are determined by the free energy difference between the reaction bottleneck, which is located at the maximum of the free energy curve, and the reactants. If the reaction deals with transferring a light particle (such as a hydrogen atom or proton) then quantum mechanical tunneling effects are important and need to be considered. After the shape of the reaction barrier is determined, semi-

* Corresponding author. E-mail: ychuang@nuk.edu.tw

classical approximations such as the Zero-Curvature Tunneling[17] (ZCT), the Small-Curvature Tunneling[18] (SCT), or the Large-Curvature Tunneling[19] (LCT) can be used to estimate the tunneling effect. The differences in these approximations are in the location of the tunneling path used for estimating the tunneling barrier in the calculation of the transmission coefficients. In the ZCT approximation, the MEP is used as the tunneling path; in the SCT approximation, the tunneling path is near the MEP and located on the concave side, and in the LCT approximation, the tunneling paths are the corner-cutting straight paths that connect the reactant and product regions on a PES. The transmission coefficient for the tunneling effect is included as a prefactor to the calculated rate constants in either canonical or microcanonical ensembles. Therefore, to carry out a proper calculation of the rate constants using VTST/MT, one needs to check the convergence of the MEP first and then check the convergence of the numerical integration of the transmission coefficients.

With the fast growth in the computing power of personal computers, scientific computations that used to be carried out on specially designed supercomputer systems such as the Cray have been increasingly performed on Beowulf-type clusters,[20] constructed by networking personal computers via a high-speed interconnect. One way to perform parallel computations on a Beowulf-type cluster system is to use the method of "divide and conquer"[21] where the workload is divided into smaller portions and each portion is distributed amongst separate computing nodes for execution. Therefore, the actual wall clock time used is expected to be less, but the total computing time (which is the CPU time required to finish the whole workload) will be the same and may even increase due to communication bottlenecks and other inefficiencies.

In order to perform VTST/MT calculations in parallel on a cluster we developed the Self-Consistent Variational Transition State Theory with Multidimensional Tunneling and Parallel Hessian Computation scheme by constructing a wrapper software program that allows us to obtain the converged rate constants with VTST/MT. By constructing an automatic iterative process that uses a parallel computing environment, we believe it will be possible to apply the VTST/MT to calculate the rate constants of complex chemical reactions. The methodology and implementation of the parallel Hessian computing scheme is presented in Section 2. In Section 3, calculated results of two reactions with different shapes of the reaction barriers are discussed. Finally, the concluding remarks are presented in Section 4.

2. METHOD AND IMPLEMENTATION

The basic design of the POLYRATE software package is summarized in Fig. 1a. The energetic information (such as energy, gradient, and Hessians) of the reactants and products are obtained by using a pre-fitted potential energy surface PES, or by *ab initio* or semi-empirical methods. The transition state of the reaction (or the saddle point) which is the maximum point on the minimum energy path (MEP) can be obtained through a geometry optimization procedure. Then the MEP is followed by using either a gradient based integrator (such as the Euler Steepest Descent[23] method) or a Hessian based integrator (such as the Page-McIver[24] method) or other improved methods (such as the Variational Reaction Path[25] method) started downhill from the transition state. The benefit of applying a Hessian based integrator is that a much larger step size can be applied to

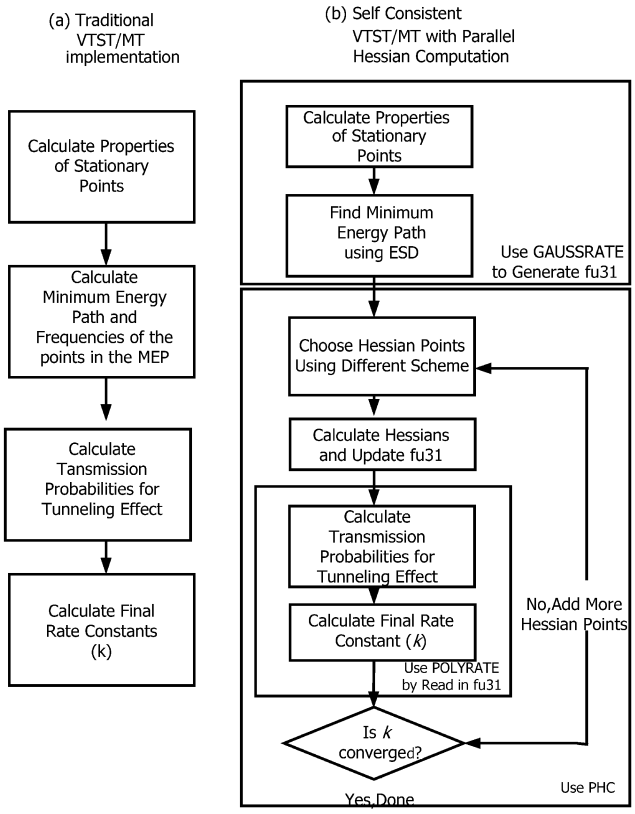

Fig. 1. Flow charts of the software implementation of: a) traditional VTST/MT, and b) self consistent-VTST/MT with parallel hessian computation.

solve the following ordinary differential equation

$$\frac{d\mathbf{x}(s)}{ds} = -\frac{\mathbf{G}(\mathbf{x})}{|\mathbf{G}(\mathbf{x})|} \quad (1)$$

where the reaction coordinate s is the arc length from the saddle point, $\mathbf{x}$ denotes points on the reaction path and $\mathbf{G}(\mathbf{x})$ is the gradient at point $\mathbf{x}$. However, the Hessian based reaction path integrator is a serial method and parallelizing it is not trivial. For simplicity and due to the fact that the Hessian calculation takes much longer than the Gradient calculation in complex *ab initio* methods, we decided to perform parallel calculations using the Euler Steepest Descent algorithm where the reaction path is followed along the gradient direction:

$$\mathbf{x}_{n+1}^{(0)} = \mathbf{x}_n - \delta s\hat{\mathbf{G}}(\mathbf{x}_n) \quad (2)$$

$$\hat{\mathbf{G}}(\mathbf{x}_n) = \frac{\mathbf{G}(\mathbf{x}_n)}{|\mathbf{G}(\mathbf{x}_n)|} \quad (3)$$

where $\mathbf{G}(\mathbf{x}_n)$ is the gradient at point $\mathbf{x}_n$ and δs is the size of integration step.

After the MEP is followed using the ESD method with a small steps (which in the POLYRATE is denoted as the gradient grid), normal mode analysis is carried out on a coarse grid (which is called the Hessian grid in POLYRATE and is controlled by the keyword option INH). The vibrational frequencies thus obtained are used to calculate the vibrational partition functions and the eigenvectors are used to estimate the curvature for the Small Curvature Tunneling (SCT) approximation or pointing out the direction of the Large Curvature Tunneling (LCT) Paths. The calculation of the LCT paths is another area that can be parallelized, but the current version of the software package does not implement it.

Fig. 1b shows the basic design of software package PHC. Using the GAUSSRATE[26] package which is an interface program that links Gaussian 03[27] and POLYRATE together for reaction rate calculation using VTST/MT, we first write out the MEP in a text file format (called *fu31*) by running the GAUSSRATE program and stopping right after the MEP is determined. Then, the PHC program performs the following tasks: 1) reads in the information from the *fu31* generated previously and selects the points that are needed to perform the Hessian computation based on the different selection schemes described in the next section, 2) divides the Hessian computation tasks amongst the computing nodes, where each node then performs one Hessian computation by using an *ab initio* software package such as Gaussian 03, 3) sends the calculated Hessians back to the master node which updates the file *fu31* by adding the computed Hessians, and 4) uses POLYRATE with unit31 input option for the mapped interpolated (IVTST-M) calculation. These four steps are repeated until the calculated rate constants converge. Convergence is achieved when a) the Unsigned Difference in the Log of CVT rate constant (UDLK) at the lowest temperature for one iteration differs from the same quantity calculated for the previous iteration by less than 0.01, i.e.

$$(\mathrm{UDLK} = |\log k_i^{CVT} - \log k_{i-1}^{CVT}|) < 0.01 \quad (4)$$

where i is the iteration number and b) when the Unsigned Difference in the Log of Transmission coefficient at the lowest temperature (UDLT) for one iteration differs from the same quantity calculated for the previous iteration by less than 0.01, i.e.

$$(\mathrm{UDLT} = |\log \kappa_i^j - \log \kappa_{i-1}^j|) < 0.01 \quad (5)$$

where i is the counter of the iterations, and j = (ZCT, SCT). The IVTST-M method which applies the Spline under tension[28] fit to the calculated energies, gradients, frequencies, and curvatures for the points along the MEP is used to generate a smooth free energy profile for the rate constant calculations.

2.B. Parallel Hessian Computation Schemes

Figs. 2, 3, 4, 5, and 6 demonstrate five different schemes developed to choose and compute the Hessians along a given MEP viz. the a) Asymmetric Cyclic Execution (ACE), b) Asymmetric Block Partitioning (ABP), c) Symmetric Cyclic Execution (SCE), d) Symmetric Block Partitioning (SBP), and e) Energy Block and Asymmetric Cyclic Execution (EBACE) schemes. For all schemes, we assume p nodes in the compute cluster and that the reaction path is asymmetric with a total of N Hessians to be computed along the MEP. For comparison, the reference values are obtained based on the calculation with the full N Hessians calculated. The number of Hessians on the reactant side is N_R, that on the product side N_P, with $N = N_R + N_P$. We also assume the reactions are in the exoergic direction. The squares within Figs. 2-6 are used to indicate the first two iterations of the calculations for demonstrating the order of computing of the Hessians along the MEP.

2.B.1 Asymmetric Cyclic Execution (ACE)

In the Asymmetric Cyclic Execution scheme, Hessians were computed starting from the saddle point towards the reactants and products. Only one Hessian computation is performed by each node in every iteration for the best performance of the cluster. Fig. 2 shows a schematic of the Asymmetric Cyclic Execution scheme with the 6 processors identified as A, B, C, D, E, and F. The solid squares indicate calculations performed in the first iteration and the solid circles for the second iteration for demonstrating the

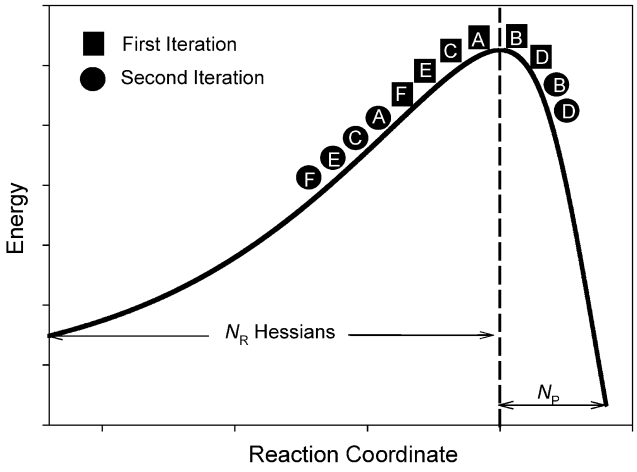

Fig. 2. The asymmetric cyclic execution (ACE) parallel hessian computing scheme with six nodes. The solid squares represent the calculations performed in the first iteration and the solid circles represent the calculations performed in the second iteration. There are N_R hessians on the reactant side and N_P hessians on the product side of the MEP.

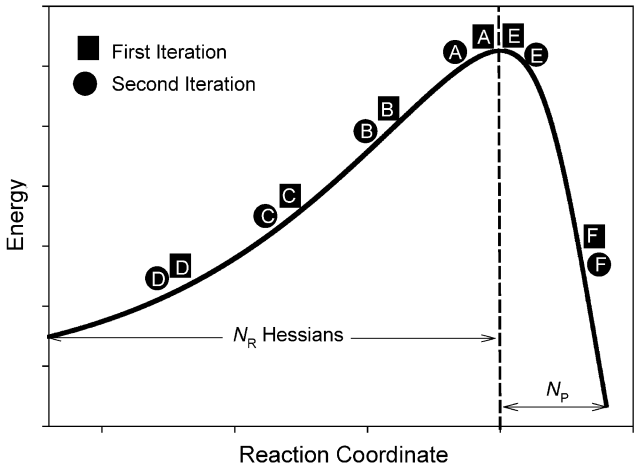

Fig. 3. The asymmetric block partitioning (SBP) parallel hessian computing scheme with six computing nodes; the solid squares represent the calculations performed in the first iteration and the solid circles represent the calculations performed in the second iteration. There are N_R hessians on the reactant side and N_P hessians on the product side of the MEP.

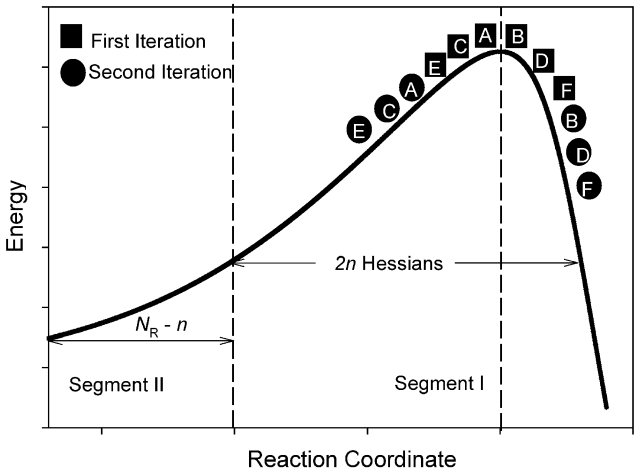

Fig. 4. The symmetric cyclic execution (SCE) parallel hessian computing scheme with six computing nodes; the solid squares represent the calculations performed in the first iteration and the solid circles represent the calculations performed in the second iteration. There are N_R hessians on the reactant side and N_P hessians on the product side of the MEP.

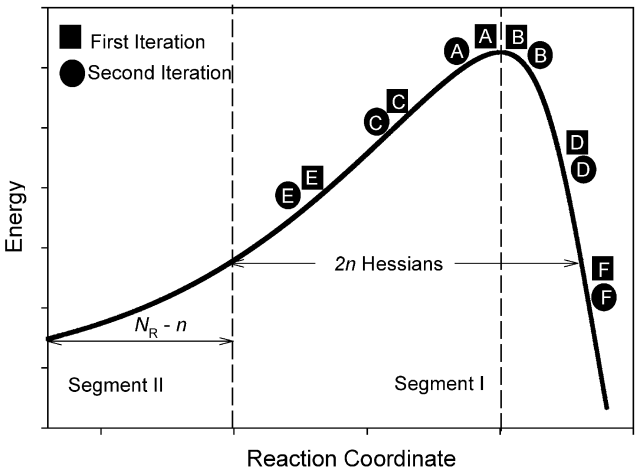

Fig. 5. The symmetric block partitioning (SBP) parallel hessian computing scheme with six computing nodes; the solid squares represent the calculations performed in the first iteration and the solid circles represent the calculations performed in the second iteration. There are N_R hessians on the reactant side and N_P hessians on the product side of the MEP.

choice of Hesians within different schemes. Since the MEP is asymmetric, the number of nodes assigned to calculate the Hessians is proportional to the ratio of the number of Hessians in the reactant and product sides, that is, for the reactant side, we use $\left(\left\lfloor \frac{N_R}{N} \right\rfloor p\right)$ computing nodes and $p\left(1-\left\lfloor \frac{N_R}{N} \right\rfloor\right)$ nodes for the product side. In Fig. 2, there are 4 nodes used to calculate the Hessians on the reactant side of the MEP and 2 nodes for the product side, assuming $N_R = 2N_P$. The reason we use a different number of nodes to calculate the Hessians in the reactant and product sides is because we try to follow the reaction path by keeping the energies of the endpoints the same. We believe this "balanced" distribution in energy will allow us to utilize computing resources efficiently, thus reaching convergence in the calculated rate constants faster. The benefit of applying the ACE scheme is that the points nearby the barrier top are calculated first. Therefore, the rate constant computed by the Canonical Variational Theory (CVT) converges within a few iterations due to the smooth free energy profile nearby the barrier top.

2.B.2 Asymmetric Block Partitioning (ABP)

Another approach to dividing up the calculation is the Block Partitioning[21] scheme where we divide the N Hessians into p blocks evenly and assign each block to one computing node. In each iteration, only p Hessians are computed, thus requiring a maximum of N/p iterations to compute all N Hessians in the MEP. For each block, the point with smallest absolute value of the reaction coordinate ($|s|$) is computed first; if both the reactant and product regions are included in the same block, the reactant Hessians are computed before the product ones. The benefit of using Asymmetric Block Partitioning is that the shape of the MEP can be described with only the first few computed Hessians, and we expect the transmission coefficients to converge within the first few iterations. Fig. 3 shows the ABP scheme with 6 computing nodes denoted as A, B, C, D, E, and F, where 4 nodes (A,B,C,D) are used for the reactant side and 2 nodes for the product side, assuming $N_R = 2N_P$. The solid squares indicate calculations done in the first iteration and the solid circles the second iteration.

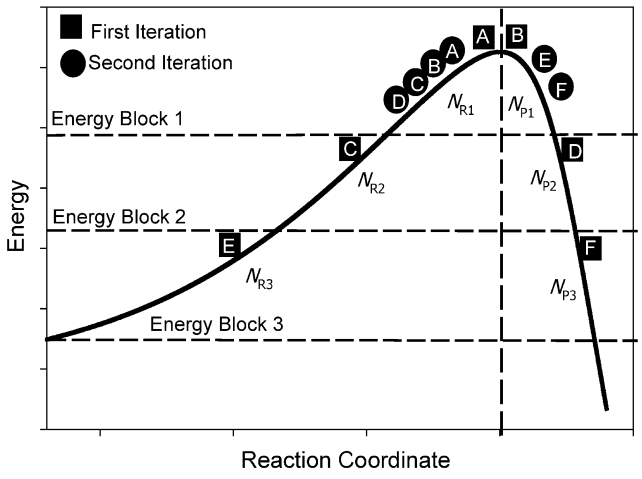

Fig. 6. The energy block asymmetric cyclic execution (EBACE) parallel hessian computing scheme with six computing nodes; the solid squares represent the calculations performed in the first iteration and the solid circles represent the calculations performed in the second iteration. There are N_R hessians on the reactant side and N_P hessians on the product side of the MEP.

2.B.3 Symmetric Cyclic Execution (SCE)

Since the MEP is asymmetric and only the part with energies above zero (with respect to the sum of the energies of the reactants) is important for estimating the tunneling effect, we can divide the MEP into two segments to achieve the convergence of rate constants faster, assuming the reaction is very exoergic and there are more Hessians on the product side. Therefore, in the SCE scheme, we divide the MEP into two segments by looking for the minimum of the number of Hessians in the reactant side (N_R) and product side (N_P).

$$n = \min(N_R, N_P) \tag{6}$$

Therefore, segment I contains total $2n$ Hessians and the segment II contains total N-$2n$ Hessians. The Hessians in segment I are computed using a method similar to one in the ACE scheme where the points with the smallest value of |s| are computed first. Fig. 4 shows the SCE scheme with 6 computing nodes denoted as A, B, C, D, E, and F and where, as before, the solid squares designate calculations in first iteration and solid circles the second one. Since segment I is symmetric with respect to the number of Hessian computations in the reactant and product branches, each branch of the MEP is calculated with 3 computing nodes. When all Hessians in segment I are finished, the Hessians in segment II are then computed using all 6 computing nodes.

2.B.4 Symmetric Block Partitioning (SBP)

Similar to the SCE scheme, the MEP can be divided

into two segments with respect to the number of Hessians where segment I contains $2n$ Hessians and the segment II contains $N - 2n$ Hessians, and where n is determined by Equation 6. If $2n$ is not exactly divisible by p, then Hessian points from the segment II are added to complete the last iteration. The Hessians in segment I are then divided into p blocks, and each node computes $2n/p$ Hessians in the order of increasing $|s|$. When all Hessians in segment I are computed, the $N - 2n$ Hessians in segment II are divided into p blocks and calculated similarly. Fig. 5 shows the SBP scheme with the 6 nodes denoted as A, B, C, D, E, and F where segment I is divided into 6 blocks with 3 computing nodes used for each side of the MEP. As before, the solid squares represent the calculations done in the first iteration and the solid circles the second iteration.

2.B.5 Energy Block and Asymmetric Cyclic Execution (EBACE)

Although partitioning blocks according to the number of Hessians along the MEP can be conveniently implemented, this kind of a partitioning scheme can result in an inefficient computation. For example, if one includes many structures on the MEP with the energies lower than the reactants, the ABP scheme will not be inefficient due to the computing cycles spent calculating the Hessians of these 'insignificant' points. The SBP scheme was the first attempt to fix this problem, but it only works if $N_R < N_P$. For the case $N_R > N_P$ the SBP is not able to solve the problem since Hessians in segment II which are located on the reactant side are also important for the tunneling calculations. A more appropriate solution would be to partition the Hessians according to the reaction barrier height. Fig. 6 shows the EBACE scheme with the 6 nodes denoted as A, B, C, D, E, and F. First, we partition the reaction barrier into $p/2$ energy blocks as separated by the horizontal lines and then the Hessians within each energy block are computed using the ACE scheme in the ascending order of |s| as indicated by the solid circles. Assuming N_{R1} Hessians on the reactant side and N_{P1} Hessians on the product side of the energy block 1, we divided the number of processors proportional to the number of Hessians, that is, $\left(\left\lfloor \frac{N_{R1}}{N_{R1}+N_{P1}} \right\rfloor p\right)$ processors for the reactant side and $p\left(1-\left\lfloor \frac{N_{R1}}{N_{R1}+N_{P1}} \right\rfloor\right)$ processors for the product side. After $N_{R1} + N_{P1}$ Hessians are computed, p processors are used to compute the Hessians in energy block 2 in a similar manner until convergence of the calculated rate constants is reached. The advantage of the EBACE scheme is that the shape of the MEP is determined by the first p calculations during the first iteration and the Hessians are computed in a top-down manner and with the number of nodes proportional to the ratio of the number of Hessian points in the reactant and product sides. The EBACE is similar to the manual procedure to check for numerical convergence in VTST/MT calculations, where the points near the barrier top are calculated first and we then extend the MEP until the calculated rate constants are converged. We denote the partition into $p/2$ energy blocks in the first iteration as EBACE-1 and the further partitioning in the first two iterations to p energy block as the EBACE-2 in the next section. In theory, the EBACE-2 method should provide a better shape of the reaction barrier due the finer grid points. Fig. 6 shows the MEP divided into three energy blocks where each block is further divided into 2 segments according to the sign of the reaction coordinate. One Hessian is calculated in each segment during the first iteration to "map" out the shape of the reaction barrier (as indicated by the solid squares). The Hessians in the first energy block are then calculated beginning in the second iteration as indicated by the solid circle.

3. RESULT AND DISCUSSION

The reaction rate constants of the following reactions

$$OH + H_2 \rightarrow H_2O + H \quad \text{(R1)}$$

$$CH_3 + H_2 \rightarrow CH_4 + H \quad \text{(R2)}$$

were calculated at the Canonical Variational Theory (CVT) level, with Zero Curvature Tunneling (CVT/ZCT), and with Small Curvature Tunneling (CVT/SCT) approximations. We used the GAUSSRATE 9.3 to calculate the rate constants of R1 at MP2 (Frozen Core)/6-31+G(d,p) level and HF/6-31+G(d,p) for R2. The scaling mass is set equal to 1.0 amu and the reaction path is followed using the Euler Steepest Descent (ESD) method with a gradient step size of 0.001 bohr for both reactions. The symmetry number of the forward reaction is 2 for R1 and 4 for R2. All calculations are carried out with the MEP being followed from -1.7 to 0.4 bohr for R1 and -1.0 to 1.2 bohr for R2 and the ratio of the Gradient to Hessian (INH) is set to 5 for both reactions. Thus, we have a total of 420 Hessians along the MEP in R1, and 440 Hessians in R2. The classical barrier height is 10.51

kcal/mol and the energy of the reaction is -20.48 kcal/mol for R1 and the barrier height of R2 is 22.23 kcal/mol with -1.46 kcal/mol for the energy of the reaction. The level of electronic structure theory used was chosen to emphasize the parallel Hessian computation schemes for different reaction barrier shapes rather than accuracy; therefore, the shape of the reaction barrier of R1 is low and asymmetric, whereas the barrier of R2 is tall and near symmetric. All calculations were carried out on a cluster of 6 dual-processor 2.0 GHz AMD Opteron computers connected using Gigabit Ethernet and running the SUSE 9.1 Linux operating system. Although there are a total of 12 processors in the cluster, we only used one processor per each machine but each Hessian computation is performed with two processes within each symmetric memory machine. By combining the strategies of message-passing between the computing nodes and utilizing the shared memory between the processors, we obtain the best performance of the whole cluster.

Table 1 gives the number of the iterations required to obtain the converged rate constants for the different parallel Hessian computation schemes for R1. As expected, when using the ACE scheme, only 8 iterations were required to reach the convergence of the CVT rate constants but 34 iterations to converge the rate constants at the CVT/ZCT level and 39 iterations at the CVT/SCT level which corresponds to computing 56% of the Hessians in the MEP. For the ABP scheme, since the shape of the MEP is being mapped out first, the rate constants using CVT, CVT/ZCT, and CVT/SCT were converged at 21 iterations which correspond to about 30% of the Hessians in the MEP being calculated. No significant improvements were observed with either the SCE or SBP schemes since they required calculating more than 40% of the Hessians along the MEP. Since we only consider geometries in the reaction barrier for the calculations with R1 and R2, we expect a similar performance of the SCE and SBP to the ACE and ABP. However, if one includes more points in the MEP, the results might be different. For a more physically intuitive approach, consider the EBACE-1 scheme where the reaction barrier is divided into $p/2$ energy blocks in the first iteration. The rate constants were converged at 18 iterations with only 26% of the Hessians in the MEP being computed. The EBACE scheme, therefore, not only gives us a reasonable physical interpretation but is also very efficient. Similarly, the EBACE-2 scheme, where the reaction barrier is divided into p energy blocks in the first two iterations and the Hessians in each energy block are computed as in the ACE scheme, gives results similar to EBACE-1. Although we can generalize to an EBACE-n scheme where n denotes the number of the iterations used for dividing energy blocks, we believe from the results of EBACE-1 and EBACE-2 schemes, no further division of the barrier height is necessary.

Table 1. Number of iterations required to converge rate constant of R1 with different parallel hessian computation schemes[a]

	CVT	CVT/ZCT	CVT/SCT[b]
ACE	8	34	39 (56%)
ABP	18	21	59 (30%)
SCE	6	44	46 (66%)
SBP	14	16	29 (41%)
EBACE-1	6	18	18 (26%)
EBACE-2	7	17	17 (24%)

[a] The cluster has 6 computing nodes ($p = 6$), total number of the Hessians computed is equal to the number of iterations × 6.
[b] Number in parenthesis denotes the percentage of the Hessians in the MEP computed to reach the convergence.

Table 2 gives the number of iterations required for the different Hessian computing schemes for reaction R2. As expected, it only takes 4 iterations for the convergence of the CVT rate constant in the ACE scheme, but takes 31 iterations to converge the rates at the CVT/ZCT and 38 iterations at the CVT/SCT, which corresponds to computing 52% of the Hessians to the converged calculation. For the ABP scheme, 14 iterations were required to reach the convergence of the CVT rate constants and 54 iterations to converge the rate constants at the CVT/SCT which corresponds to computing 74% Hessians to the converged calcu-

Table 2. Number of iterations required to converge rate constant of R2 with different parallel hessian computation schemes[a]

	CVT	CVT/ZCT	CVT/SCT[b]
ACE	4	31	38 (53%)
ABP	10	14	54 (74%)
SCE	4	31	37 (52%)
SBP	5	10	56 (76%)
EBACE-1	4	11	27 (37%)
EBACE-2	6	8	27 (37%)

[a] The cluster has 6 computing nodes ($p = 6$), total number of the Hessians computed is equal to the number of iterations × 6.
[b] Number in parenthesis denotes the percentage of the Hessians in the MEP computed to reach the convergence.

lation. Again, the results of the SCE and SBP schemes are similar to the ACE and ABP schemes due to the near symmetric feature of the reaction barrier. Again, the EBACE-1 and EBACE-2 schemes are the most efficient, with only 37% of the Hessians in the MEP being required to be computed for the convergence of the rate constants.

To quantify the deviation of the calculated rate constants to the converged ones, the mean unsigned percentage deviation, abbreviated MUPD for each calculation is defined by

$$\mathrm{MUPD} = \frac{100}{3 * n_T} \sum_{j=1}^{3} \sum_{i=1}^{n_T} \frac{\left| k^j(T_i) - k^j_{\mathrm{ref}}(T_i) \right|}{k^j_{\mathrm{ref}}(T_i)} \qquad (7)$$

where n_T is the number of temperatures, T_i is one of the five temperatures for R1 and one of the seven temperatures for R2 and j is one of the three types of dynamics calculations (CVT, CVT/ZCT, and CVT/SCT). Tables 3 and 4 give the MUPD of the calculated rate constants with different converged threshold values for reactions R1 and R2, respectively. When the threshold value for convergence check is set to 0.01, which corresponds to a 2% error in the calculated rate constant, we obtain at most 5% deviation in the calculated rate constants to the reference values. If a larger threshold value is applied, then the calculations might tend to converge to the wrong rate constants and cause larger deviations to the reference values. The logarithm of the calculated rate constants in ACE, SBP and EBACE-1 at the lowest temperature with CVT/SCT approximation during the iterative process for R1 and R2 are given in Figs. 7 and 8. The calculated rate constants are converged in a much smoother manner for R2 than R1; also as observed from these figures, the EBACE-1 scheme outperforms both the ACE and ABP schemes.

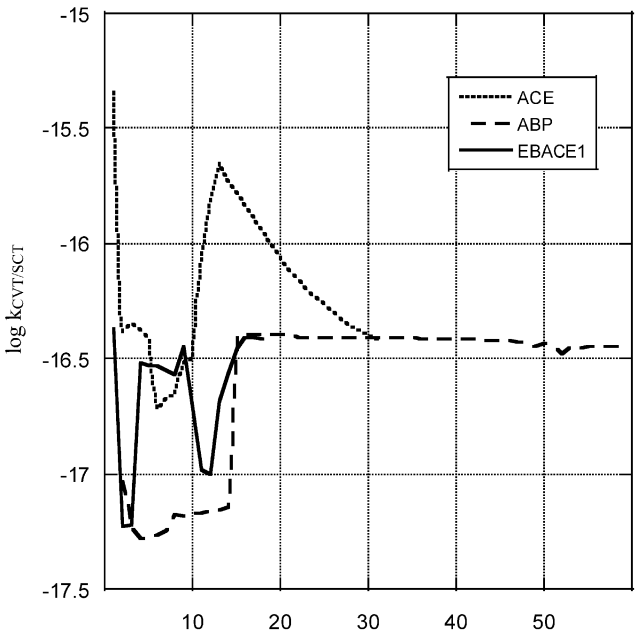

Fig. 7. Logarithm of the calculated CVT/SCT rate constants at 250 K for R1 with different hessian computing schemes. The dotted line represents asymmetric cyclic execution scheme (ACE), the dashed line represents asymmetric block partitioning scheme (ABP), and the solid line represents energy block and asymmetric block partitioning scheme (EBACE-1).

Table 3. Mean unsigned percentage difference to the reference rate constants R1 with different threshold values for convergence[a]

	Threshold Value			
	0.01	0.02	0.04	0.05
ACE	4 (39)	5 (32)	15 (22)	42 (16)
ABP	5 (59)	12 (24)	48 (6)	50 (4)
SCE	4 (46)	5 (40)	22 (11)	23 (7)
SBP	4 (29)	41 (7)	43 (6)	43 (6)
EBACE-1	5 (18)	5 (18)	22 (6)	22 (6)
EBACE-2	0 (17)	6 (10)	22 (7)	22 (7)

[a] Number in parenthesis denotes the number of iterations to reach the convergence.

Table 4. Mean unsigned percentage difference to the reference rate constants R2 with different threshold values for convergence[a]

	Threshold Value			
	0.01	0.02	0.04	0.05
ACE	1 (38)	5 (27)	14 (19)	14 (16)
ABP	2 (54)	13 (19)	18 (12)	21 (7)
SCE	1 (37)	14 (19)	14 (18)	14 (15)
SBP	1 (56)	18 (10)	24 (7)	24 (10)
EBACE-1	1 (27)	1 (21)	8 (12)	18 (21)
EBACE-2	0 (27)	5 (10)	9 (7)	10 (10)

[a] Number in parenthesis denotes the number of iterations to reach the convergence.

4. CONCLUSION

We have developed a software program PHC for iteratively carrying out Variational Transition State Theory with Multidimensional Tunneling (VTST/MT) calcula-

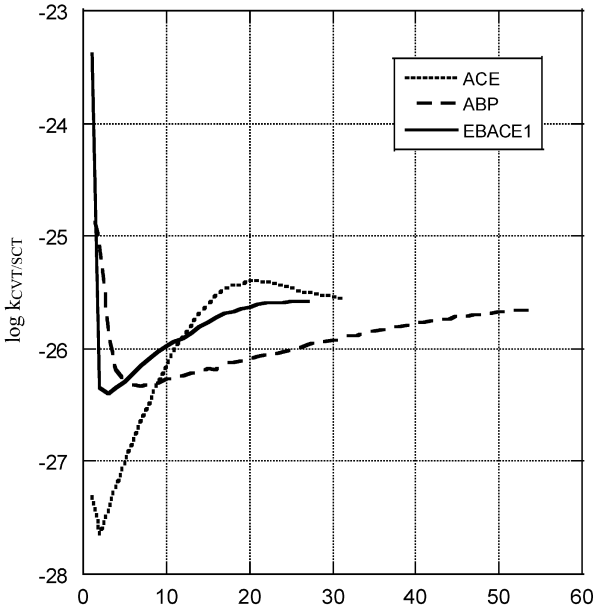

Fig. 8. The logarithm of the calculated CVT/SCT rate constants at 298 K for R2 in different hessian computing schemes. The dotted line represents asymmetric cyclic execution scheme (ACE), the dashed line represents asymmetric block partitioning scheme (ABP), and the solid line represents energy block and asymmetric block partitioning scheme (EBACE-1).

tions using five different parallel Hessian computing schemes based on a predetermined MEP taking advantage of the parallel computing cluster. The Energy Block and Asymmetric Cyclic Execution (EBACE) scheme is not only physically intuitive but also generates converged rate constants with the least number of computed Hessians required compared to the simple Asymmetric Cyclic Execution (ACE) and Asymmetric Block Partitioning (ABP) schemes. If the number of the computing nodes used is large enough, further partitioning of energy blocks (such as in EBACE-2) can be done but no significant improvements were observed. The PHC software will enable more researchers to use VTST/MT for calculating the rate constants of complex chemical systems by performing calculations in a parallel computing environment. Due to the limitation in POLYRATE, the transmission coefficient in Large Curvature Tunneling (LCT) approximation cannot be calculated via the *fu31* file input option. However, a separate program for parallel generating the LCT paths and modifications to the POLYRATE program for reading in the required information of the LCT rate constants via external files are undergoing development. With these new implementations, we can take advantage of the parallel computing environment while carrying out full VTST/MT calculation. When the chemical system of interest becomes large, the evaluation of Hessian matrix can be costly in terms of computing and storage resources; a new proposed method[29] for applying the updated Hessians with VTST/MT is discussed in a separated paper and it can be used with the EBACE scheme developed here together for faster calculation of the rate constants of large complex systems using VTST/MT.

ACKNOWLEDGMENT

This work is supported by the National Science Council of Taiwan (NSC92-2113-M-390-005). The authors would like to thank Dr. Jay Srinivasan and Prof. Wei-Ping Hu for reading the manuscript and many helpful suggestions.

Received May 8, 2006.

REFERENCES

1. (a) Truhlar, D. G.; Garrett, B. C. In *Annual Reviews of Physical Chemistry*; Rabinovitch, B. S., Schurr, J. M., Strauss, H. L., Eds; Annual Reviews: Palo Alto, **1984**, *35*, 159. (b) Truhlar, D. G.; Isaacson, A. D.; Garrett, B. C. In *The Theory of Chemical Reaction Dynamics*; Bear, M., Ed.; CRC Press: Boca Raton, **1985**, *4*, 65. (c) Truhlar, D. G.; Garrett, B. C.; Klippenstein, S. J. *J. Phys. Chem. A* **1996**, *100*, 12771. (d) Truhlar, D. G.; Hase, W. L.; Hynes, J. T. *J. Phys. Chem.* **1983**, *87*, 2664; 5523(E).
2. (a) Corchardo, J. C.; Espinosa-Garcia, J.; Roberto-Neto, O.; Chuang, Y.-Y.; Truhlar, D. G. *J. Phys. Chem. A* **1998**, *102*, 4899. (b) Pu, J.; Corchado, H. C.; Truhlar, D. G. *J. Chem. Phys.* **2001**, *115*, 6266.
3. (a) Chuang, Y.-Y.; Cramer, C. J.; Truhlar, D. G. *Int. J. Quantum. Chem.* (Sanibel Issue) **1998**, *80*, 887. (b) Chuang, Y.-Y.; Radhakrishanan, M. L.; Fast, P. L.; Cramer, C. J.; Truhlar, D. G. *J. Phys. Chem. A* **1999**, *103*, 4893. (c) Chuang, Y.-Y.; Truhlar, D. G. *J. Am. Chem. Soc.* **1999**, *121*, 10157.
4. (a) Wonchoba, S. E.; Truhlar, D. G. *J. Chem. Phys.* **1993**, *99*, 9637. (b) Wonchoba, S. E.; Hu, W.-P.; Truhlar, D. G. In *Theoretical and Computational Approaches to Interface Phenomena*; Sellers, H., Golab, J. T., Eds.; Plenum: New York, 1994; p 1. (c) Wonchoba, S. E.; Hu, W.-P.; Truhlar, D. G. *Phys. Rev. B* **1995**, *51*, 9985.
5. (a) Alhambra, C.; Corchado, J. C.; Sanchez, M. L.; Gar-

cia-Viloca, M.; Gao, J.; Truhlar, D. G. *J. Phys. Chem. B* **2001**, *105*, 11326. (b) Truhlar, D. G.; Gao, J.; Alhambra, C.; Garcia-Viloca, M.; Corchado, J. C.; Sanchez, M. L.; Villa, J. *Acc. Chem. Res*. **2002**, *35*, 341. (c) Alhambra, C.; Gao, J.; Corchado, J. C.; Villa, J.; Truhlar, D. G. *J. Am. Chem. Soc.* **1999**, *121*, 2253. (d) Garcia-Viloca, M.; Alhambra, C.; Truhlar, D. G.; Gao, J. *J. Chem. Phys*. **2001**, *114*, 9953. (e) Alhambra, C.; Gao, J.; Corchado, J. C.; Villa, J.; Truhlar, D. G. *J. Am. Chem. Soc*. **1999**, *121*, 2253. (f) Alhambra, C.; Corchado, J. C.; Sanchez, M. L.; Gao, J.; Truhlar, D. G. *J. Am. Chem. Soc.* **2000**, *122*, 8197.

6. Corchado, J. C.; Chuang, Y.-Y.; Fast, P. L.; Villa, J.; Hu, W.-P.; Liu, Y.-P.; Lynch, G. C.; Nguyen, K.; Jackels, C. F.; Melissas, V. S.; Lynch, B. J.; Rossi, I.; Coitino, E. L.; Fernandez-Ramos, A.; Pu, J.; Albu, T.; Steckler, R. S.; Garrett, B. C.; Isassacson, A. D.; Truhlar, D. G. POLYRATE 9.3, University of Minnesota, 2004. [http://comp.chem.umn.edu/polyrate]
7. Heidrich, D.; Kliesch, W.; Quapp, W. *Properties of Chemically Interesting Potential Energy Surfaces*; Springer-Verlag: Berlin, 1991.
8. (a) Melissas, V. S.; Truhlar, D. G. *J. Chem. Phys*. **1993**, *99*, 1013. (b) Corchado, J. C.; Coitino, E. L.; Chuang, Y.-Y.; Fast, P. L.; Truhlar, D. G. *J. Phys. Chem. A* **1998**, *102*, 2424.
9. (a) Hu, W.-P.; Liu, Y.-P.; Truhlar, D. G. *Faraday Trans. Chem. Soc*. **1994**, *90*, 1715. (b) Chuang, Y.-Y.; Truhlar, D. G. *J. Phys. Chem. A* **1997**, *101*, 3808; **1997**, *101*, 8741 (E).
10. Szabo, A.; Ostlund, N. S. In *Modern Quantum Chemistry*; McGraw Hill: New York, 1982.
11. Dewar, M. J. S.; Zoebish, E. G.; Heely, E. F.; Stewart, J. J. P. *J. Am. Chem. Soc.* **1985**, *107*, 3902.
12. Truhlar, D. G. In *The Reaction Path in Chemistry: Current Approaches and Perspectives*; Heidrich, D., Ed.; Kluwer: Dordrecht, 1995; p 229.
13. (a) Marcus, R. A. *J. Chem. Phys*. 1966, *45*, 4493; 1968, *49*, 2610, 2617. (b) Truhlar, D. G.; Kuppermann, A. *J. Chem. Phys*. **1970**, *52*, 3842; *J. Am. Chem. Soc.* **1971**, *93*, 1840. (c) Fukui, K. In *The World of Quantum Chemistry*; Daudel, R.; Pullman, B. Eds.; Reidel, Dordrecht, 1974; p 113. (d) Fukui, K. *J. Am. Chem. Soc.* **1981**, *14*, 363.
14. McQuarrie, D. A. *Statistical Mechanics*; Harper & Row: New York, 1976; pp 133-136.
15. Califano, S. *Vibrational States*; Wiley: New York, 1976.
16. Isaacson, A. D.; Truhlar, D. G. *J. Chem. Phys.* **1982**, *76*, 1380.
17. Garrett, B. C.; Truhlar, D. G.; Grev, R. S.; Magnuson. A. W. *J. Phys. Chem*. **1980**, *84*, 1730; **1983**, *87*, 4554 (E).
18. Liu, Y.-P.; Lynch, G. C.; Troung, T. N.; Lu, D.-h.; Truhlar, D. G.; Garrett, B. C. *J. Am. Chem. Soc*. **1993**, *115*, 2408.
19. (a) Fernandez-Ramos, A.; Truhlar, D. G. *J. Chem. Phys*. **2001**, *114*, 1491. (b) Fernandez-Ramos, A.; Truhlar, D. G.; Corchado, J. C.; Espinosa-Garcia, J. *J. Phys. Chem. A* **2002**, *106*, 4857.
20. Wilkinson, B.; Allen, M. *Parallel Programming: Techniques and Applications Using Networked Workstations and Parallel Computers*; Pearson: London, 2005.
21. Grama, A.; Gupta, A.; Karypis, G.; Kumar, V. *Introduction to Parallel Computing*; Pearson: London, 2003.
22. Burns, G.; Daoud, R.; Vaigl, J. *LAM: An Open Cluster Environment for MPI*; Proceedings of Super Computing Symposium, 1994; pp 379-386. [http://www.lam-mpi.org/download/files/lam-papers.tar.gz]
23. Melissas, V. S.; Truhlar, D. G.; Garrett, B. C. *J. Chem. Phys.* **1992**, *96*, 5758.
24. Page, M.; McIver, Jr., J. W. *J. Chem. Phys.* **1988**, *88*, 922.
25. (a) Fast, P. L.; Truhlar, D. G. *J. Chem. Phys.* **1998**, *109*, 3721. (b) Villa, J.; Truhlar, D. G. *Theo. Chem. Acc.* **1997**, *97*, 317.
26. Corchado, J. C.; Chuang, Y.-Y.; Coitino, E. L.; Truhlar, D. G. GAUSSRATE 9.3, University of Minnesota, 2004, based on POLYRATE 9.3 and GAUSSIAN 03. [http://comp.chem.umn.edu/gaussrate].
27. Gaussian 03, Revision C.02, Frisch, M. J.; Trucks, G. W.; Schlegel, H. B.; Scuseria, G. E.; Robb, M. A.; Cheeseman, J. R.; Montgomery, Jr., J. A.; Vreven, T.; Kudin, K. N.; Burant, J. C.; Millam, J. M.; Iyengar, S. S.; Tomasi, J.; Barone, V.; Mennucci, B.; Cossi, M.; Scalmani, G.; Rega, N.; Petersson, G. A.; Nakatsuji, H.; Hada, M.; Ehara, M.; Toyota, K.; Fukuda, R.; Hasegawa, J.; Ishida, M.; Nakajima, T.; Honda, Y.; Kitao, O.; Nakai, H.; Klene, M.; Li, X.; Knox, J. E.; Hratchian, H. P.; Cross, J. B.; Bakken, V.; Adamo, C.; Jaramillo, J.; Gomperts, R.; Stratmann, R. E.; Yazyev, O.; Austin, A. J.; Cammi, R.; Pomelli, C.; Ochterski, J. W.; Ayala, P. Y.; Morokuma, K.; Voth, G. A.; Salvador, P.; Dannenberg, J. J.; Zakrzewski, V. G.; Dapprich, S.; Daniels, A. D.; Strain, M. C.; Farkas, O.; Malick, D. K.; Rabuck, A. D.; Raghavachari, K.; Foresman, J. B.; Ortiz, J. V.; Cui, Q.; Baboul, A. G.; Clifford, S.; Cioslowski, J.; Stefanov, B. B.; Liu, G.; Liashenko, A.; Piskorz, P.; Komaromi, I.; Martin, R. L.; Fox, D. J.; Keith, T.; Al-Laham, M. A.; Peng, C. Y.; Nanayakkara, A.; Challacombe, M.; Gill, P. M. W.; Johnson, B.; Chen, W.; Wong, M. W.; Gonzalez, C.; Pople, J. A. Gaussian, Inc., Wallingford CT, 2004.
28. (a) Renka, R. *J. SIAM J. Stat. Comput.* **1987**, *8*, 393. (b) Renka, R. J. *ACM Trans. Math. Software* **1993**, *19*, 81.
29. Chuang, Y.-Y. *J. Comp. Chem.,* submitted.