Theor Chim Acta (1988) 74: 85-99

Theoretica Chimica Acta

Application of a dynamic method of minimisation in the study of reaction surfaces

Colin M. Smith

Division of Molecular Engineering, Kyoto University, Kyoto 606, Japan

(Received January 30, 1987; revised and accepted February 18, 1988)

This paper describes the application of Snyman's dynamic minimisation method to a fitted potential surface of H_3. Comparisons are made with conventional algorithms. A method is described to extend Snyman's method so that it will find only a particular kind of stationary point. It is emphasized that this method enables saddle points to be found without having to resort to approaches based on trial and error.

Key words: Dynamic minimisation — Optimisation — Stationary point — Transition state — Reaction surface

1. Introduction

A fundamental technique used in the theoretical study of chemical reactions is that of searching for stationary points on a potential energy surface. The stationary points which are minima correspond to stable structures of the system and those for which the Hessian has one and only one negative eigenvalue correspond to transition states. Standard methods of optimisation theory have to be adapted to create algorithms which will search for such saddle points, since most methods use theory which is based on being able to approximate the surface locally by a quadratic form and find the required saddle point only when started in a region in which the Hessian has only one negative eigenvalue. The adapted theory has to enable a reasonable starting region to be found and trial and error techniques have to be adopted [1-7]. In this paper the performance of a prototype minimisation algorithm based on the solution of an associated dynamic system [8] is studied in connection with a fitted adiabatic surface of H_3 [9], and an adaptation is described which will make the algorithm converge only to a desired kind of saddle point.

2. Method

Snyman's minimisation algorithm is a novel method for performing an unconstrained minimisation of a function of many variables for which first derivatives are known explicitly [8]. It does not require the use of second derivatives and it is not even necessary to calculate the value of the function at all during the calculation.

The algorithm is different conceptually from standard methods. It treats the function to be minimised, $f(\boldsymbol{x})$, as the potential energy of a particle of unit mass moving in a conservative force field. Thus the negatives of the first spatial derivatives give the forces acting on the particle. In each iteration the classical equations of motion for the velocity and position of the particle are solved using Euler backward and forward approximate integration respectively.

$$\boldsymbol{v}_{k+1} = \boldsymbol{v}_k - \boldsymbol{g}_{k+1}\Delta T \tag{1a}$$

$$\boldsymbol{x}_{k+1} = \boldsymbol{x}_k + \boldsymbol{v}_k\Delta T \tag{1b}$$

where ΔT is a suitably chosen time step, $\boldsymbol{g}_i$ is the vector of first derivatives (=−force vector), and $\boldsymbol{v}_i$ and $\boldsymbol{x}_i$ are the velocity and position of the particle at the start of the ith time step (minimisation iteration) respectively.
Given a time step, an initial position and initial velocity, enables one to find a unique trajectory in each iteration. (Note that as far as the algorithm is concerned, ΔT is merely a parameter. It does not have any units. It is thought of as a time step only in a physical interpretation.) On this trajectory the total energy (potential plus kinetic) is conserved. If the particle's speed increases as it moves along this path, then its potential energy (the function $f(\boldsymbol{x})$) must decrease. Therefore, if it starts from rest it must follow a path on which its potential energy is non-increasing. This means that unless the initial position is a minimum, the method can be sure to find a path along which $f(\boldsymbol{x})$ decreases. The basic Snyman algorithm may be summarized:
Consider $\|\boldsymbol{v}_{k+1}\|$, $k = 0, 1, 2, \ldots$.
Whenever $\|\boldsymbol{v}_{k+1}\| < \|\boldsymbol{v}_k\|$ reset $\boldsymbol{v}_k$ to $\boldsymbol{0}$ and repeat step k.

The performance of the routine depends on the choice of the size of the time step ΔT. A small value for ΔT means that the numerical approximation is good and an accurate solution of the equations of motion (1a, b) is found. Choosing ΔT too small can result in a slow calculation, and choosing ΔT too large, apart from obtaining an inaccurate path, can result in a minimum being overshot repeatedly. Snyman discusses the practical implications of the method giving heuristic procedures [8] which make it more efficient, gives an automatic time step routine [10], and extends the algorithm for global minimisation [11] using Bayesian statistics. The time step is allowed to increase at a preset "variable compound interest" rate as long as the change in modulus of $\boldsymbol{x}$ does not exceed a certain value, in which case ΔT is held fixed and the maximum allowable change in $\boldsymbol{x}$ is taken. The current value of ΔT is halved and the algorithm is re-started half-way between the present and previous positions, at a velocity equal to $(\boldsymbol{v} + \boldsymbol{v}_{\text{old}})/4$, if the scalar product of successive first derivative vectors $\boldsymbol{g}$ is negative in three consecutive iterations. This ensures that the time step may

become small enough near to a stationary point, which is necessary for non-erratic behaviour of the particle.

As the aim of the calculation is to minimise $f(\boldsymbol{x})$ rather than to solve the equations of motion as accurately as possible, Snyman shows how the approximate solution of the equations of motion using (1a, b) leads to a modified energy conserving relationship and changes the basic algorithm given above accordingly, but asserts that the algorithm can never converge to a stationary point which is not a proper local minimum [8].

These heuristic procedures increase the efficiency of the algorithm and the automatic time step method can make the method more efficient for a steep function and sometimes it can cause the program to converge accidentally.

3. Application to H_3 fitted surface

To find out how the Snyman algorithm would perform on a reaction surface, it was applied to the H_3 fitted adiabatic surface of Hall and Okada [9] which is given very conveniently in the form,

$$f(\boldsymbol{x}) = f(X, Y, Z) = \sum_{r,s,t=0}^{r+s+t=N} a_{rst} X^r Y^s Z^t$$

where $X = \exp(-ax_1)$, $Y = \exp(-bx_2)$, $Z = \exp(-cx_3)$ and $a = b = c$ and $a_{rst} = a_{srt} = a_{rts}$, and x_1, x_2 and x_3 are the distances between the three Hydrogen atoms (they are the components of $\boldsymbol{x}$) and N is the order of the fit.

It is straightforward to calculate the first derivatives of this function and formulae for $f(\boldsymbol{x})$ and these first derivatives were inserted into Snyman's algorithm for analysis. The time step ΔT was set initially to 0.5 and the automatic time step method [10] was used so that ΔT could become as large as possible during the calculation. The maximum allowable change in the modulus of $\boldsymbol{x}$ during an iteration was set to 1, and the calculation was assumed to have converged when the modulus of the vector of first derivitives $\boldsymbol{g}$ had been reduced to 10^{-8}.

The algorithm found a local minimum, which corresponds to a linear configuration, from a variety of starting points. It should be noted that this is not an unconstrained minimisation because x_1, x_2 and x_3 must satisfy the triangle inequality, so that when $x_3 = x_1 + x_2$, i.e. a linear configuration, a degree of freedom is lost and the derivatives with respect to x_3 are equal to linear combinations of derivatives with respect to x_1 and x_2, e.g.

$$\frac{\partial f}{\partial x_3} = \frac{1}{2}\left[\frac{\partial f}{\partial x_1} + \frac{\partial f}{\partial x_2}\right] \quad \text{if } x_3 = x_1 + x_2,$$

and the Hessian has a zero eigenvalue. Therefore the algorithm is given values for the first derivative with respect to x_3 which show a discontinuity at the line $x_3 = x_1 + x_2$. Similarly it experiences a discontinuity in derivatives with respect to x_1 on the line $x_1 = x_2 + x_3$ and with respect to x_2 on the line $x_2 = x_3 + x_1$.

As will be discussed later, although the method is usually guaranteed not to converge to a saddle point, it did find saddle points corresponding to isosceles triangle arrangements if it was started from such a configuration. However, if started close to such a saddle point from a near isosceles arrangement, symmetry being broken, it found a path which moved the particle away to a proper local minimum. Similarly when started from an equilateral triangle configuration it found the H_3 saddle point. In both cases, if the algorithm was allowed to continue, so that the first derivatives could become very small (i.e. around 10^{-12}) it was found that rounding errors could cause the isosceles or equilateral symmetry to be broken and slowly a path was found which lead to a proper minimum.

4. Adaptation to find saddle points

The progress of the minimisation strategy may be monitored by observing the first derivatives of $f(\boldsymbol{x})$ along the principal curvatures at the current iteration position. Denote these by the column vector $\boldsymbol{g}_p$. (The components of $\boldsymbol{g}_p$ may be thought of as minus the principal forces, i.e. the forces acting along the principal curvatures of the surface.) If the orthogonal transformation which diagonalises the Hessian matrix is represented by the matrix $\boldsymbol{U}$, i.e.

$$\boldsymbol{U}^T \cdot \boldsymbol{H} \cdot \boldsymbol{U} = \boldsymbol{\Lambda}$$

where $\boldsymbol{\Lambda}$ is a diagonal matrix then,

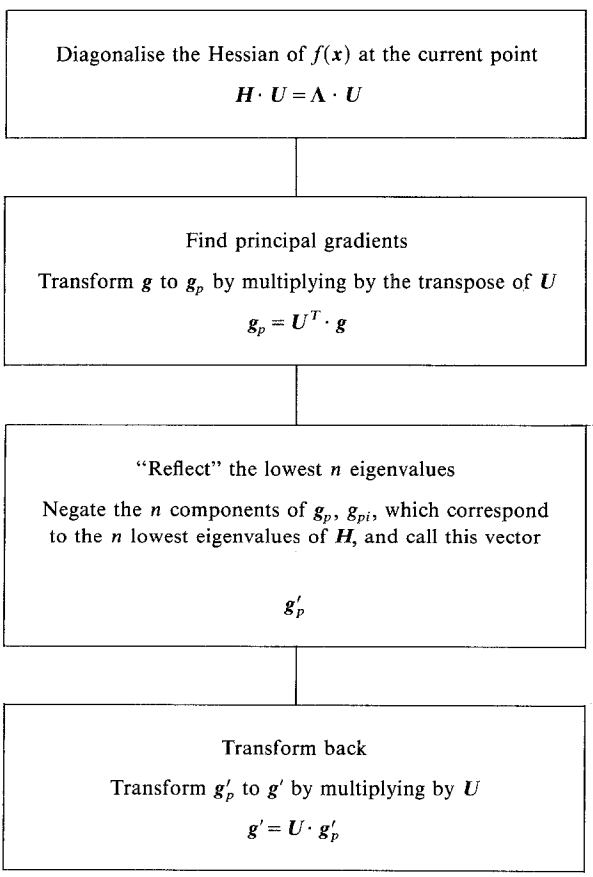
$$\boldsymbol{g}_p = \boldsymbol{U}^T \cdot \boldsymbol{g}$$

where the superscript T of a matrix or a vector denotes its transpose. When an iteration results in $f(\boldsymbol{x})$ being reduced, the velocity increases, hence the dominant elements of $\boldsymbol{g}_p$ do not change sign, see equations (1a, b). If the Hessian is positive definite, (all eigenvalues are positive) then the algorithm is certain to find a direction which minimises the function and reduces the moduli of all the elements of $\boldsymbol{g}_p$ so that a point closer to a minimum is found. If the Hessian is not positive definite, i.e. it has n negative eigenvalues, then even if the current point is close to a saddle point the algorithm eventually will find a path on which at least one of the n elements of $\boldsymbol{g}_p$ corresponding to a negative eigenvalue will increase as $f(\boldsymbol{x})$ decreases. However, if one or more of these n elements are zero identically, for a continuous set of points, the algorithm behaves in a way that this subset is not used in the search for a minimum. The method was found never to converge to a saddle point provided; a) that the convergence parameter, which determines how small the first derivatives must be to stop the minimisation, is set low enough and b) such a continuous set of points is not encountered. It is for reason b) that an isosceles triangle saddle point was identified as a minimum when the starting point also had an isosceles arrangement. In this case the symmetry of the isosceles triangle ensured that the Hessian's lowest eigenvalue had the eigenvector $(0, 1, -1)^T$ (in this paper, vectors are represented by column matrices and for this example positions 2 and 3 correspond to the equal sides of the triangle) and because the relevant two first derivatives were equal (also due to this symmetry), the component of $\boldsymbol{g}_p$ which corresponds to this lowest eigenvalue was zero for all such configurations. Except for unexpected effects due to rounding errors,

symmetry constrained the algorithm to perform the minimisation within the set of isosceles triangle configurations and in that sense it did find a proper minimum.

This behaviour near stationary points suggests a way in which the algorithm may be adapted so that it will converge to a saddle point and not to any other kind of stationary point. This is done by applying a transformation to $f(\boldsymbol{x})$ such that the image surface $f'(\boldsymbol{x})$ has a minimum where $f(\boldsymbol{x})$ has a saddle point (and a saddle point where $f(\boldsymbol{x})$ has a minimum). Let (N, n) saddle point denote a saddle point on an N-dimensional surface which has n negative principal curvatures. Consider an (N, n) saddle point on a surface $f(\boldsymbol{x})$. This is always a local minimum on a related surface $f'(\boldsymbol{x})$ which is defined as having its principal curvatures corresponding to its $N-n$ highest eigenvalues equal to those of $f(\boldsymbol{x})$ for all $\boldsymbol{x}$, and the remaining n principal curvatures equal to minus those of $f(\boldsymbol{x})$ for all $\boldsymbol{x}$. Hence a saddle point on $f(\boldsymbol{x})$ may be found by minimising $f'(\boldsymbol{x})$. Such a procedure may be applied very easily to Snyman's algorithm and the first spatial derivatives of $f'(\boldsymbol{x})$ may be found as shown in the flowchart below. This is not in the spirit

The procedure for obtaining first derivatives for the image surface



of the Snyman method, however, because the Hessian needs to be calculated exactly to define $f'(\mathbf{x})$, but it does give a method which will converge only to the required kind of saddle point and may be started *abitrarily near a minimum point* as long as the convergence parameter is set low enough.

This new method for finding saddle points does not suffer from the inherent problem in other methods which is due to the fact that only one orthogonal trajectory passes through a saddle point. Because the saddle point to be found is a minimum on the image surface, there are infinitely many orthogonal trajectories of that surface which end at it. It should be noted that the gradient extremal method [12, 13] also avoids the above problem very effectively. However, effort needs to be made to "walk along" the gradient extremal, which leads from a given minimum to a given saddle point, rather closely and this path may not always be a very direct route.

It is a non-trivial exercise to imagine the shape of the image surface described above above for a given surface $f(\boldsymbol{x})$. Generally, negating a principal curvature at a point implies applying a reflection which depends on the eigenvectors of the Hessian at that point. $f(\boldsymbol{x})$ for the H_3 fitted surface has saddle points at (1.43, 4.14, 4.14) and (3.42, 1.76, 1.76) which have one negative principal curvature, and a saddle point at (2.11, 2.11, 2.11) which has two. Therefore $f'(\boldsymbol{x})$ has minima at (1.43, 4.14, 4.14) and (3.42, 1.76, 1.76) and a saddle point with one negative curvature at (2.11, 2.11, 2.11). *Valley bottoms are transformed to cols with one negative curvature and vice versa.*

In the search for an (N, n) saddle point, points for which $f(\boldsymbol{x})$ has $n+1$ (or greater) equal lowest eigenvalues may be approached. As an optimisation path passes near a point whose lowest $n+1$ principal curvatures are equal, there is a degeneracy which can cause a different set of eigenvalues to be negated. In effect this degeneracy introduces a discontinuity in the first derivatives $\boldsymbol{g}'$ at this point and it can behave like a stationary point. Such a pseudo stationary point would always have at least one negative eigenvalue on $f'(\boldsymbol{x})$ and therefore would not cause the program to converge, but may slow the particle down.

It should be noted that it is only useful to implement this "reflection" technique to find saddle points, on a minimisation routine which; (a) can distinguish minimima from other stationary points and (b) does not use function values at all.

The modified method was used to search for all the saddle points on the fitted H_3 surface and was seen to be very robust and always found the correct type of saddle point although sometimes convergence was rather slow for a bad starting point. However, if the initial point was very close to a $(3, n^*)$ saddle point the algorithm had no problem in finding a set of paths which terminated at a desired $(3, n)$ saddle point $(n \neq n^*)$.

To apply the method to an *ab-initio* potential energy surface, strictly it is necessary to be able to calculate exact first and second spatial derivatives for all points on the surface. If the algorithm were incorporated into an SCF program, these derivatives would have to be calculated at the end of the SCF stage, and then the algorithm would use them to find the first derivatives for the image surface

and determine the next point on the optimisation path. Therefore these derivitives would have to be evaluated at each point visited on the path. Such a scheme would be very time consuming for a large quantum mechanical system. It is suggested here that this new method be used together with a standard one. At the start of the calculation it would search for a region where the Hessian of the image surface is positive definite and then the standard method would be used to finish the calculation. An example to illustrate this for a simple system is given later. The starting point can be any point on $f(\boldsymbol{x})$ for which the first derivatives are non-zero (they may be very small but they must not be zero). The advantage gained on a complicated reaction surface, for which there is very little chemical insight, would be that the intrinsic ability of the algorithm, to distinguish between different kinds of stationary point, makes the use of trial and error methods to climb the hill from a minimum [1-7] unnecessary. It may also need fewer iterations than a twisty gradient extremal [12, 13]. To reduce computing costs for this composite algorithm it is probably sufficient to use approximate derivatives for part of the Snyman section if the distance between successive iterates is small although it is necessary to know the second derivatives at some point to know when to switch to the standard method.

5. Comparison with standard methods

Quasi-Newton minimisation routines and a non-linear simultaneous equation solver using Brent's method were also used to study the H_3 surface to enable comparisons to be made with Snyman's method. These other routines are contained in the Scientific Subroutine Library II at Kyoto University. When the starting configuration for each run was chosen to be (1.650001, 1.65, 1.65), Brent's method was seen to converge to the point (2.105, 2.105, 2.105), where the value of f is -0.08, in 8 iterations. This point is actually a (3, 2) saddle point. The Quasi-Newton method, which used exact first derivatives, converged to the point (3.33, 1.67, 1.67) in 525 iterations. The function value at this point is -0.154 but it is not a stationary point for the gradients are of the order 10^{-1} (in this case convergence meant that the function $f(\boldsymbol{x})$ could not be made any lower). Another Quasi-Newton routine, which used approximate numerical first derivatives, was also employed. This converged to a minimum "near ∞" in 207 iterations where the numerical first derivatives were $\sim 10^{-5}$. Snyman's method converged to the straight-line minimum configuration (3.77, 5.21, 1.44) in 329 iterations where $f = -0.173$, when the convergence parameter was 10^{-8}. When a larger value of 10^{-7} was used instead, the algorithm stopped near a saddle point at (3.42, 1.76, 1.76) where $f = -0.156$. Each method was also timed for a run which printed only final values of the variables. These results, together with those for another starting point, (1.650001, 1.65, 3.33), are given in Table 1.

6. Conclusion

The results in Table 1 show that Snyman's routine is a robust minimisation method which is able to distinguish minima from other stationary points in a

Table 1. Final points obtained by Snyman's and standard routines from the same starting points on the H_3 fitted surface of Hall and Okada

Starting from (1.650001, 1.65, 1.65)					
Method	Final point	$\|g\|$	f	Steps	CPU s
Snyman	(3.42, 1.76, 1.76)	10^{-7}	−0.156	97	0.3
Snyman	(5.21, 1.44, 3.77)	10^{-8}	−0.173	329	0.8
Q-Newton	(3.34, 1.67, 1.67)	10^{-1}	−0.154	525	0.5
Brent	(2.10, 2.10, 2.10)	10^{-7}	−0.083	8	0.1
Starting from (1.650001, 1.65, 3.33)					
Method	Final point	$\|g\|$	f	Steps	CPU s
Snyman	(1.76, 1.76, 3.42)	10^{-5}	−0.156	63	0.2
Snyman	(3.77, 1.44, 5.21)	10^{-7}	−0.173	241	0.6
Q-Newton	(3.98, 1.46, 5.44)	10^{-1}	−0.173	171	0.2
Brent	(1.76, 1.76, 3.42)	10^{-7}	−0.156	5	0.1

way that standard methods cannot. Brent's method, as expected, converged to the nearest stationary point as it was merely solving a set of non-linear equations, and converged very quickly. However the Quasi-Newton method, which used first derivatives, seemed to find the H_3 surface particularly difficult to cope with and was unable to find a proper minimum. It may be important to note that the modulus of the first derivative vector in the straight line configuration is $\sim 10^{-2}$ if the three variables are treated as independent. This severe discontinuity may have caused Snyman's routine to need a large number of iterations, and also confused the Quasi-Newton routine, whose convergence criterion depends upon the absolute change in x during an iteration. The fact that Synman's routine could converge in such circumstances whereas the Newton routine could not, may be significant in an application on a more complicated surface.

It should be noted that this is a prototype routine and contains parameters which determine its efficiency for a particular problem [8]. It has not been compared with a minimisation method which has been tuned to work well on a Quantum Chemical problem. The aim here has been to compare the performance of the basic new algorithm, with the basic algorithms of standard methods, on a simple Quantum problem and suggest that these results together with the fact that it may be made to find *only* saddle points, by using the reflection method, mean that it is worth refining and using in Chemical Reaction studies.

7. Application to model two-dimensional surfaces

In order to illustrate the nature of the optimisation paths of Snyman's method and their dependence upon the initial time step parameter ΔT, the two two

dimensional functions, on which Simons et al. [3] tested their surface walking method, were studied as model potential surfaces. The first was the Rosenbrook function, studied by Crippen and Scheraga [4] and also by Snyman [8]. It is given by;

$$V_{CS}(x, y) = 100(y - x^2)^2 + (1 - x)^2$$

$V_{CS}(x, y)$ has a minimum at the point (1, 1) and a stream bed along the line $y = x^2$. The routine was started from various points using various values for initial ΔT and was found to converge to the minimum each time. Results for an initial starting point of (−5, −5) are given in Table 2. A run was assumed to have converged when the modulus of the vector of first derivatives $\boldsymbol{g}$, had been reduced to 10^{-5}. The number of minimisation steps is seen not to depend upon the initial time step very sensitively and the final value of the time step is approximately the same for each case. Most of the minimsation run was spent on the curve $y = x^2$ and it was here that the automatic time step routine reduced and/or kept the value of ΔT to around 0.04. Only a few iterations were needed to reach $y = x^2$ from the point (−5, −5) when the initial value of ΔT was 0.5 or greater. The paths for initial ΔT values of 0.05 and 50 are illustrated in Fig. 1a,b respectively.

The results show that the minimisation algorithm requires a small time step of around 0.04 near the stream bed $y = x^2$ to ensure that stable minimisation paths can be found which do not repeatedly overshoot the minimum. When started with an intiial ΔT of 0.05 many steps were required to reach the minimum but each path taken was a stable solution of the equations of motion. When the initial time step was 50 many of the paths near $y = x^2$ were unstable and showed erratic behavior. However the latter was the most efficient minimisation.

The second surface studied was that of Cerjan and Miller [5] and is given by;

$$V_{CM}(x, y) = (a - by^2)x^2 \exp(-x^2) + 0.5cy^2$$

In $V_{CM}(x, y)$ a, b and c were given the values 1, 1.2 and 1 respectively. This potential has a minimum at (0, 0) and saddle points at (1, 0) and (−1, 0). The convergence parameter was set to 10^{-8} and minimisation runs starting from (1.34, −1.15) were investigated. Results for different initial time steps are given in Table

Table 2. Values of parameters when the modulus of the first derivative vector of the Crippen-Scheraga function has reached 10^{-5}, in a minimisation run starting from the point (−5, −5) using Snyman's method with different initial time steps

Initial ΔT	Steps	Final ΔT	$V_{CX}(x, y)$
0.005	267	0.037	10^{-12}
0.05	262	0.047	10^{-12}
0.5	230	0.028	10^{-12}
5.0	233	0.031	10^{-14}
50.0	179	0.031	10^{-11}

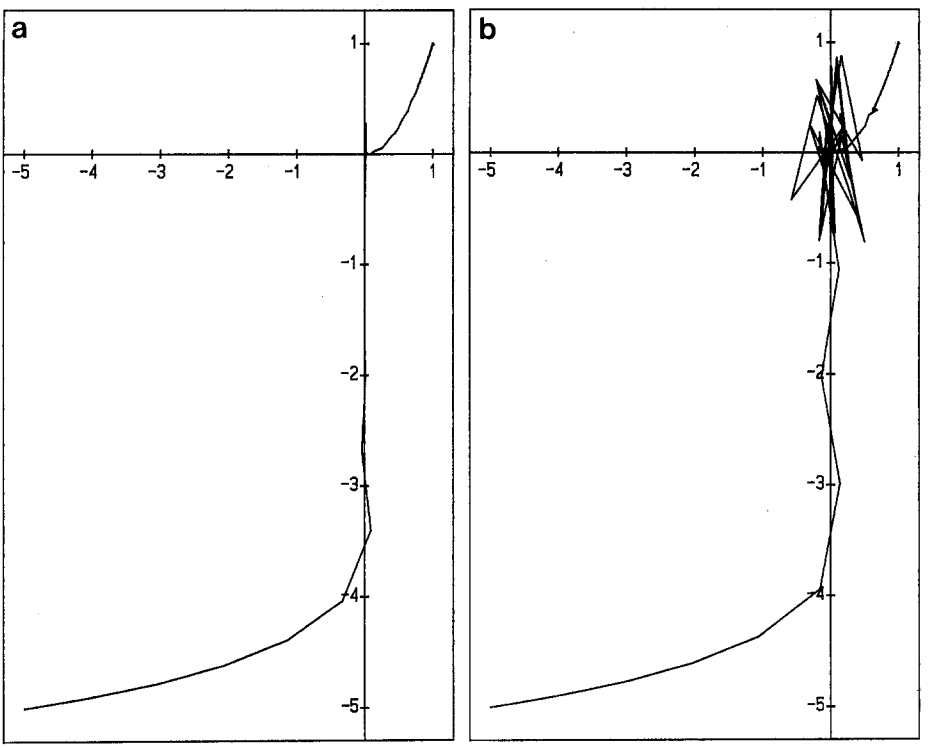


Fig. 1. **a** The paths on surface V_{CS} to the minimum (1, 1) from the starting point (−5, −5) with initial time step of 0.05. **b** Erratic paths on surface V_{CS} to the minimum (1, 1) from the starting point (−5, −5) with initial time step of 50

3. It can be seen that for this potential the final time step for each case was around 0.7, over 10 times that used by the Rosenbrook function. As before, starting with a large time step, meant that inaccurate paths lead to the minimum where ΔT was repeatedly halved until small enough to allow the method to converge, and also a case with erratic paths was the most efficient minimisation. The paths for initial time steps of 0.5 and 50 are illustrated in Fig. 2a, b respectively.

Table 3. Values of parameters when the modulus of the first derivative vector of the Cerjan-Miller function has reached 10^{-8}, in a minimisation run starting from the point (1.34, −1.15) using Snyman's method with different intial time steps

Initial ΔT	Steps	Final ΔT	$V_{CM}(x, y)$
0.005	179	0.57	10^{-17}
0.05	154	0.73	10^{-18}
0.5	104	0.70	10^{-17}
5.0	78	0.67	10^{-17}
50.0	76	0.81	10^{-18}

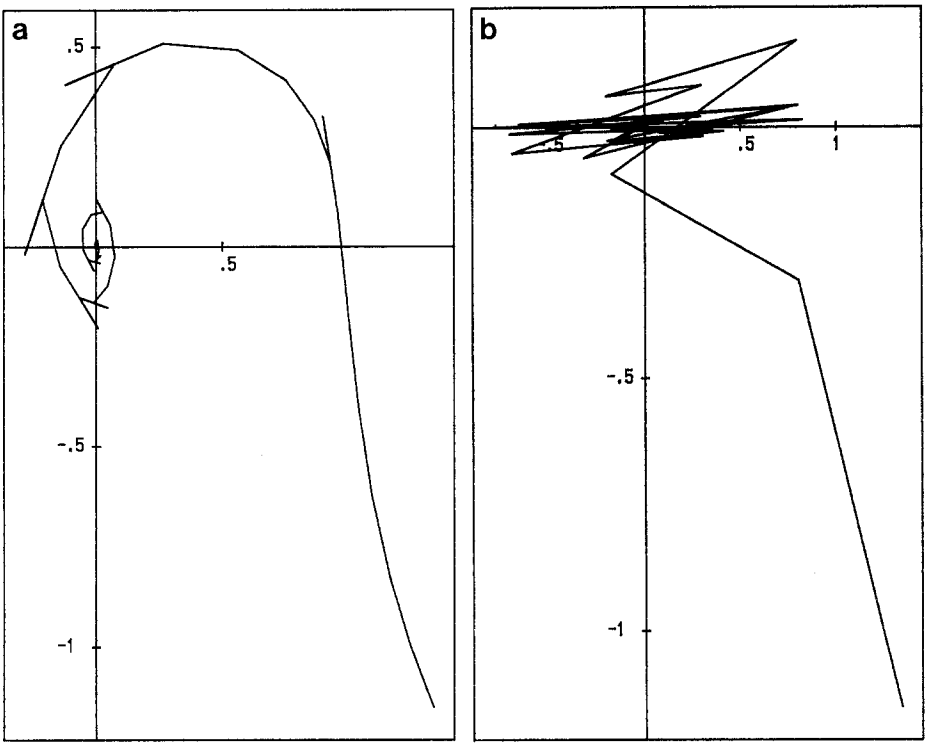


Fig. 2. a The paths on surface V_{CM} to the minimum (0, 0) from the starting point (1.34, −1.15) with initial time step of 0.5. **b** Erratic paths on surface V_{CM} to the minimum (0, 0) from the starting point (1.34, −1.15) with initial time step of 50

It is interesting to note that in Fig. 2a a smooth path passes very close to the saddle point at (1, 0) but, although it is so close and the first derivatives on the path are small ($\sim 10^{-2}$), the algorithm does not converge to this point. In Fig. 2b all of the erratic paths point towards the minimum and some overshoot it because the current time step is too large.

The potential V_{CM} has some characteristics which make finding its saddle points rather difficult. Along the x-axis $\partial V_{CM}/\partial y$ is zero and along the lines $x = 0$ and $x = 1$ $\partial V_{CM}/\partial x$ is zero, therefore V_{CM} is a good test case for using the "reflection" method described in Sect. 4 together with Snyman's program. At the points (± 1, ± 0.94849) the Hessian of V_{CM} has both eigenvalues equal to 0.117089. These points behave like saddle points on the image surface of V_{CM}. Saddle point runs starting at $(\pm 1 + \varepsilon, \pm a)$ where ε is arbitrarily small and the magnitude of a is less than 0.94849 converge very easily to $(\pm 1, 0)$, but if the modulus of a is bigger than 0.94849 the initial path is approximately along the line $x = \pm 1$ in the opposite direction to the nearest saddle point. Hence $(1 + \varepsilon, 1)$ is a good starting point to test the algorithm. Results from saddle point runs which started from $(1 - 10^{-12}, 1)$ are given in Table 4. The convergence parameter was set to 10^{-5}. Some runs converged to the saddle point at (1, 0) others to (−1, 0) and one case did not

Table 4. Values of parameters when the modulus of the first derivative vector of the Cerjan-Miller function has reached 10^{-5}, in a saddle point run starting from the point $(1\text{-}10^{-12}, 1)$ using the adapted Snyman method with different initial time steps. The case for an initial time step of 0.05 diverged

Initial ΔT	Steps	Final ΔT	Final point
0.005	157	0.93	$(-1, 0)$
0.05	300+	0.05	$\sim(-1, -\infty)$
0.5	87	0.65	$(1, 0)$
5.0	60	1.30	$(-1, 0)$
50.0	60	0.40	$(-1, 0)$

converge. Again the most efficient run showed erratic behaviour. In each case the direction of the path was initially away from the nearest saddle point and the algorithm found it difficult to find a path which lead away from the line $x = 1$. Once such a path was found convergence to $(-1, 0)$ or $(1, 0)$ was obtained. No run converged to the minimum. The paths of runs with initial time steps of 0.5 and 50 are given in Fig. 3a, b respectively. In Fig. 3a it can be seen that the paths

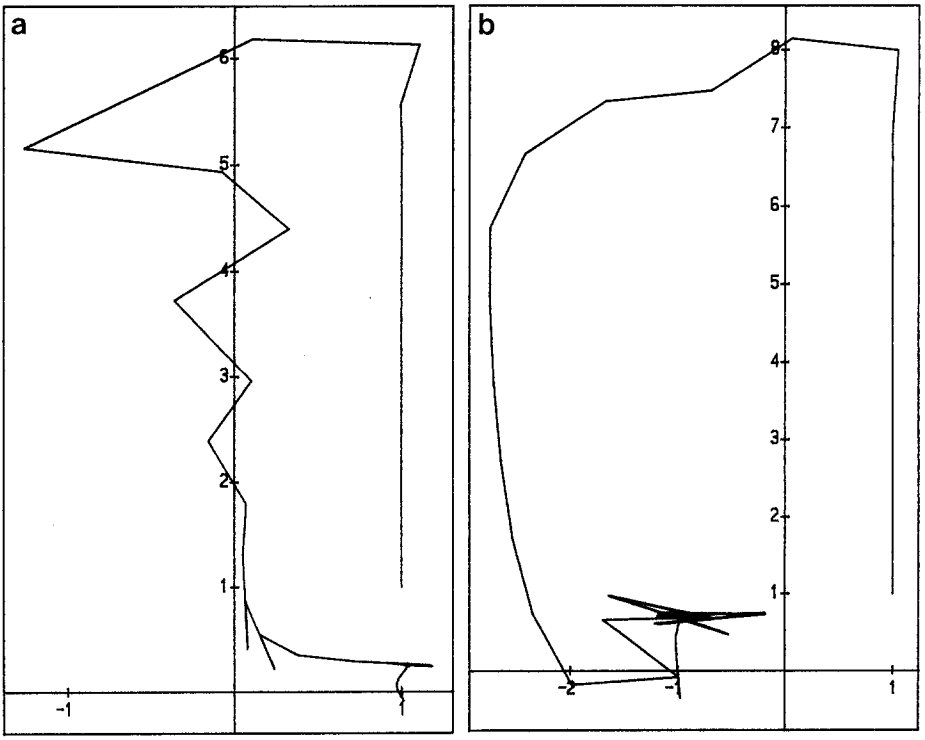


Fig. 3. a The paths on surface V_{CM} to the saddle point $(1, 0)$ from the starting point $(1\text{-}10^{-12}, 1)$ with initial time step of 0.5. **b** Erratic paths on surface V_{CM} to the saddle point $(-1, 0)$ from the starting point $(1\text{-}10^{-12}, 1)$ with initial time step of 50

Table 5. Values of parameters when the modulus of the first derivative vector of the Cerjan–Miller function has reached 10^{-5}, in a saddle point run starting from the point $(10^{-5}, 10^{-5})$ using the adapted Snyman method with different initial time steps

Initial ΔT	Steps	Final ΔT	Final point
0.005	149	1.11	(1, 0)
0.05	170	0.66	(−1, 0)
0.5	126	0.65	(−1, 0)
5.0	42	0.65	(−1, 0)
50.0	61	0.42	(−1, 0)

are diverted away from (0, 0) by the topology of the image surface near (0, 0.645497) where the Hessian of V_{CM} has both eigenvalues equal to 1. This point behaves like a saddle point on the image surface.

A more realistic starting point in reaction surface studies is one which is close to a minimum of a potential surface. Results are given for runs starting from the point $(10^{-5}, 10^{-5})$ in Table 5. Each run converged to a saddle point. The runs which started with small enough time steps to allow stable paths to be found, proceeded slowly up the y-axis until they came near to the point (0, 0.645497), where the Hessian of the function V_{CM} has both eigenvalues equal to 1, and then left the y-axis and, provided that the time step remained small enough, proceeded slowly to the region whose topology was influenced by the point (±1, 0.94849), where the Hessian has both eigenvalues equal to 0.117089, and then proceeded to (±1, 0). The shape of the image function near these points where the eigenvalues are equal caused the time step to be reduced and changed the general direction of the paths. This is shown in Fig. 4a which illustrates the paths for an initial time step of 0.5. Figure 4b shows the most efficient saddle point run; that which started with a time step of 5. Notice that paths from (0, 0) along the x-axis to (±1, 0) were not taken. This is because the point (±0.34976, 0) on the x-axis

Table 6. Values of parameters when the modulus of the first derivative vector of the Cerjan–Miller function has reached 10^{-5}, in a saddle point run starting from the point $(10^{-5}, 10^{-5})$ using the adapted Snyman method with different initial time steps accelerated by Newton's method whenever the Hessian has one positive and one negative eigenvalue and the modulus of the first derivative vector is below 10^{-1}

Initial ΔT	Steps	Final point
0.005	143	(1, 0)
0.05	104	(−1, 0)
0.5	47	(−1, 0)
5.0	11	(−1, 0)
50.0	5	(1, 0)

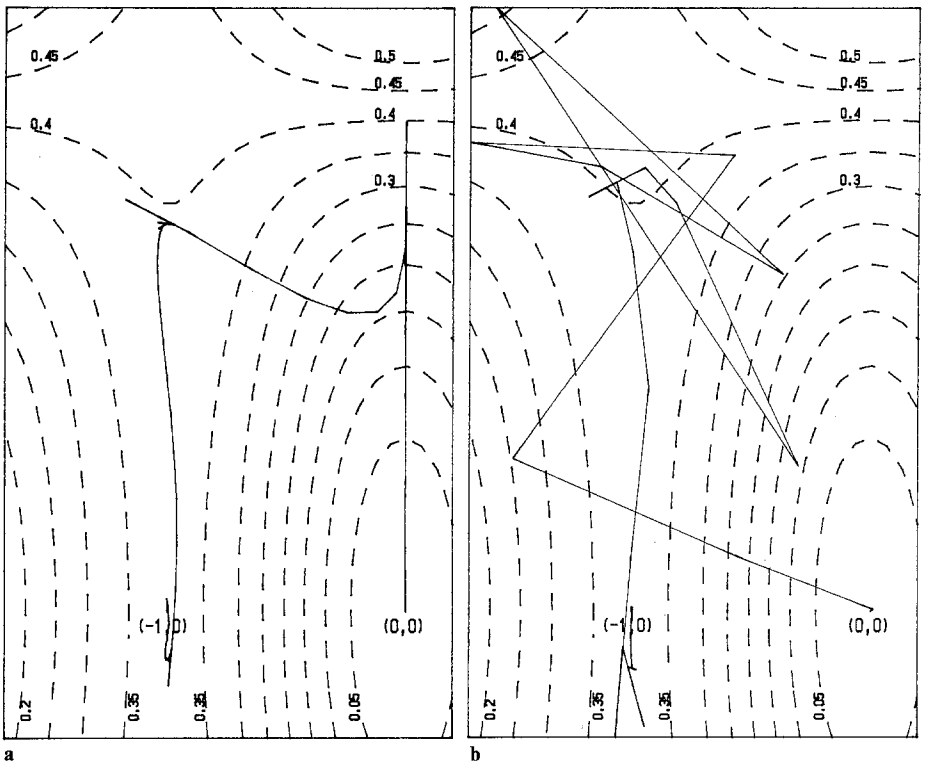


Fig. 4. a The paths on surface V_{CM} to the saddle point $(-1, 0)$ from the starting point $(10^{-5}, 10^{-5})$ with initial time step of 0.5. *Broken lines* show contours at intervals of 0.05. **b** Erratic paths on surface V_{CM} to the saddle point $(-1, 0)$ from the starting point $(10^{-5}, 10^{-5})$ with initial time step of 5. *Broken lines* show contours at intervals of 0.05

behaves like a saddle point on the image surface (the Hessian of V_{CM} has both eigenvalues equal to 0.740208 at this point).

The common message of all of these results is that Snyman's minimisation algorithm started with a large initial time step is quick to find a region near a minimum point but is slower at converging to it. This is seen to be more true for the saddle point runs and is due to the fact that the image surface of the Cerjan-Miller surface is very flat near its minima at $(\pm 1, 0)$. So far this paper has demonstrated a new method for finding saddle points that is robust and can distinguish intrinsically between different kinds of critical points. To create a more efficient algorithm for finding saddle points one should combine the quickness of this method with a large intial time step for getting close to the required stationary point and the good merits of another method which converges rapidly to a stationary point when started close enough to it. To demonstrate the power of such a composite algorithm in a simple example the calculations in Table 5 were repeated with the modification that Newton's method would be used whenever the Hessian of V_{CM} had one positive and one negative eigenvalue and the modulus of g' was less than 10^{-1} and only in these circumstances. The results

are given in Table 6 and show that a much faster algorithm is obtained which is very efficient if the initial time step is large.

As a final comment, a unique path which leads from a minimum to a saddle point may be defined as one for which the principal velocities of the particle, corresponding to all of the eigenvalues of the Hessian matrix except the smallest, are zero. This path, which may be approximated by using a very small time step in the adapted method and starting the calculation as close as possible to a minimum, corresponds to the reaction path [14]. The actual path obtained in this manner would oscillate about the true reaction path; the amplitude depending upon the time step. This method in principle allows a reaction path calculation to be started from a minimum, but would require a very large number of iterations in view of the small time step required to ensure that the oscillation amplitude is small enough.

Acknowledgement. The author is indebted to The Japan Society for the Promotion of Science for the provision of a research fellowship.

References

1. McDougall JJW, Robb MA, Bernardi F (1986) Chem Phys Lett 129:595
2. Jensen HJAa, Jørgensen P, Helgaker T (1986) J Chem Phys 85:3917
3. Simons J, Jørgensen P, Taylor H, Ozment J (1983) J Phys Chem 87:2745
4. Crippen GM, Scheraga HA (1971) Arch Biochem Biophys 144:462
5. Cerjan CJ, Miller WH (1981) J Chem Phys 75:2800
6. Komornicki A, Ishida K. Morokuma K, Ditchfield R, Conrad M (1977) Chem Phys Lett 45:595
7. McIver Jr JW, Komornicki A (1972) J Am Chem Soc 94:2625
8. Snyman JA (1982) Appl Math Modelling 6:449
9. Hall GG, Okada M (1985) J Mol Struct (Theochem) 123:179
10. Snyman JA (1983) Appl Math Modelling 7:217
11. Snyman JA, Fatti LP (1987) J Optimisation Theory Appl 54:121
12. Hoffman DK, Nord RS, Ruedenberg K (1986) Theor Chim Acta 69:265
13. Jørgensen P, Jensen HJAa, Helgaker T (1988) Theor Chim Acta 73:55
14. Fukui K (1970) J Phys Chem 74:4161