Transition States and Reaction Paths

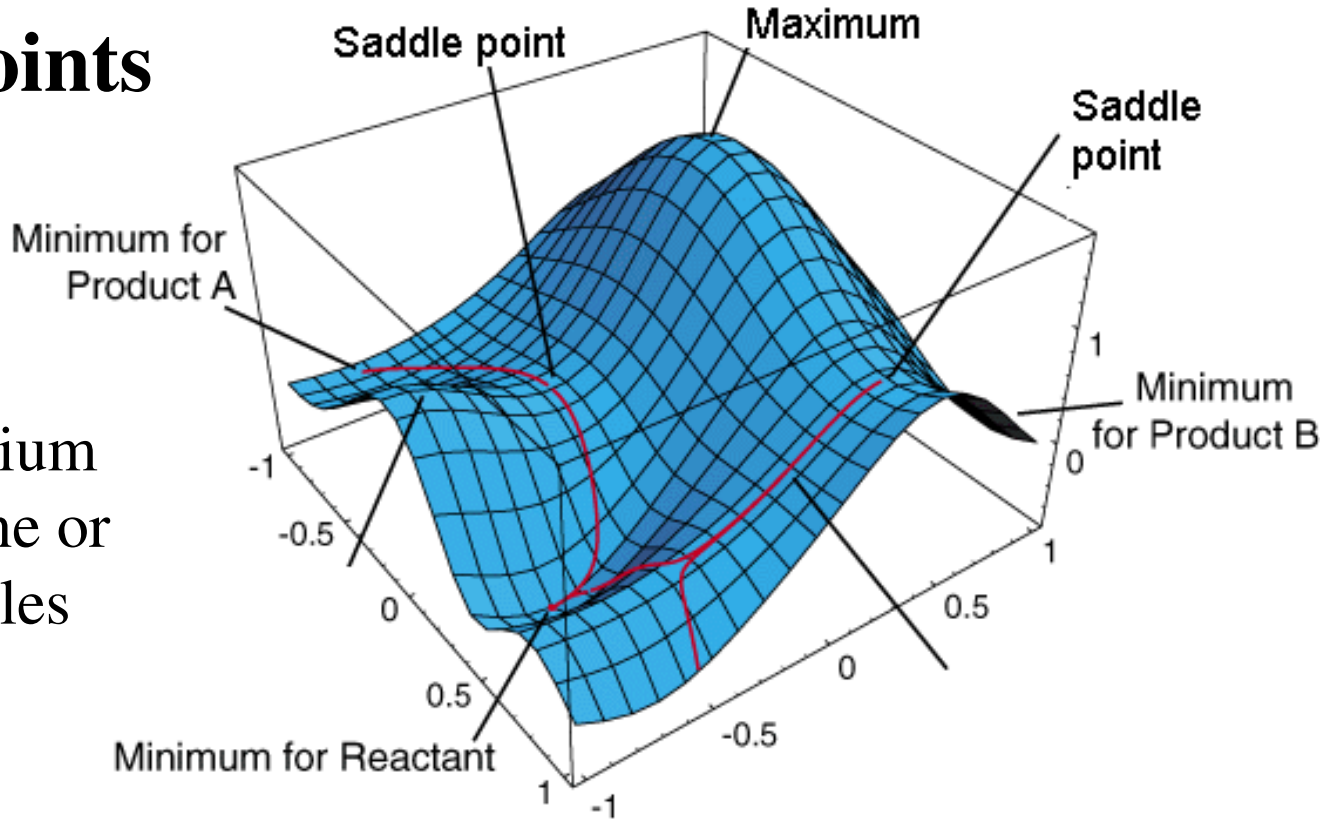
Computational Chemistry lab

2013

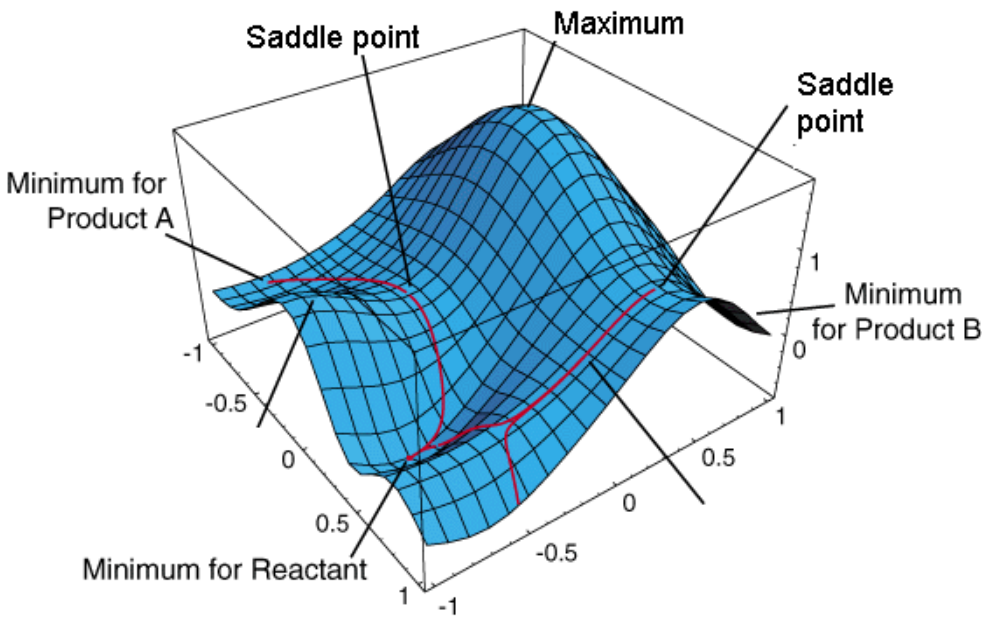
Potential Energy Surface (PES)

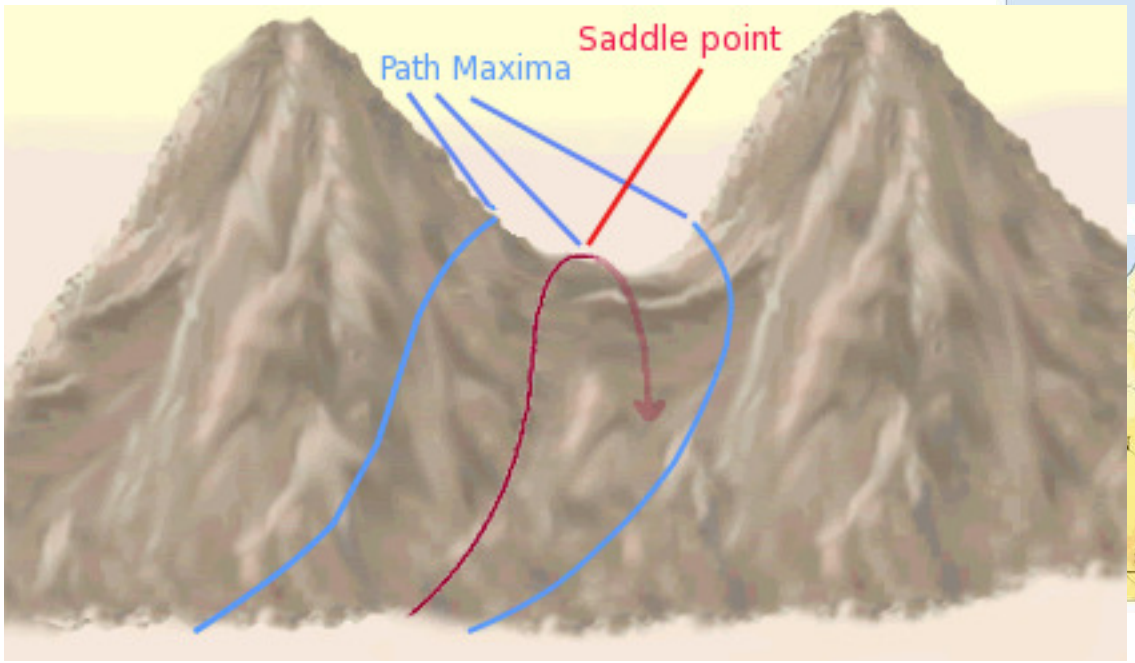
A 3N-6 -dimensional hypersurface in, where *N* is the number of atoms.

Saddle points



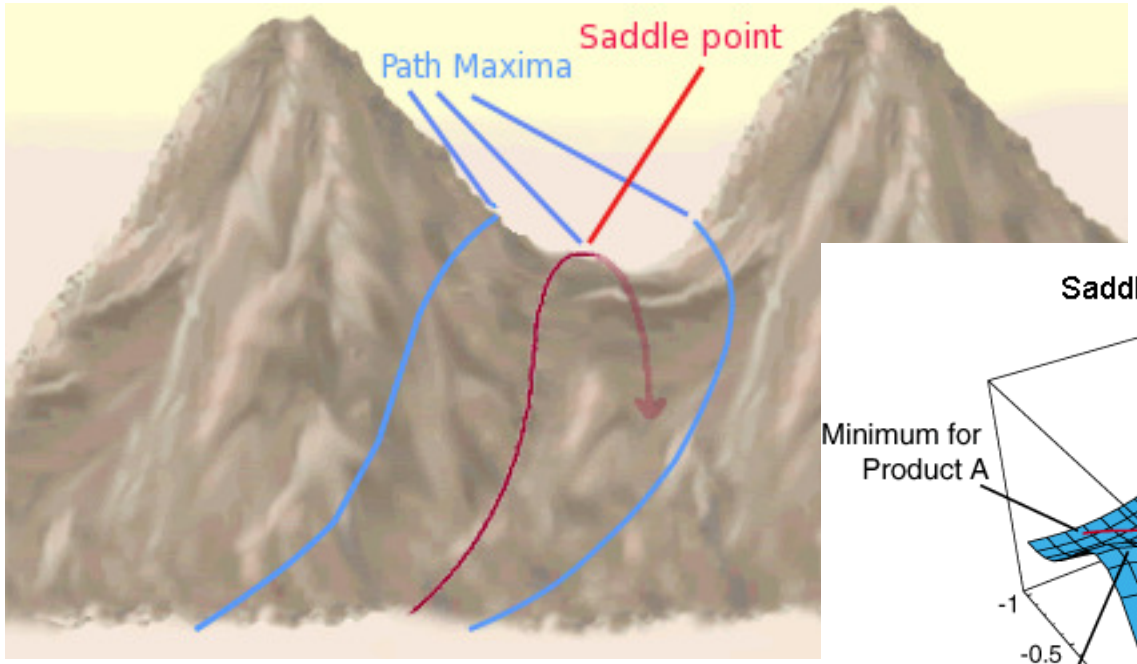
Minima– equilibrium structures of one or several molecules

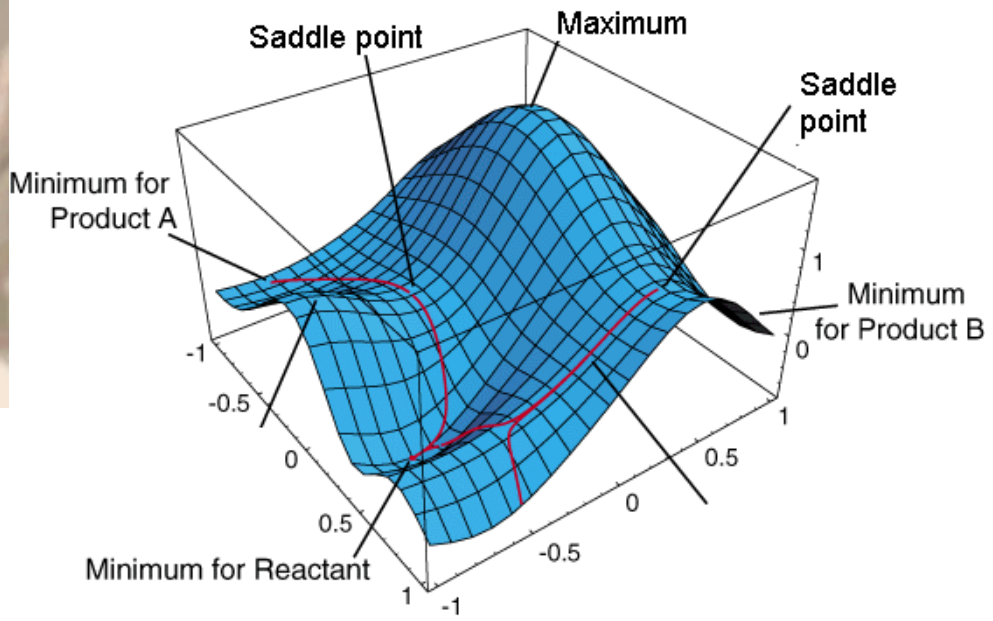

Saddle point
Maximum
Saddle point
Minimum for Product A
Minimum for Product B
Minimum for Reactant
-1
-0.5
0
0.5
1


Saddle point
Path Maxima


ISRAEL
Topographical map of
Central Mountain Range and
Northern Galilee Region.
Minimum
Saddle point
Saddle point
Mediterranean Sea
LEBANON
UPPER GALILEE
LOWER GALILEE
HAURAN
Sea of Kinneret
Jordan valley
Coastal Plain
Central Mountains
Transjordanian Plateau
Dead Sea
ARABAH
NEGEV

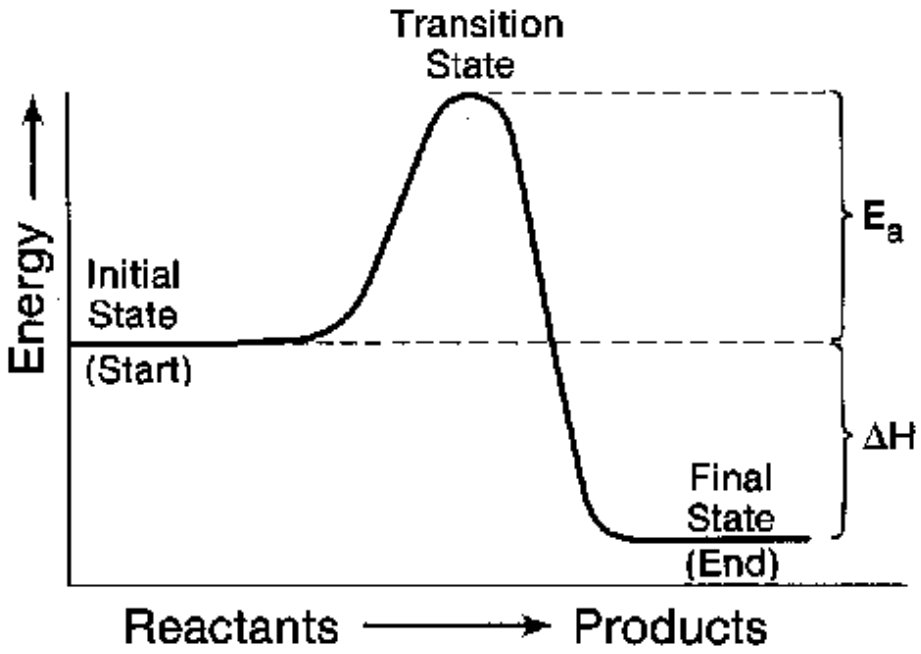
Saddle point - Transition state





Minimal maximum –
maximum in one direction, minimum in other directions

Transition State

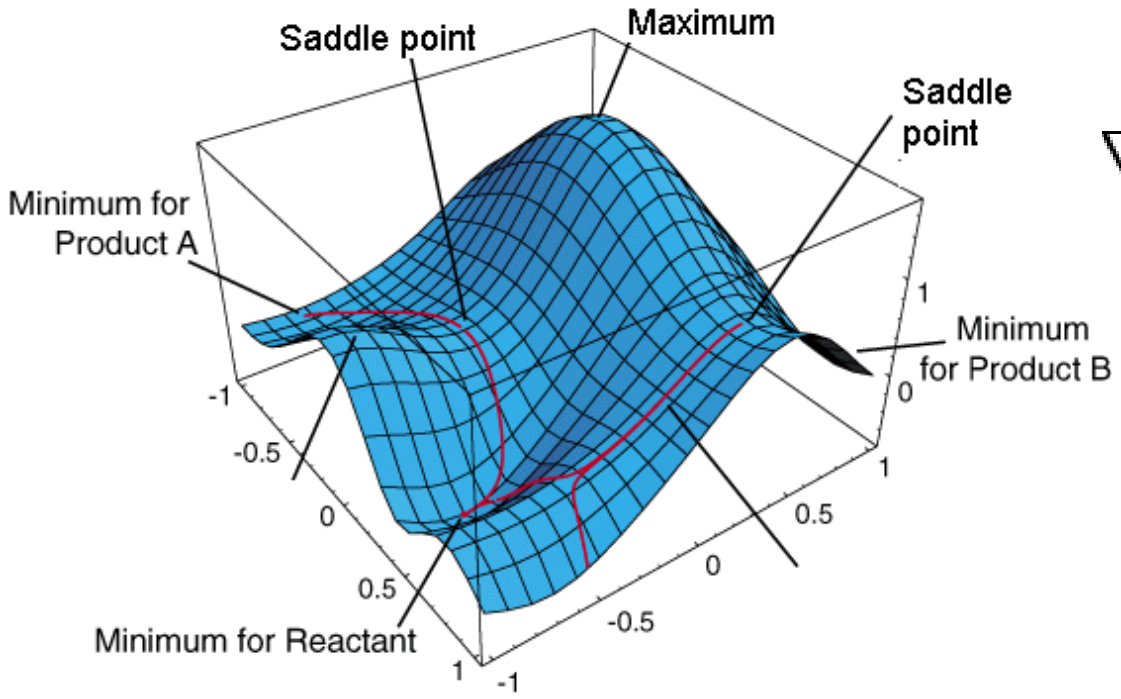


Arrhenius equation

$$k = Ae^{-E_A / k_b T}$$

k - rate constant
E_A – activation energy
T – temperature
k_b – Boltzmann constant

Multidimensional Optimization



$$\nabla f = \left(\frac{\partial f}{\partial x_1}, \ldots, \frac{\partial f}{\partial x_n} \right)$$

- gradient

The coordinates can be Cartesian (x,y,z..), or internal, such as bondlength and angle displacements)

Force: $F = -\nabla E$ (Potential energy)

$\nabla f = 0$ - Stationary Point (minimum, maximum, or saddle point)

Hessian matrix

A matrix of second-order derivatives of the energy with respect to atomic coordinates (e.g., Cartesian or internal coordinates)
Sometimes called force matrix – matrix size of (3N-6)x(3N-6)

$$H_{ij}(f) = \frac{\partial^2 f}{\partial x_i \partial x_j}$$

$$H(f) = \begin{bmatrix} \frac{\partial^2 f}{\partial x_1^2} & \frac{\partial^2 f}{\partial x_1 \partial x_2} & \cdots & \frac{\partial^2 f}{\partial x_1 \partial x_n} \\ \frac{\partial^2 f}{\partial x_2 \partial x_1} & \frac{\partial^2 f}{\partial x_2^2} & \cdots & \frac{\partial^2 f}{\partial x_2 \partial x_n} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\partial^2 f}{\partial x_n \partial x_1} & \frac{\partial^2 f}{\partial x_n \partial x_2} & \cdots & \frac{\partial^2 f}{\partial x_n^2} \end{bmatrix}$$

Quadratic approximation

Approximate the complex energy landscape by harmonic potentials

$$E(\vec{x}) = E(\vec{x}^{(st)}) + \cancel{\nabla E(\vec{x}^{(st)})(\vec{x} - \vec{x}^{(st)})} + \tfrac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{n} H_{ij}(\vec{x}^{(st)})(x_i - x_i^{(st)})(x_j - x_j^{(st)})$$

around a stationary point $\vec{x}^{(st)} = (x_1^{(st)}, \ldots x_n^{(st)})$ $\quad [\nabla E(\vec{x}^{(st)}) = 0]$

Quadratic approximation

Approximate the complex energy landscape by harmonic potentials

around a stationary point $\vec{x}^{(st)} = (x_1^{(st)},...x_n^{(st)}) \qquad [\nabla E(\vec{x}^{(st)}) = 0]$

$$E(x_1,...x_n) = E(x_1^{(st)},...x_n^{(st)}) + \tfrac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{n} H_{ij}(\vec{x}^{(st)})(x_i - x_i^{(st)})(x_j - x_j^{(st)})$$

Hessian matrix diagonalization (eigenproblem) $\varepsilon_k l_j^{(k)} = \sum_{i=1}^{n} H_{ij} l_i^{(k)}$

Eigenvalues $\varepsilon_k = m\omega_k^2$ ω_k - vibrational frequencies

Eigenvectors $l_j^{(k)}$ give normal coordinates $q_k = \sum_{i=1}^{n} l_i^{(k)}(x_i - x_i^{(0)})$

$$E(x_1,...x_n) = E(x_1^{(st)},...x_n^{(st)}) + \tfrac{1}{2}\sum_{k=1}^{n} \varepsilon_k q_k^2$$

Eigenvalues $\varepsilon_k = m\omega_k^2$ ω_k - vibrational frequencies

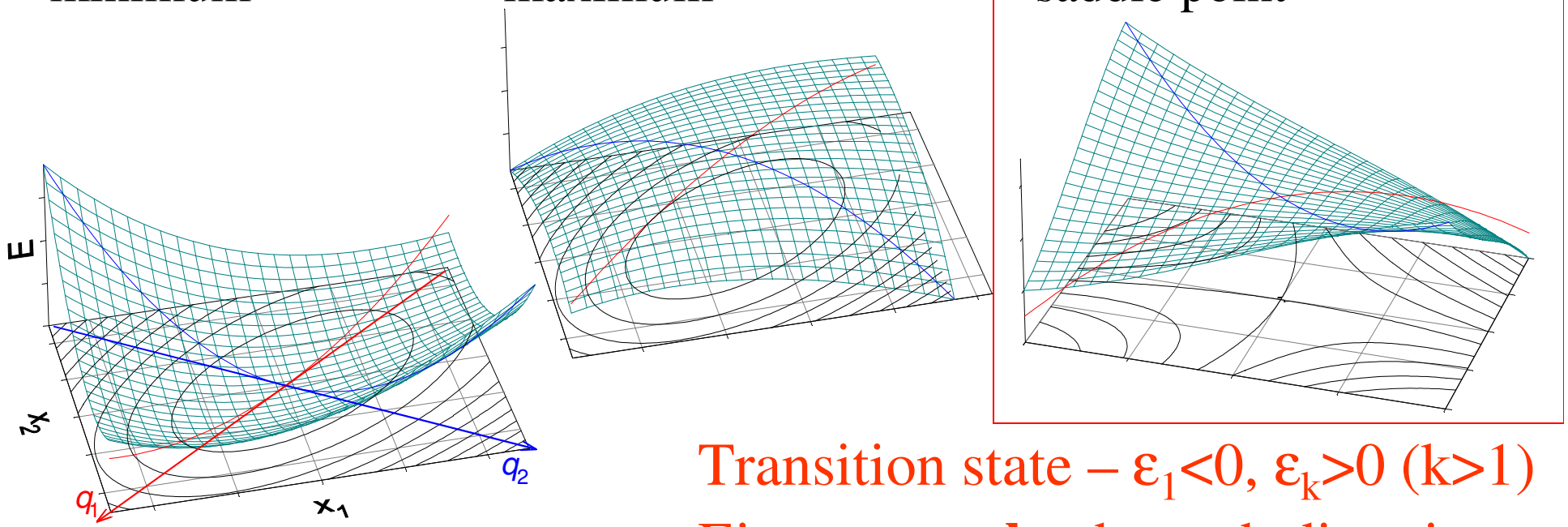
Eigenvectors $l_j^{(k)}$ give normal coordinates $q_k = \sum_{i=1}^{n} l_i^{(k)}(x_i - x_i^{(0)})$

$$E(x_1,\ldots x_n) = E(x_1^{(st)},\ldots x_n^{(st)}) + \tfrac{1}{2}\sum_{k=1}^{n} \varepsilon_k q_k^2$$

All $\varepsilon_k > 0 (\omega_k - \text{real})$ minimum

All $\varepsilon_k < 0 (\omega_k - \text{imaginary})$ maximum

$\varepsilon_k <> 0 (\omega_k - \text{imaginary/real})$ saddle point



Transition state – $\varepsilon_1<0$, $\varepsilon_k>0$ (k>1)

Eigenvector $\mathbf{l}_1$- the path direction

Algorithms for Finding Transition States

No general methods which are guarantied to find!

Global methods – interpolation between reactant and product

- Linear Synchronous Transit (LST)
- Quadratic Synchronous Transit (QST)

Local methods – augmented Newton-Raphson

- Eigenvector following
- Berny algorithm

Synchronous Transit-Guided Quasi-Newton (STQN) method – QST+quasi-Newton

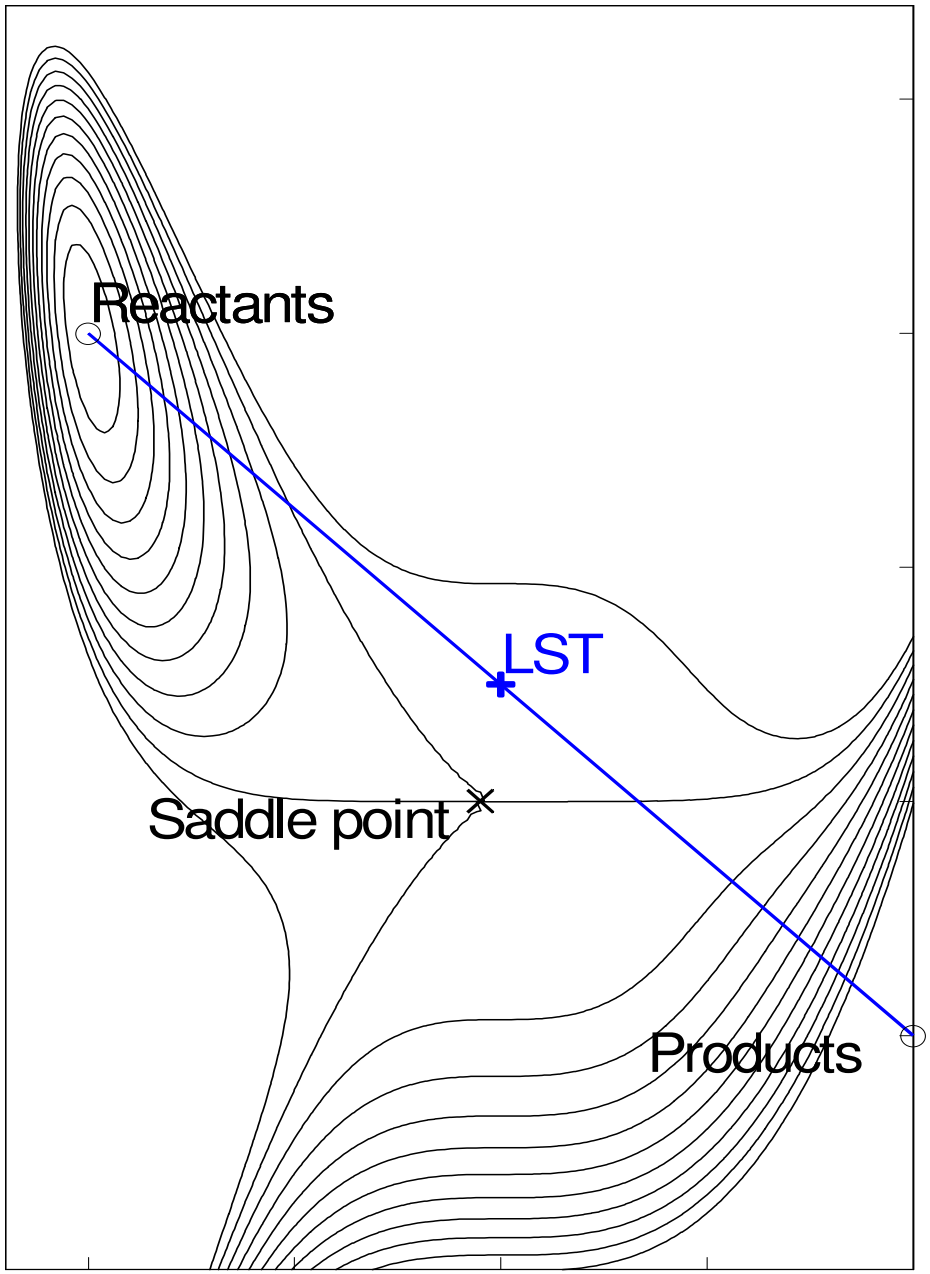
Force-field parameters in molecular mechanics are defined for equilibrium structures and can be inapplicable to transition structures.

Quantum calculations only

(recent semi-empirical potentials are applicable too)

Linear synchronous transit (LST)

-search for a maximum along a linear path between reactants and products

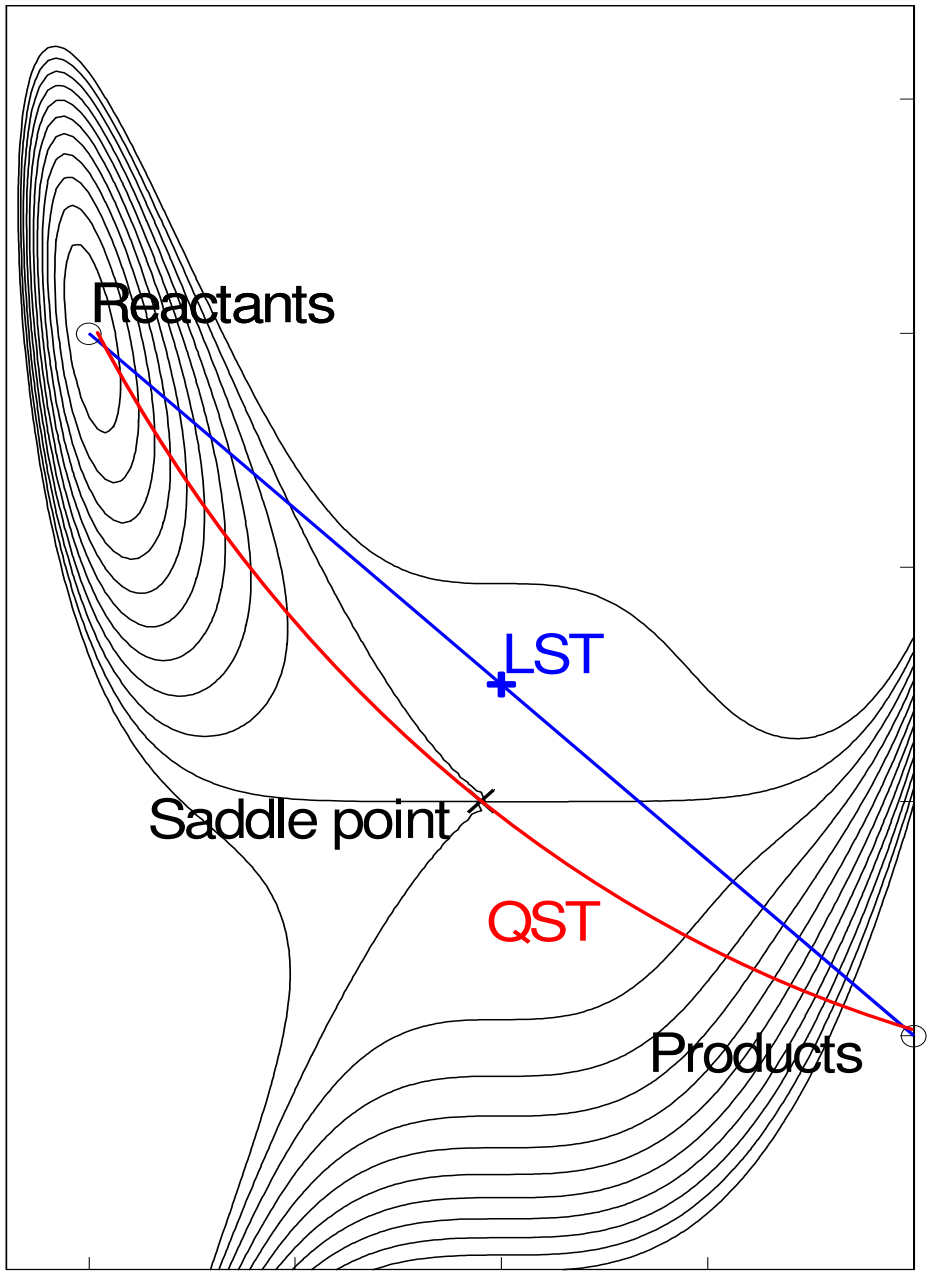


Linear synchronous transit (LST) - search for a maximum along a linear path between reactants and products

Quadratic synchronous transit (QST) - search for a maximum along an arc connecting reactants and products, and for a minimum in all directions perpendicular to the arc

Best case - the transition state is found.

General case – search is finished in a wrong saddle point or in a point with wrong number of negative eigenvalues (>1).



Newton(-Raphson) method:

Quadratic approximation : $E(\mathbf{R}) = E(\mathbf{R}_0) - \sum_{k=1}^{n} (F_k q_k - \frac{1}{2}\varepsilon_k q_k^2)$

$\mathbf{R}=(x_1,...x_n)$ - (3N-6) -dimensional vectors

q_k- normal coordinates, $\mathbf{R} - \mathbf{R}_0 = \sum_{k=1}^{n} q_k \mathbf{l}_k$, $\nabla E(\mathbf{R}_0) = -\sum_{k=1}^{n} F_k \mathbf{l}_k$

ε_k and $\mathbf{l}_k$ - eigenvalues and eigenvectors of the Hessian matrix at $\mathbf{R}_0$

Quasi-Newton – no explicit computation of the Hessian matrix.

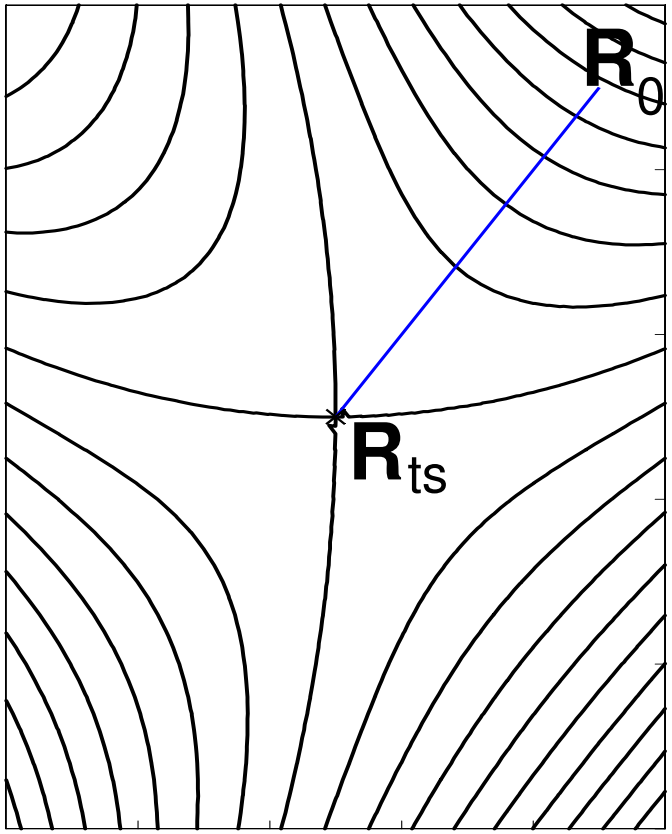
One-step search for quadratic functions $q_k = F_k / \varepsilon_k$

For arbitrary functions

$$\mathbf{R}_{i+1} = \mathbf{R}_i + \sum_{k=1}^{n} \mathbf{l}_k(\mathbf{R}_i) F_k(\mathbf{R}_i) / \varepsilon_k(\mathbf{R}_i)$$

Finds the closest stationary point (either minimum, maximum, or saddle point).

Transition state – $\varepsilon_1(\mathbf{R})<0$, $\varepsilon_k(\mathbf{R})>0$ $(k>1)$



Rational function optimization (RFO)

$$\mathbf{R}_{i+1} = \mathbf{R}_i + \sum_{k=1}^{n} \mathbf{l}_k(\mathbf{R}_i) \frac{F_k(\mathbf{R}_i)}{\varepsilon_k(\mathbf{R}_i) - \lambda_k}$$

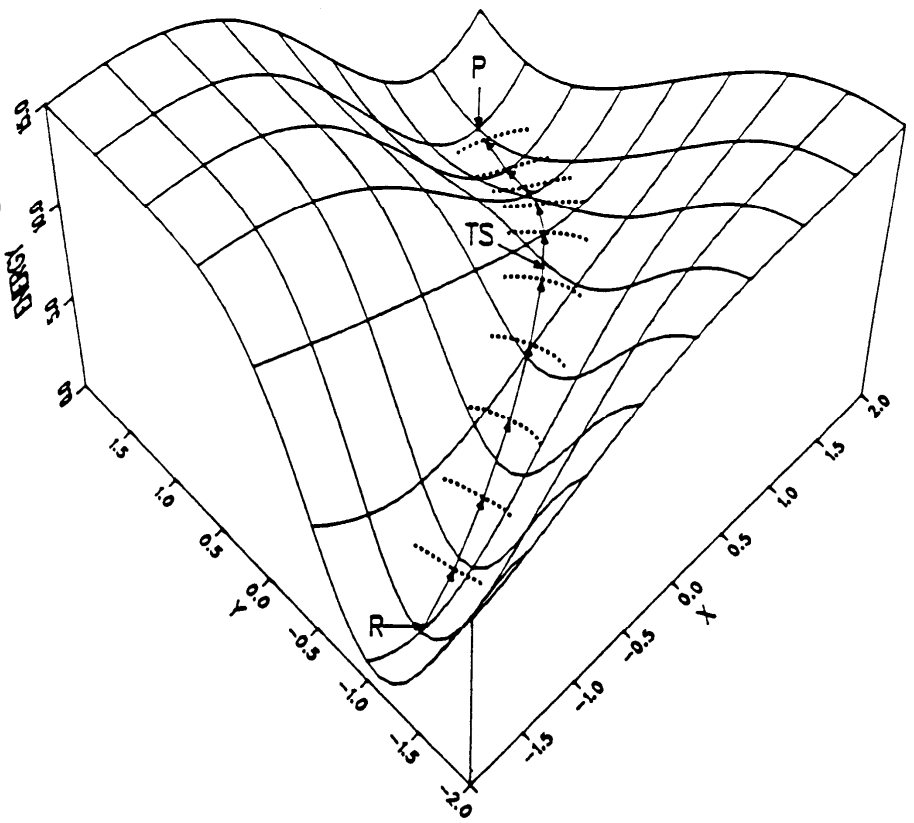
Berny algorithm (**Gaussian, Opt=TS**)

$$\varepsilon_1 = \min(\varepsilon_k) < \lambda_k \equiv \lambda < \varepsilon_2/2$$

Eigenvector Following Method (**Gaussian, Opt=EF; Hyperchem**)

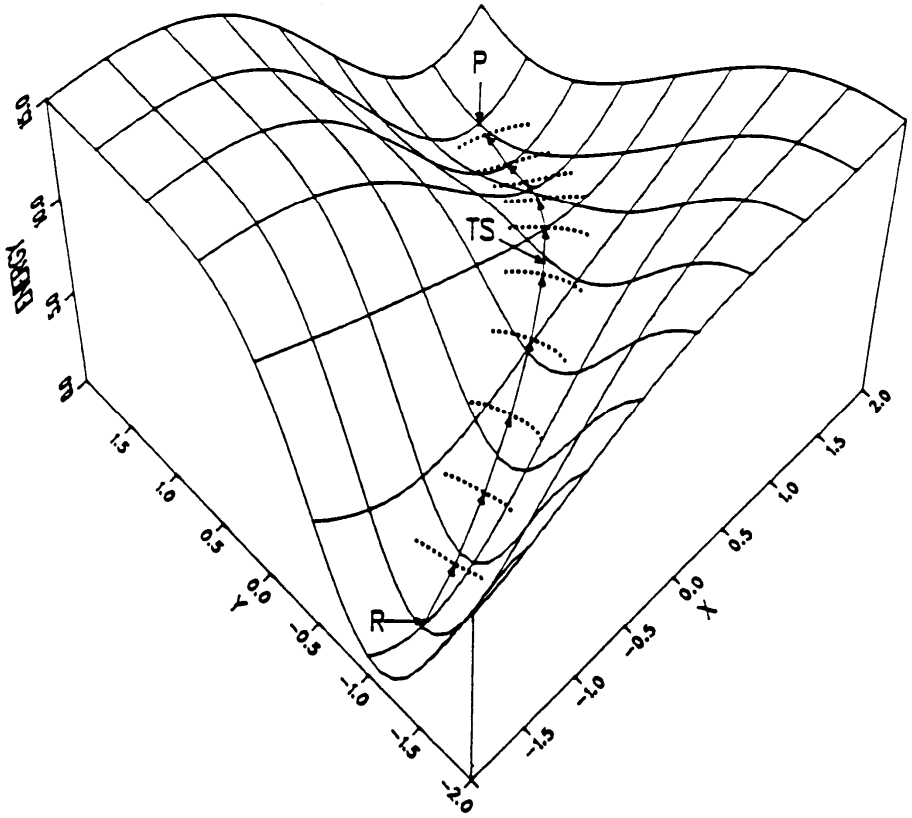
$$\lambda_1 = (\varepsilon_1 + \sqrt{\varepsilon_1^2 + 4F_1^2})/2$$

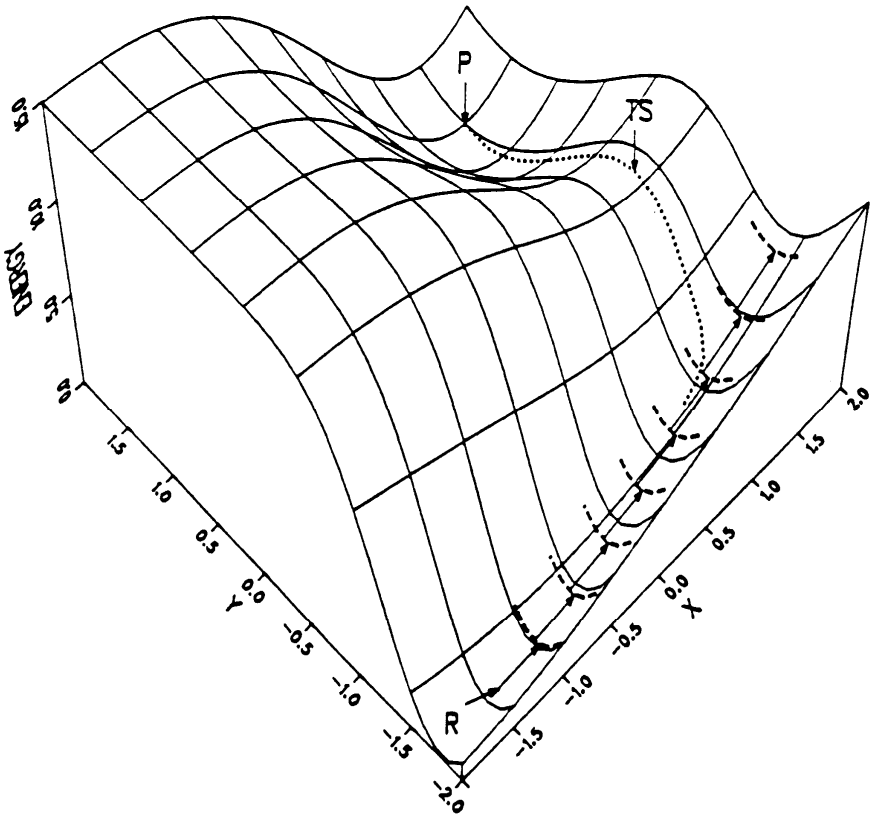
$$k > 1 : \lambda_k \equiv \lambda = \sum_{i>1} F_i^2 / (\lambda - \varepsilon_i)$$



Clamber “up valley”

Clamber “up valley”





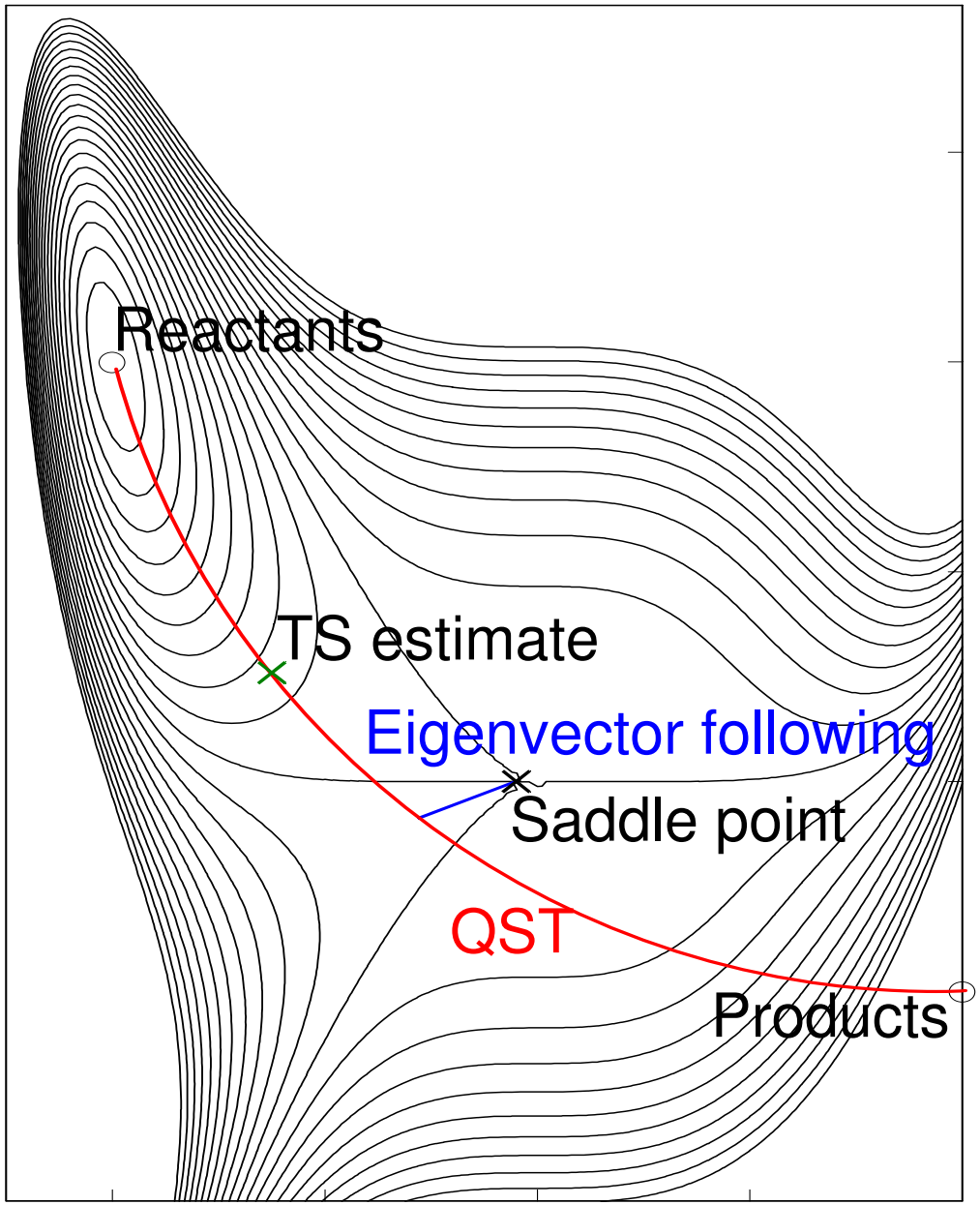
Synchronous Transit-Guided Quasi-Newton (STQN)

approach to the quadratic region - quadratic synchronous transit

complete the optimization- eigenvector-following algorithm

Gaussian, Opt=QST3: input reactants, products, and estimate of transition state

Gaussian, Opt=QST2; Hyperchem: input reactants and products only, automatic estimate of transition state



Is the correct TS found?

Look at the transition state geometry to make sure it's the right one.

Use several algorithms.

Try several estimates of transition state.

Follow reaction path to be sure that the transition state connects the correct reactants and products.

Reaction Paths

Steepest descent path from transition state to reactants and products

IRC-internal reaction coordinate

Gaussian - IRC

New point $\mathbf{x}_{k+1}$ – minimization on the hypersphere surface