Rare Events
&
Nudged Elastic Band method

Carlo Sbraccia (Princeton University)
and
Matteo Cococcioni (MIT)

Minneapolis, 30/05/2006
Potential Energy Landscape

Classical system of \( N \) particles:

\[
H(p^{3N}, q^{3N}) = \sum_{i=1}^{3N} \frac{p_i^2}{2m_i} + U(q_1, \ldots, q_{3N})
\]

The potential energy function \( U(q^{3N}) \) determines the structure, the dynamics, and the thermodynamics of the system:

- stable configurations occur at the minima of \( U(q^{3N}) \), defining the possible structures of the system;

- \(-\nabla U(q^{3N})\) gives the force acting on the system, determining its dynamics via integration of the Newton's equations of motion;

- the partition function, \( Z \), which determines the thermodynamics of the system, is also a functional of \( U(q^{3N}) \).
Analogous statements can be made for a quantum mechanical system within the Born-Oppenheimer approximation:

- the potential energy function $U(q^{3N})$ corresponds to the electronic ground-state energy, solution of the Schroedinger equation at frozen nuclear positions.
Let's consider a potential energy function with two minima $\{q_A^{3N}\}$ and $\{q_B^{3N}\}$.

A two-dimensional example:
The probability of finding the system in the neighbourhood of any configuration \(\{q^{3N}\}\) is proportional to the Boltzmann factor computed at such configuration:

\[
P(q^{3N}) \sim \frac{e^{-\beta U(q^{3N})}}{\mathcal{Z}}
\]
Equivalently (ergodicity) the relative time spent around a configuration \( \{q^{3N}\} \) is proportional to the Boltzmann factor.

The system spends more time around configurations of low potential energy. Fluctuations are responsible for the transition between different minima.
rare events

Brownian motion in a double-well potential at a temperature $K_B T \sim 0.5 \ E_A$ (50% of the barrier height):

diffusive behaviour
rare events

Brownian motion in a double-well potential at a temperature $k_B T \sim 0.2 E_A$ (20% of the barrier height):
rare events

Brownian motion in a double-well potential at a temperature $K_B T \sim 0.08 E_A$ (8% of the barrier height):

“instantonic” behaviour
rare events

\[ E_A \gg K_B T \]

\[ \Delta E^+ \]
\[ \Delta E^- \]

\[ K_B T \]

\[ \tau_A \sim e^{\beta \Delta E^+} \]
\[ \tau_B \sim e^{\beta \Delta E^-} \]
\[ \frac{\tau_A}{\tau_B} = e^{\beta (E_B - E_A)} \]
rare events

What is the characteristic time scale of this transition process?

Energy

saddle point

$E_A \gg k_B T$

$E_A$

$K_B T$

reactants

products

reaction coordinate
What is the characteristic time scale of this transition process?

Van't-Hoff - Arrhenius (1890)

\[ t_{\text{jump}} \sim t_{\text{vib}} \cdot e^{\frac{E_A}{k_B T}} \]
What is the characteristic time scale of this transition process?

Van't-Hoff - Arrhenius (1890)

\[ t_{\text{jump}} \sim t_{\text{vib}} \cdot e^{\frac{E_A}{K_BT}} \]

\[ t_{\text{vib}} \sim 10^{-13} \text{ s} \]

\[ E_A = 0.75 \text{ eV} \]

\[ T = 300 \text{ K} \]

\[ t_{\text{jump}} \sim 1 \text{ s} \]
rare events

Assuming a time step of one femtosecond, $\sim 10^{15}$ steps of Molecular Dynamics are necessary to have a sizeable probability of observing at least a transition from reactants to products!
Assuming a time step of one femto-second, $\sim 10^{15} \text{ steps}$ of Molecular Dynamics are necessary to have a sizeable probability of observing at least a transition from reactants to products!

Nevertheless, when an appropriate fluctuation occurs, the process is extremely fast:

$$t \sim \text{femto-seconds} \ (10^{-15} \text{ s})$$
What is macroscopically perceived as a slow process is instead a rare event.
Alternative approach:

The transition rate can be estimated using equilibrium statistical mechanics:

- once the saddle point has been identified we can use harmonic Transition State Theory (hTST) to compute the rate constant:

\[
k_{\text{reactants} \rightarrow \text{products}} = A \cdot e^{-\frac{E_A}{k_B T}}
\]

\[
A = \frac{\prod_{i=1}^{3N} \nu_{i_{\text{reactants}}}}{\prod_{i=1}^{3N-1} \nu_{i_{\text{saddle point}}}}
\]
rare events

The goal is locating all the relevant saddle points, but:

\[ E_A > K_B T \]
rare events

The goal is locating all the relevant saddle points, but:

saddle points are unstable

their direct location is a rather difficult task
saddle points in multidimensional systems: the Mueller PES
saddle points in multidimensional systems: the Mueller PES
saddle points in multidimensional systems: the Mueller PES
saddle points in multidimensional systems: the Mueller PES

The path characterized by the "highest" transition probability, at zero temperature, is the **Minimum Energy Path**. **MEP**: the components of the force orthogonal to the path are zero.

\[ \nabla V(x(s)) - \tau(s) (\tau(s) \nabla V(x(s))) = 0 \]

**normalised tangent**
saddle points in multidimensional systems: the Mueller PES

The path characterized by the "highest" transition probability, at zero temperature, is the Minimum Energy Path.

**MEP:** the components of the force orthogonal to the path are zero.

\[ \nabla V(x(s)) - \tau(s) \left( \tau(s) \nabla V(x(s)) \right) = 0 \]
saddle points in multidimensional systems: the Mueller PES

The path characterized by the “highest” transition probability, at zero temperature, is the Minimum Energy Path.

**MEP:** the components of the force orthogonal to the path are zero.

\[ \nabla V(x(s)) - \tau(s) \left( \tau(s) \nabla V(x(s)) \right) = 0 \]
saddle points in multidimensional systems: the Mueller PES

The path characterized by the "highest" transition probability, at zero temperature, is the Minimum Energy Path.

**MEP:** the components of the force orthogonal to the path are zero.

\[ \nabla V(x(s)) - \tau(s) \left( \tau(s) \nabla V(x(s)) \right) = 0 \]
saddle points in multidimensional systems: the Mueller PES

The path characterized by the "highest" transition probability, at zero temperature, is the Minimum Energy Path. **MEP:** the components of the force orthogonal to the path are zero.

The **MEP** crosses the saddle points.
how to locate the MEP

1) Path discretisation
   (“chain of images”):

   \[ s \rightarrow i \cdot \delta s \]

   \[ x(s) \rightarrow x_i \]

   \[ \tau(s) \rightarrow \tau_i = \frac{x_{i+1} - x_{i-1}}{|x_{i+1} - x_{i-1}|} \]
1) Path discretisation
   ("chain of images") :

   \[ s \rightarrow i \cdot \delta s \]
   \[ x(s) \rightarrow x_i \]
   \[ \tau(s) \rightarrow \tau_i = \frac{x_{i+1} - x_{i-1}}{|x_{i+1} - x_{i-1}|} \]

2) Orthogonal forces :

   \[ F(x_i)_{\perp} = -\left[ \nabla V(x_i) - \tau_i (\tau_i |\nabla V(x_i)|) \right] \]
how to locate the MEP

1) Path discretisation
   ("chain of images"):
   \[ s \rightarrow i \cdot \delta s \]
   \[ x(s) \rightarrow x_i \]
   \[ \tau(s) \rightarrow \tau_i = \frac{x_{i+1} - x_{i-1}}{|x_{i+1} - x_{i-1}|} \]

2) Orthogonal forces:
   \[ F(x_i)_\perp = -[\nabla V(x_i) - \tau_i (\tau_i |\nabla V(x_i)|)] \]

3) MEP condition:
   \[ ||F(x_i)_\perp|| = 0 \]
how to locate the MEP

1) Path discretisation
   ( "chain of images" ) :
   \[ s \rightarrow i \cdot \delta s \]
   \[ x(s) \rightarrow x_i \]
   \[ \tau(s) \rightarrow \tau_i = \frac{x_{i+1} - x_{i-1}}{|x_{i+1} - x_{i-1}|} \]

2) Orthogonal forces :
   \[ F(x_i)_\perp = - [\nabla V(x_i) - \tau_i (\tau_i |\nabla V(x_i))]| \]

3) MEP condition :
   \[ \| F(x_i)_\perp \| = 0 \]

4) path dynamics
   ( steepest-descent ) :
   \[ x_i^{k+1} = x_i^k + \lambda F(x_i^k)_\perp \]
how to locate the MEP

1) Path discretisation
   ("chain of images"):
   \[ s \rightarrow i \cdot \delta s \]
   \[ x(s) \rightarrow x_i \]
   \[ \tau(s) \rightarrow \tau_i = \frac{x_{i+1} - x_{i-1}}{|x_{i+1} - x_{i-1}|} \]

2) Orthogonal forces:
   \[ F(x_i)_\perp = -[\nabla V(x_i) - \tau_i (\tau_i |\nabla V(x_i)))] \]

3) MEP condition:
   \[ \|F(x_i)_\perp\| = 0 \]

4) Path dynamics
   (steepest-descent):
   \[ x_{i}^{k+1} = x_{i}^{k} + \lambda F(x_{i}^{k})_{\perp} \]

5) Alternatively, Broyden acceleration:
   \[ x_{i}^{k+1} = x_{i}^{k} + J^{-1} F(x_{i}^{k})_{\perp} \]
sliding down

The path dynamics does not preserve the inter-image distance (path's parametrisation):

\[ L' = \sqrt{L^2 + \Delta x^2} \]

\[ L' - L = L \left( \sqrt{1 + \frac{\Delta x^2}{L^2}} - 1 \right) \approx \frac{1}{2} \frac{\Delta x^2}{L} \]
The path dynamics does not preserve the inter-image distance (path's parametrisation):

**Consequences:**

1) Many images are required to represent the path.

\[
L' = \sqrt{L^2 + \Delta x^2}
\]

\[
L' - L = L \left( \sqrt{1 + \frac{\Delta x^2}{L^2}} - 1 \right) \approx \frac{1}{2} \frac{\Delta x^2}{L}
\]
The path dynamics does not preserve the inter-image distance (path's parametrisation):

**Consequences:**

1) Many images are required to represent the path.

2) The images can eventually slide down to the two minima.

\[
L' = \sqrt{L^2 + \Delta x^2}
\]

\[
L' - L = L \left( \sqrt{1 + \frac{\Delta x^2}{L^2}} - 1 \right) \approx \frac{1}{2} \frac{\Delta x^2}{L}
\]
The path dynamics does not preserve the inter-image distance (path's parametrisation):

**Consequences:**

1) Many images are required to represent the path.

2) The images can eventually slide down to the two minima.

Possible solutions:

1) **NEB**: the images are connected by springs.

2) **STRING**: images are kept equispaced using Lagrange constraints.
Nudged Elastic Band method

Subsequent images of the chain are connected by springs (to enforce continuity).
Nudged Elastic Band method

➔ Subsequent images of the chain are connected by springs (to enforce continuity).

➔ Each image feels forces due to external potential + springs.
**Nudged Elastic Band method**

- Subsequent images of the chain are connected by springs (to enforce continuity).

- Each image feels forces due to **external potential + springs**.

- NEB idea [1,2]: elastic forces are projected along the path and external forces are projected orthogonally to the path.


Nudged Elastic Band method

➔ Subsequent images of the chain are connected by springs (to enforce continuity).

➔ Each image feels forces due to external potential + springs.

➔ NEB idea [1,2]: elastic forces are projected along the path and external forces are projected orthogonally to the path.

➔ Projections are defined by the path's tangent: the tangent definition plays a crucial role.

NEB on the Mueller PES

Initial guess: linear interpolation between the two end points.

\[ \vec{x}_i = \vec{x}_0 + \frac{i}{N} (\vec{x}_N - \vec{x}_0) \]
NEB on the Mueller PES

minimum energy path on the mueller PES

NEB
Improvements

1) Higher resolution around the saddle point: variable elastic constants.

2) Accurate identification of saddle point: climbing image.
Variable elastic constants

- The user specifies the minimum ($k_{\text{min}}$) and the maximum ($k_{\text{max}}$) values of the elastic constants.

- Spring constants can be chosen so that springs are stiffer where the potential energy is higher ($k_{\text{max}} > k_{\text{min}}$): higher resolution around saddle points.

- The value of the elastic constant for each image $x_i$ is obtained by interpolating between $k_{\text{min}}$ and $k_{\text{max}}$:

$$k_i = \frac{1}{2} \left( k_{\text{max}} + k_{\text{min}} - (k_{\text{max}} - k_{\text{min}}) \cos \left( \pi \frac{(V(x_i) - V_{\text{min}})}{(V_{\text{max}} - V_{\text{min}})} \right) \right)$$
Climbing image

A given image \((x_i)\) can be made "to climb" up-hill the PES once the climbing direction is specified.

In the CI scheme the direction is given by the path's tangent.

\[
F(x_{i_{max}}) = -\nabla V(x_{i_{max}}) + 2 \tau_{i_{max}} \left( \tau_{i_{max}} | \nabla V(x_{i_{max}}) \right)
\]

- The image can be automatically chosen during the optimisation as the one with the highest energy (CI_scheme="auto").
- One or more images can be forced to climb (CI_scheme="manual").
- Climbing Image should be used after some optimisation steps.
NEB: input variables

A detailed explanation of all the keywords can be found in the file Doc/INPUT_PW.

&CONTROL
  calculation = "neb" <= mandatory
...  
nstep <= optional (0)
... /
...
...
&IONS
  num_of_images <= mandatory
  CI_scheme <= optional (no-CI)
  opt_scheme <= optional (quick-min)
  ds <= optional (1.0)
  first_last_opt <= optional (.FALSE.)
  k_max <= optional (0.1)
  k_min <= optional (0.1)
  path_thr <= optional (0.05)
... /

NEB: input variables

A detailed explanation of all the keywords can be found in the file Doc/INPUT_PW.

first_image
   X 0.0 0.0 0.0 { if_pos(1) if_pos(2) if_pos(3) }
   Y 0.5 0.0 0.0 { if_pos(1) if_pos(2) if_pos(3) }
   Z 0.0 0.2 0.2 { if_pos(1) if_pos(2) if_pos(3) }

intermediate_image 1
   X 0.0 0.0 0.0
   Y 0.9 0.0 0.0
   Z 0.0 0.2 0.2

intermediate_image ...
   X 0.0 0.0 0.0
   Y 0.9 0.0 0.0
   Z 0.0 0.2 0.2

last_image
   X 0.0 0.0 0.0
   Y 0.7 0.0 0.0
   Z 0.0 0.5 0.2

<= mandatory
<= optional
<= mandatory
NEB: output

Files in the working directory (./):

- `prefix.path` <= file containing data required to restart a NEB calculation
- `prefix.axsf` <= file containing the path in `xcrysden` format
- `prefix.xyz` <= file containing the path in `xyz` format
- `prefix.dat` <= file containing the reaction coordinate, the energy and the error of each image
- `prefix.int` <= file containing a cubic interpolation for the energy profile
**NEB: output**

**Files in the working directory (./):**

- `prefix.path` <= file containing data required to restart a NEB calculation
- `prefix.axsf` <= file containing the path in *xcrysden* format
- `prefix.xyz` <= file containing the path in *xyz* format
- `prefix.dat` <= file containing the reaction coordinate, the energy and the error of each image
- `prefix.int` <= file containing a cubic interpolation for the energy profile

![Diagram](image)
Example 17: collinear proton transfer
plain NEB
Example 17: collinear proton transfer

variable elastic constants

\[ \text{H} - \text{H} \quad \rightarrow \quad \text{H} - \text{H} \]

![Graph showing energy vs. reaction coordinate](image.png)
Example 17: collinear proton transfer
climbing image (manual on image 5)
Notes on parallel execution

PWscf has two levels of parallelisation plus one specific for NEB:

1) \textbf{R&G} : wave-functions are distributed among the CPUs so that each CPU works on a subset of plane-waves. The same is done on real-space grid-points. By default all the CPUs are used for this parallelisation scheme. Example with 8 CPUs:

```
prompt> mpirun -np 8 pw.x -in input_file > output_file
```

2) \textit{k-points} : k-points (when present) are distributed among pools of CPUs. Each pool can contain one or more CPUs. In this latter case \textbf{R&G} parallelisation is used within the pool. This scheme is selected by specifying in the command line the required number of k-points pools. Example with 8 CPUs and 2 k-points pools:

```
prompt> mpirun -np 8 pw.x -npool 2 -in input_file > output_file
```

3) \textbf{NEB-images} : NEB images are distributed among pools of CPUs. Within each image \textbf{R&G} and \textit{k-points} parallelisation schemes can also be used. This scheme is selected by specifying in the command line the required number of images pools. Example with 8 CPUs, 2 k-points pools and 2 images pools:

```
prompt> mpirun -np 8 pw.x -npool 2 -nimage 2 -in input_file > output_file
```
Notes on parallel execution

- High scalability of the memory usage.
- High intra-pool communication.
- Good work-load balance among CPUs.
- Best choice. Bad scaling when number of plane-waves per CPU is very small.

- NO scalability of the memory usage.
- Low inter-pool communication.
- Work-load among pools can result to be unbalanced.
- Good when npool is a whole divisor of the number of k-points.

- NO scalability of the memory usage.
- Extremely low inter-pool communication.
- Work-load among pools can result to be unbalanced.
- Good for paths with several images.