



# L11 – Statistical Mechanics 3

Statistical errors and spectrum simulations

# Statistical errors in molecular dynamics

Accurate and precise



Accurate but not precise



Low statistics

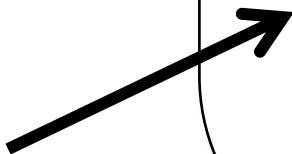
Precise but not accurate



Not precise and not accurate



Bad Hamiltonian  
Too large time steps



Consider the following example.

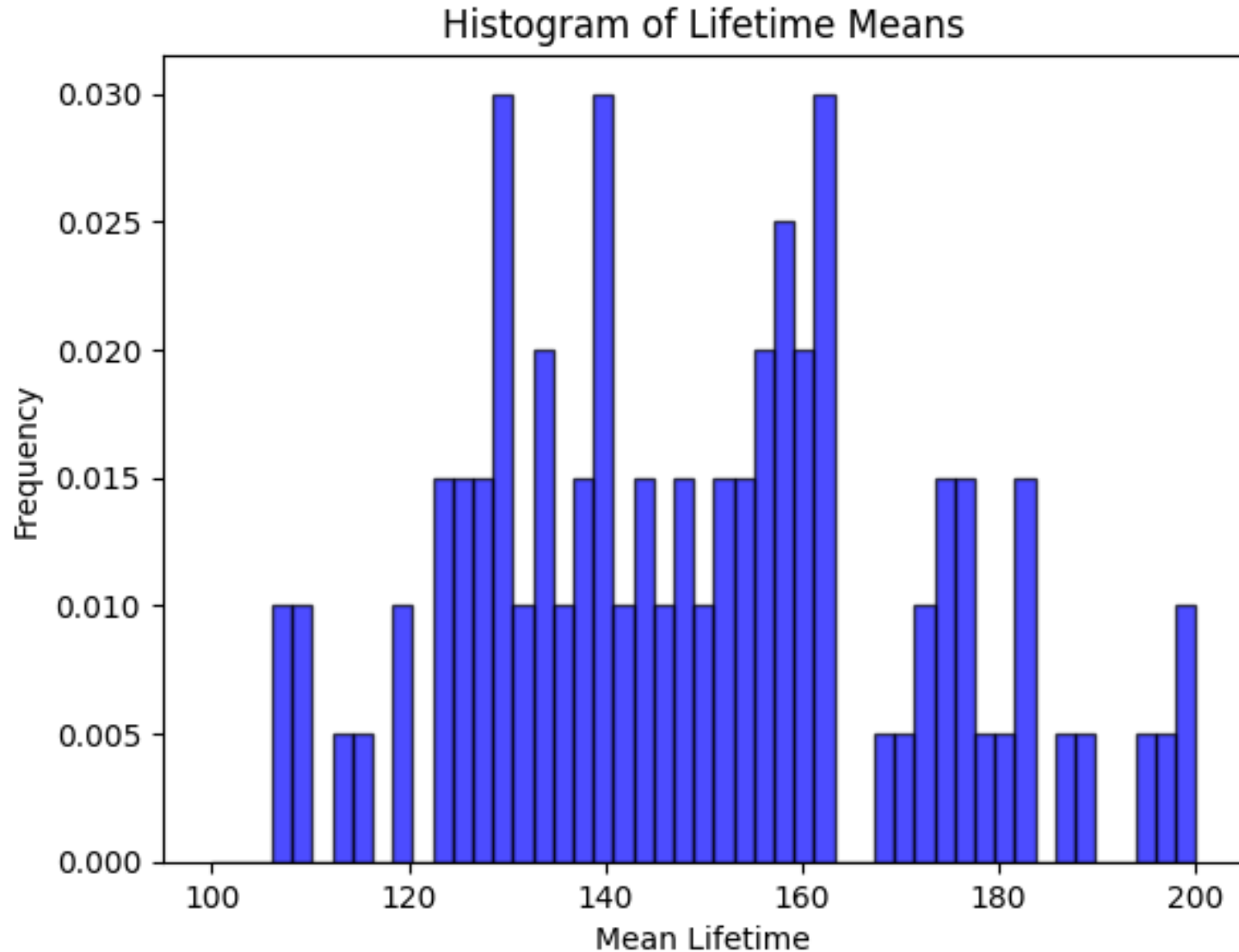
We ran 100 trajectories starting from an isomer C and find:

<b>Channel</b>	<b>Proportion of trajectories (%)</b>
No isomerization	5
Isomer A	52
Isomer B	43

The **mean value** of the isomerization time into isomer B was 150 fs with a **standard deviation** of also 150 fs.

**What is the margin of error (precision) for these numbers?**

The **mean value** of the isomerization time into isomer B was 150 fs with a **standard deviation** of 150 fs.



Suppose you repeat the simulation 100 times. Each time, the mean value will be a bit different.

# Margin of error for a population mean

If  $\mu_a$  is the mean value of the repetitions' means, the mean value of each repetition will be within the  $\mu_a - \varepsilon_a$  and  $\mu_a + \varepsilon_a$  in  $x\%$  of times for

$$\varepsilon_a = z(x\%) \frac{\sigma}{\sqrt{N_a}}$$

- $\sigma$  is the population standard deviation for process  $a$
- $N_a$  is the population size of the process  $a$
- $z(95\%) = 1.96$  (confidence interval)

For the 43 trajectories returning to the ground state, the **mean value** of the hopping time was 150 fs with a **standard deviation** of 150 fs

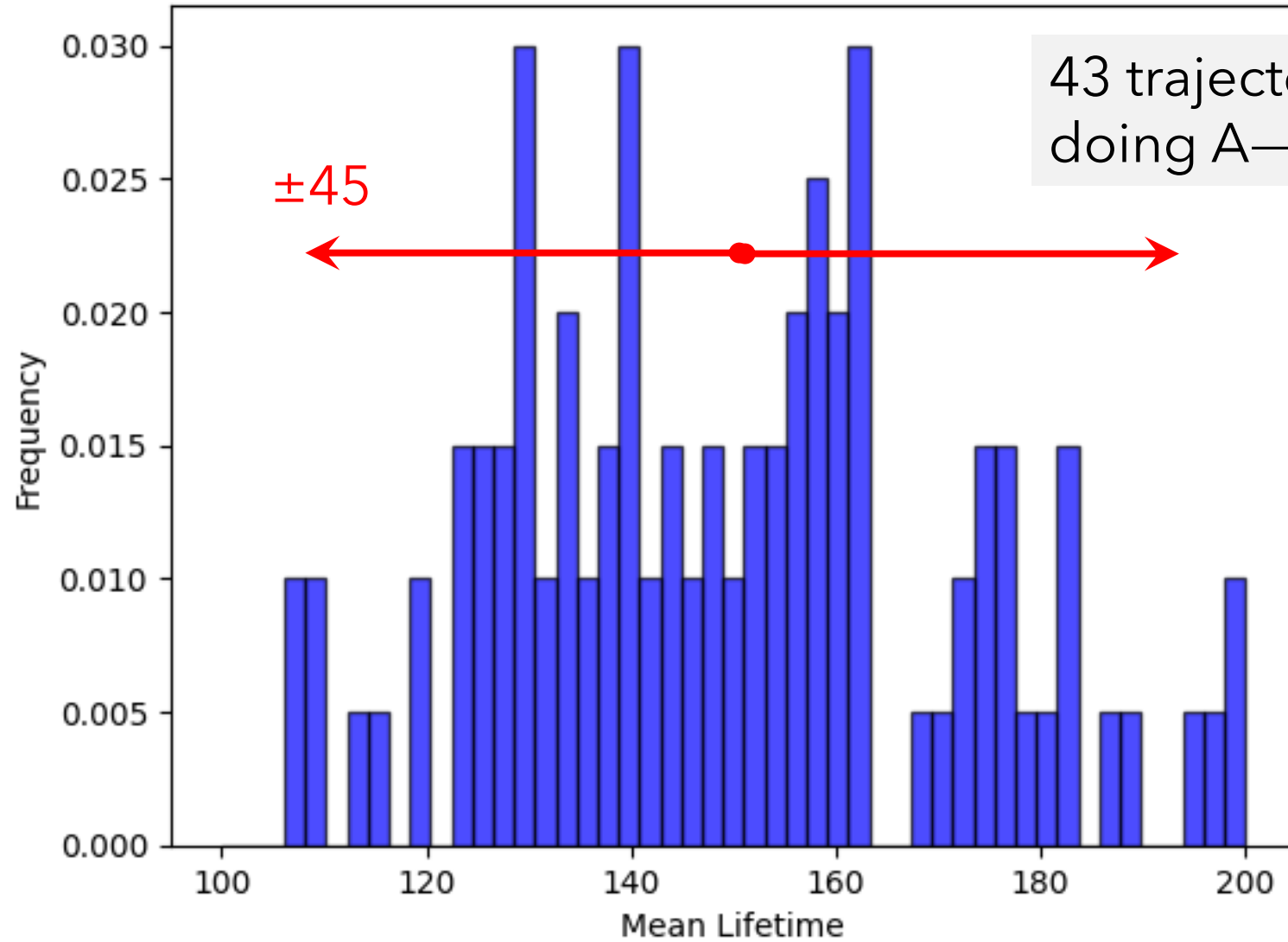
For 95% confidence, the margin of error of the mean A→B isomerization time is

$$\varepsilon_{C \rightarrow B} = 1.96 \frac{150}{\sqrt{43}} = 45$$

Isomerization B time:

$$\tau_{C \rightarrow B} = 150 \pm 45 \text{ fs}$$

Histogram of Lifetime Means

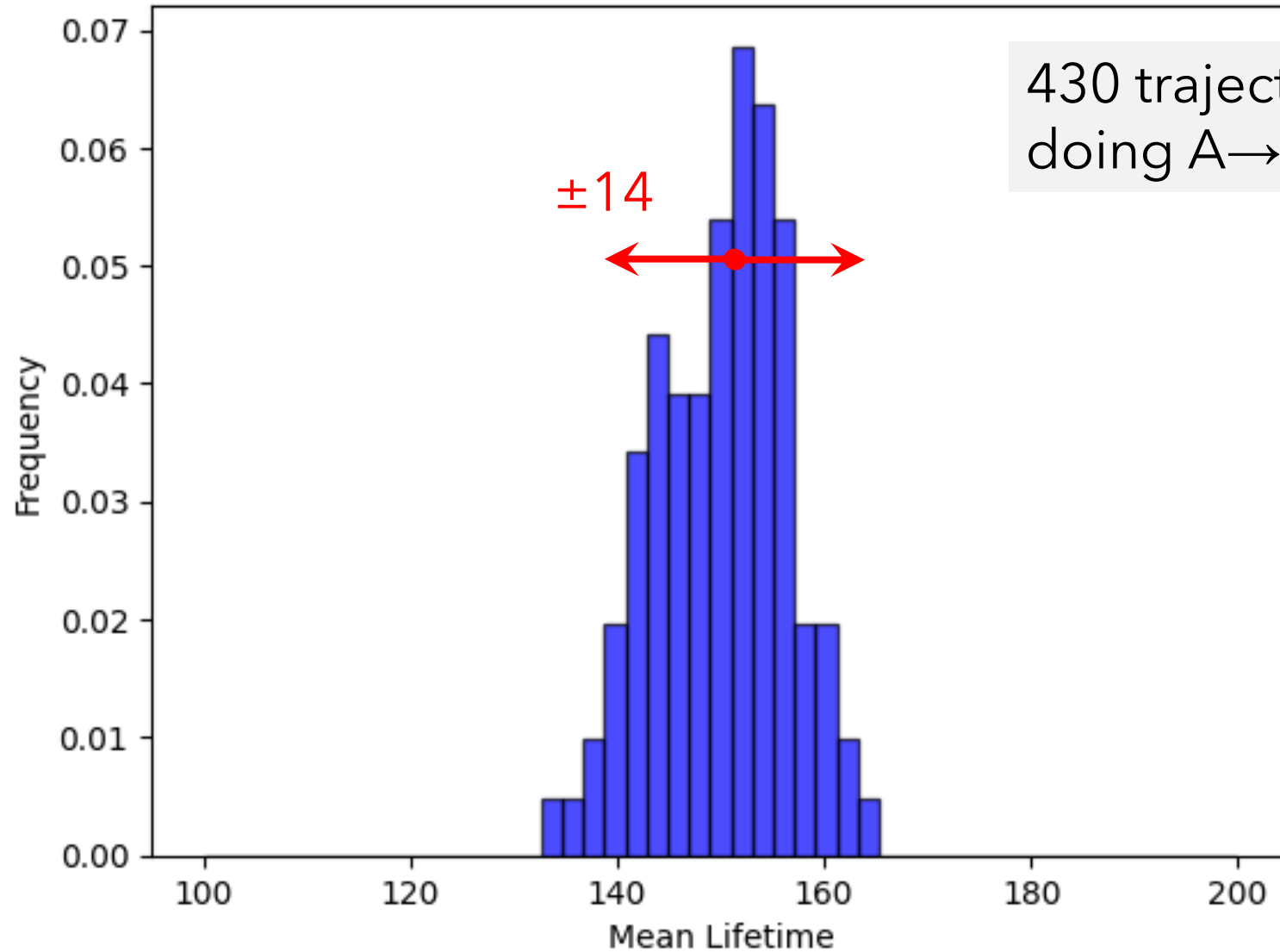


43 trajectories  
doing A→B isomerization

$\pm 45$

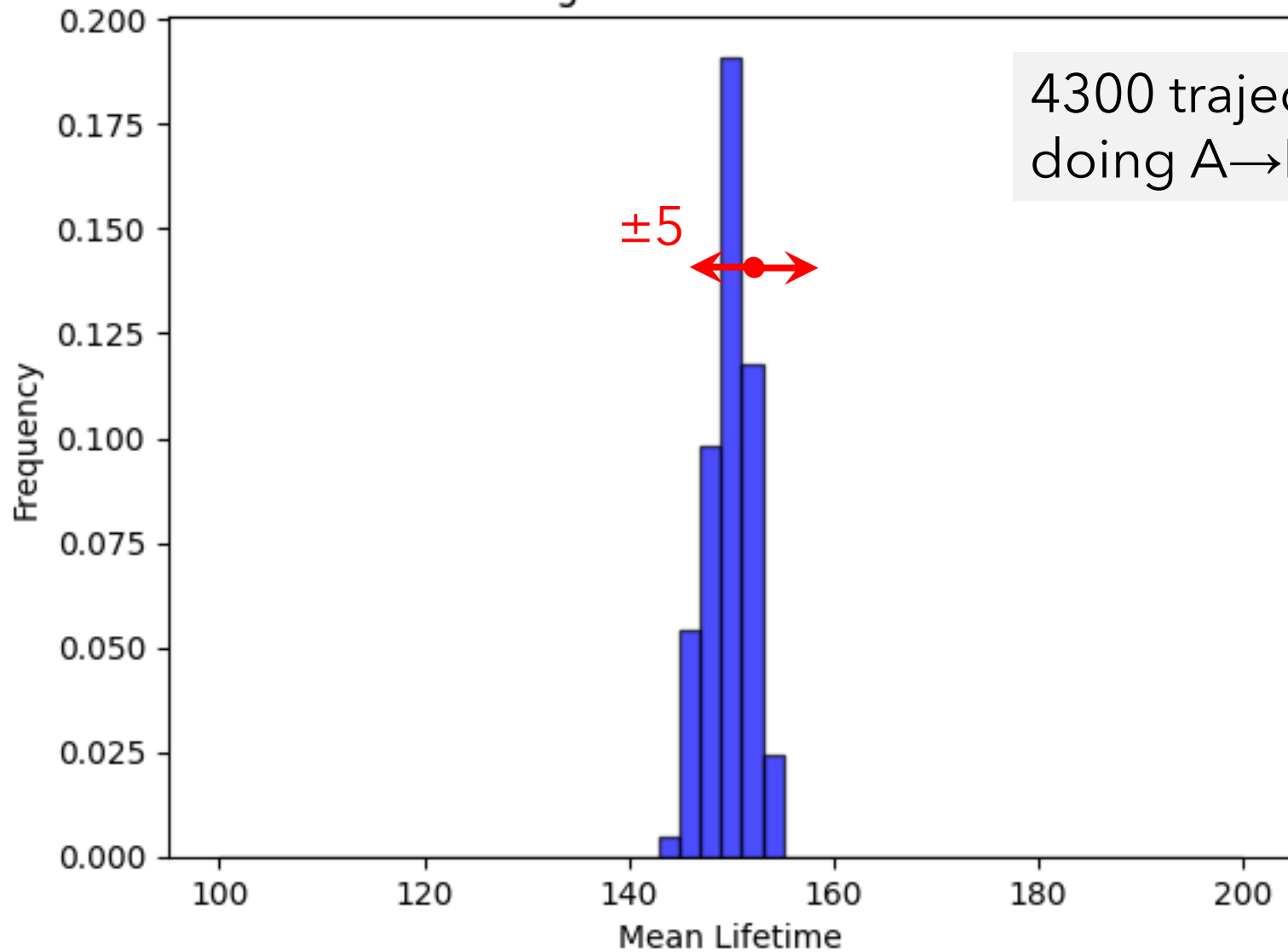


Histogram of Lifetime Means



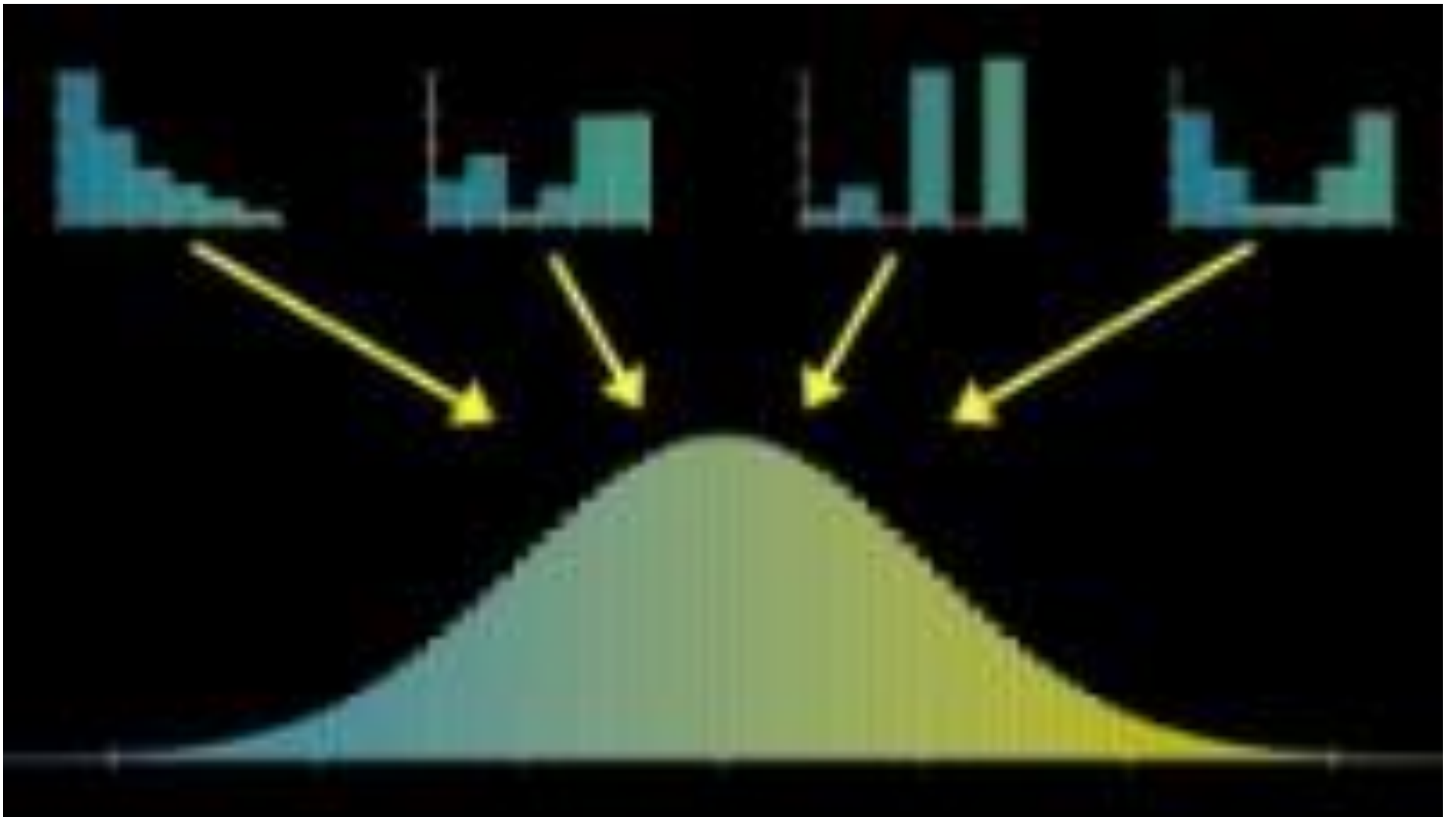
430 trajectories  
doing A→B isomerization

Histogram of Lifetime Means



4300 trajectories  
doing A→B isomerization

$\pm 5$



[www.youtube.com/watch?v=zeJD6dqJ5lo](http://www.youtube.com/watch?v=zeJD6dqJ5lo)

$N = 100$  trajectories

<b>Channel</b>	<b>Proportion of trajectories (%)</b>
No isomerization	5
Isomer A	52
Isomer B	43

# Margin of error for a population proportion

$$\varepsilon_p = z(x\%) \sqrt{\frac{p(1-p)}{N}}$$

- $p$  is the population proportion
- $N$  is the population size
- $z(95\%) = 1.96$  (confidence interval)

$N = 100$  trajectories

<b>Channel</b>	<b>Proportion of trajectories (%)</b>
No isomerization	5
Isomer A	52
Isomer B	43

For a confidence of 95% ( $Z = 1.96$ ) the margin of error for "no isomerization" proportion is

$$\varepsilon_p = 1.96 \sqrt{\frac{p(1-p)}{N}} = 1.96 \sqrt{\frac{0.05(1-0.05)}{100}} = 4\%$$

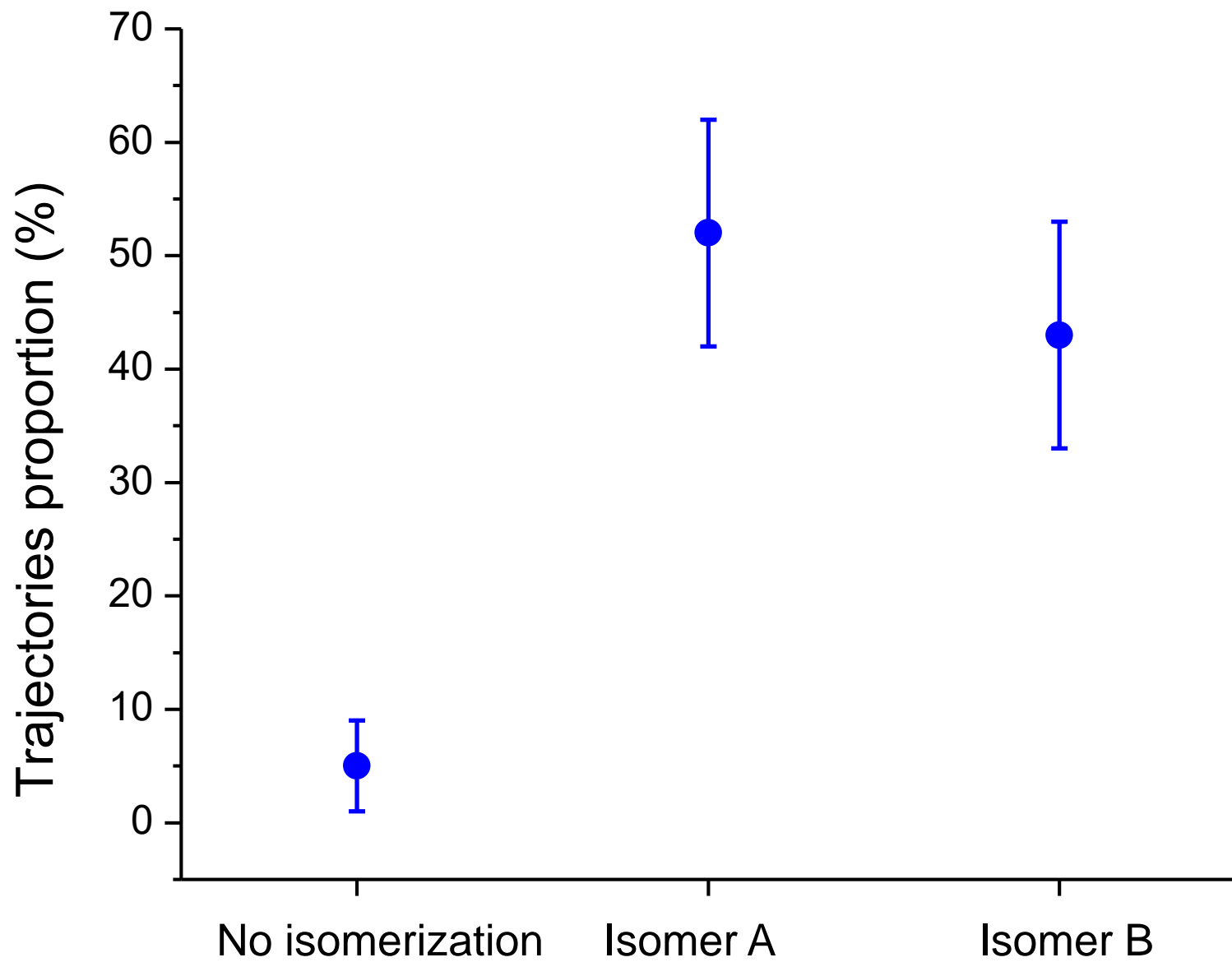
No isomerization is  $5 \pm 4$  % for 95% confidence.

This means if we repeat the simulations 100 times, we are confident that the number of "no isomerization" should be between 1% and 9% in 95 of the simulations.

<b>Channel</b>	<b>Proportion of trajectories (%)</b>	<b>Margin of error (%)</b>
No isomerization	5	4
Isomer A	52	10
Isomer B	43	10

**Which of the three channels is the most important?**





We cannot tell that isomer A is more abundant than B.

# Statistical errors in molecular dynamics: Bootstrapping

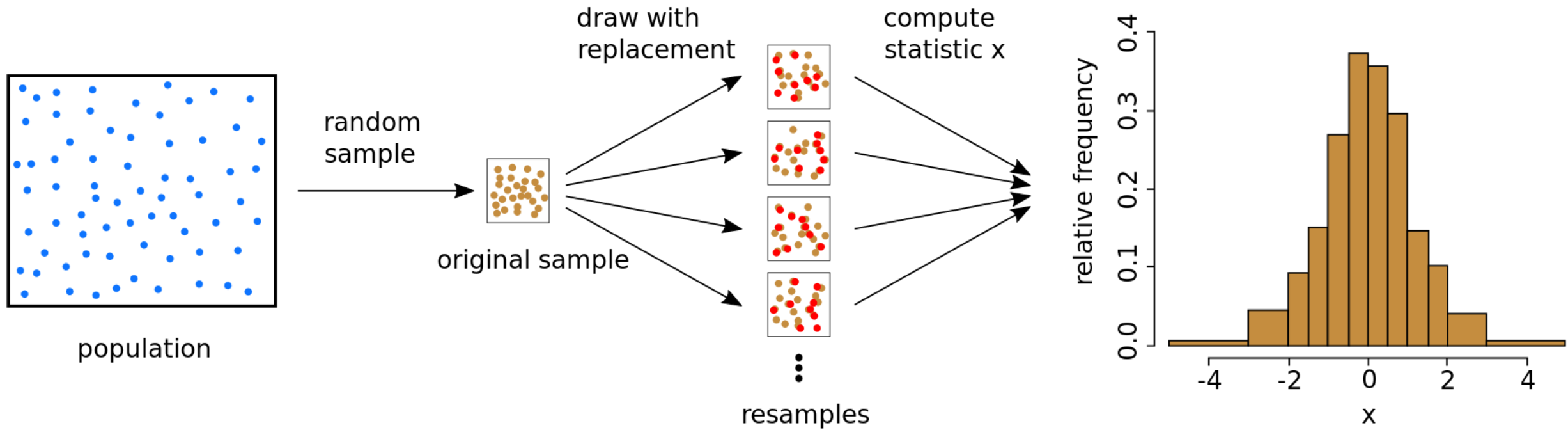
The previous discussion assumed a normal distribution of populations.

**Bootstrapping** is an alternative to estimating statistical uncertainty if you don't want to make such an assumption

## Bootstrapping:

Suppose you have  $N$  results and want to determine the margin of error.

1. Compute the mean of the  $N$  results.
2. From these  $N$  results, random pick  $N$  values. You can pick the same value more than once.
3. Compute the mean of this new set of this resampling of  $N$  values.
4. Repeat the resampling many times, computing the mean for each one.
5. Compute the standard deviation and confidence interval for the mean values.



## Simulation

Traj 1	92				
Traj 2	101				
Traj 3	98				
Traj 4	88				
Traj 5	94				
Traj 6	102				
Traj 7	109				
Traj 8	120				
Traj 9	89				
Traj 10	61				
Traj 11	114				
Traj 12	93				
Traj 13	138				
Traj 14	89				
Traj 15	108				
Traj 16	113				
Traj 17	115				
Traj 18	98				
Traj 19	105				
Traj 20	90				
<b>Mean</b>	<b>101</b>				
<b>Std dev</b>	16				
<b>95%SE</b>	<b>7</b>				

Result:

- Gaussian  
101 ± 7

	<b>Simulation</b>	<b>Resamp 1</b>				
<b>Traj 1</b>	92	89				
<b>Traj 2</b>	101	108				
<b>Traj 3</b>	98	138				
<b>Traj 4</b>	88	114				
<b>Traj 5</b>	94	98				
<b>Traj 6</b>	102	113				
<b>Traj 7</b>	109	108				
<b>Traj 8</b>	120	94				
<b>Traj 9</b>	89	61				
<b>Traj 10</b>	61	138				
<b>Traj 11</b>	114	108				
<b>Traj 12</b>	93	90				
<b>Traj 13</b>	138	109				
<b>Traj 14</b>	89	114				
<b>Traj 15</b>	108	113				
<b>Traj 16</b>	113	93				
<b>Traj 17</b>	115	93				
<b>Traj 18</b>	98	89				
<b>Traj 19</b>	105	120				
<b>Traj 20</b>	90	138				
<b>Mean</b>	<b>101</b>	106				
<b>Std dev</b>	16					
<b>95%SE</b>	<b>7</b>					

Result:  
• Gaussian  
101 ± 7

	Simulation	Resamp 1	Resamp 2			
Traj 1	92	89	109			
Traj 2	101	108	90			
Traj 3	98	138	109			
Traj 4	88	114	115			
Traj 5	94	98	98			
Traj 6	102	113	108			
Traj 7	109	108	108			
Traj 8	120	94	89			
Traj 9	89	61	98			
Traj 10	61	138	138			
Traj 11	114	108	92			
Traj 12	93	90	108			
Traj 13	138	109	92			
Traj 14	89	114	93			
Traj 15	108	113	89			
Traj 16	113	93	105			
Traj 17	115	93	113			
Traj 18	98	89	98			
Traj 19	105	120	88			
Traj 20	90	138	109			
<b>Mean</b>	<b>101</b>	106	102			
<b>Std dev</b>	16					
<b>95%SE</b>	<b>7</b>					

Result:  
• Gaussian  
101 ± 7



	Simulation	Resamp 1	Resamp 2	...	Resamp 9998	Resamp 9999	Resamp 10000
Traj 1	92	89	109		89	109	92
Traj 2	101	108	90		61	108	102
Traj 3	98	138	109		61	98	115
Traj 4	88	114	115		113	115	98
Traj 5	94	98	98		114	93	94
Traj 6	102	113	108		105	101	105
Traj 7	109	108	108		94	114	92
Traj 8	120	94	89		61	98	98
Traj 9	89	61	98		89	98	113
Traj 10	61	138	138		115	138	101
Traj 11	114	108	92		120	98	98
Traj 12	93	90	108		92	101	109
Traj 13	138	109	92		90	98	101
Traj 14	89	114	93		89	109	88
Traj 15	108	113	89		92	105	98
Traj 16	113	93	105		101	98	113
Traj 17	115	93	113		92	93	90
Traj 18	98	89	98		108	120	61
Traj 19	105	120	88		89	89	94
Traj 20	90	138	109		113	109	93
<b>Mean</b>	<b>101</b>	106	102	...	94	105	98
<b>Std dev</b>	16						
<b>95%SE</b>	<b>7</b>						

Result:

- Gaussian  
101 ± 7
- Bootstrapping  
101 ± 6

**95%SE**  
**6**

```
def bootstrap_stats(data, num_samples=1000, confidence_level=0.95):
    boot_samples = np.random.choice(data, size=(num_samples, len(data)), replace=True)

    # Calculate mean for each bootstrap sample
    means = np.mean(boot_samples, axis=1)

    # Calculate standard deviation of the mean
    std_dev_mean = np.std(means)

    # Calculate 95% confidence intervals for mean
    ci_mean = np.percentile(means, [(1-confidence_level)/2*100, (1+confidence_level)/2*100])

    # Calculate mean of the original data
    original_mean = np.mean(data)

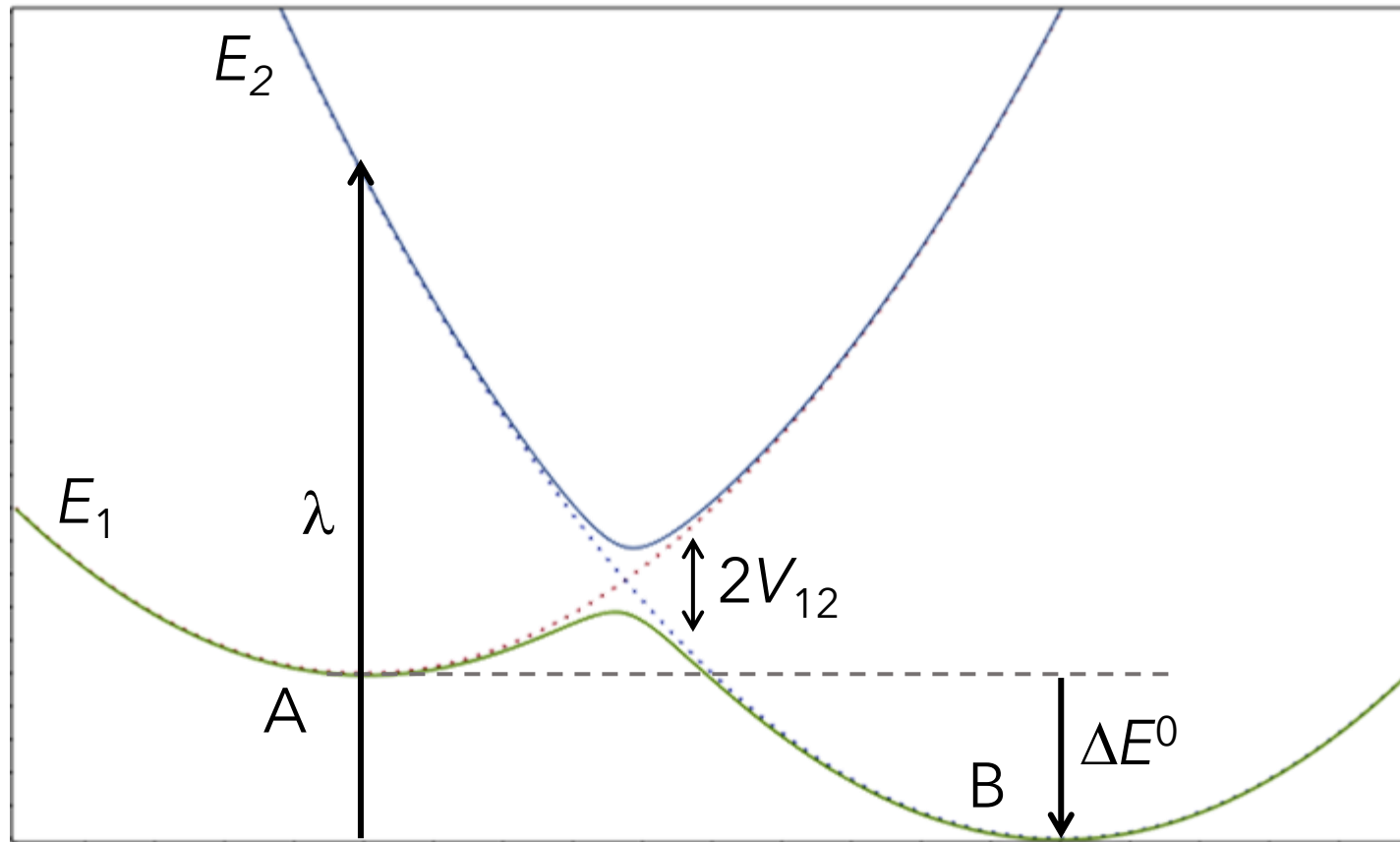
    # Print results
    print("Original Mean:", original_mean)
    print("Standard Deviation of Mean:", std_dev_mean)
    print(f"95% ME of Bootstrapped Mean: {(ci_mean[1]-ci_mean[0])/2}")

# Example usage:
times = [61, 88, 89, 89, 90, 92, 93, 94, 98, 98, 101, 102, 105, 108, 109, 113, 114, 115, 120, 138]
result = bootstrap_stats(times)
```

# Monte Carlo Sampling of Errors

Suppose you want to compute Marcus rate for the reaction  $A \rightarrow B$

$$W_{AB} = \frac{2\pi}{\hbar} |V_{12}|^2 \frac{1}{\sqrt{4\pi\lambda k_B T}} \exp\left(-\frac{(\lambda + \Delta E^0)^2}{4\lambda k_B T}\right)$$



- Adiabatic energy gap  
 $\Delta E^0 = -0.5 \pm 0.1$  eV
- Reorganization energy  
 $\lambda = 2.0 \pm 0.1$  eV
- Diabatic coupling  
 $V_{12} = 0.05 \pm 0.01$  eV
- Temperature  
 $T = 300 \pm 1$  K

The uncertainties are 95% CI

Suppose you want to compute Marcus rate for the reaction  $A \rightarrow B$

$$W_{AB} = \frac{2\pi}{\hbar} |V_{12}|^2 \frac{1}{\sqrt{4\pi\lambda k_B T}} \exp\left(-\frac{(\lambda + \Delta E^0)^2}{4\lambda k_B T}\right)$$

- Boltzmann constant  
 $k_B = 8.617333262 \times 10^{-5} \text{ eV/K}$
- Reduced Planck constant  
 $\hbar = 6.582119569 \times 10^{-16} \text{ eV.s}$

*Naked* results:

Marcus rate =  $223 \text{ ns}^{-1}$

Marcus lifetime =  $4.5 \text{ ps}$

- Adiabatic energy gap  
 $\Delta E^0 = -0.5 \pm 0.1 \text{ eV}$
- Reorganization energy  
 $\lambda = 2.0 \pm 0.1 \text{ eV}$
- Diabatic coupling  
 $V_{12} = 0.05 \pm 0.01 \text{ eV}$
- Temperature  
 $T = 300 \pm 1 \text{ K}$

The uncertainties are 95% CI

# What about the uncertainties?

```
def marcus_rate(DE, RE, V, TEMP): # Marcus rate
    W = 2*math.pi/hbar*
        1/math.sqrt(4*math.pi*RE*kB*TEMP)*
        math.exp(-(RE+DE)**2/(4*RE*kB*TEMP))

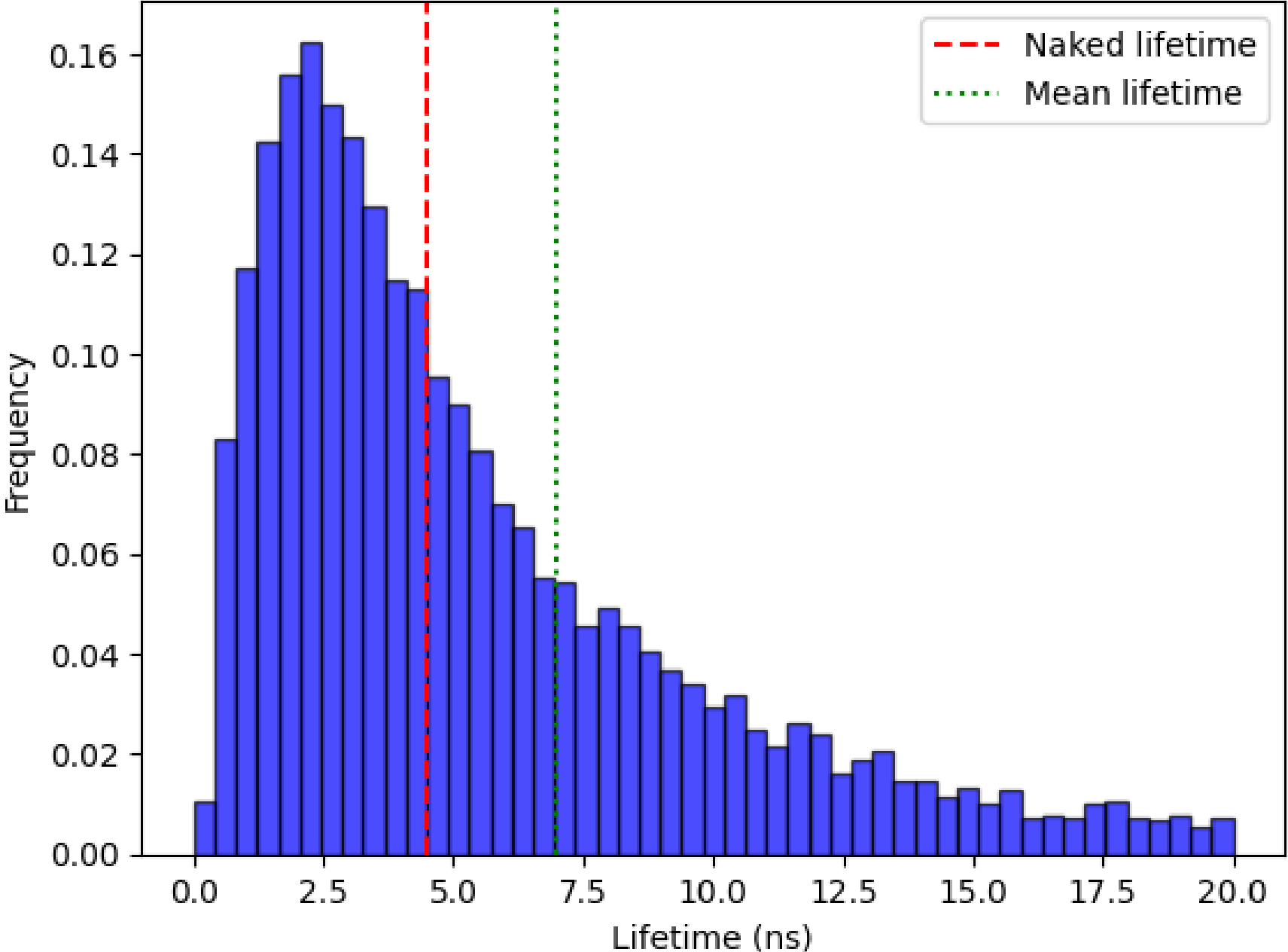
    return W

Ntot = 100000 # Number of points
Z = 1.96 # 95% CI
all_values = []
for i in range(1, Ntot+1): # Loop over random values
    DE = np.random.normal(loc=DE0, scale=DDE0/Z)
    RE = np.random.normal(loc=lamb, scale=Dlamb/Z)
    V = np.random.normal(loc=V12, scale=DV12/Z)
    TEMP = np.random.normal(loc=T, scale=DT/Z)

    time = 1/marcus_rate(DE, RE, V, TEMP) # Lifetime

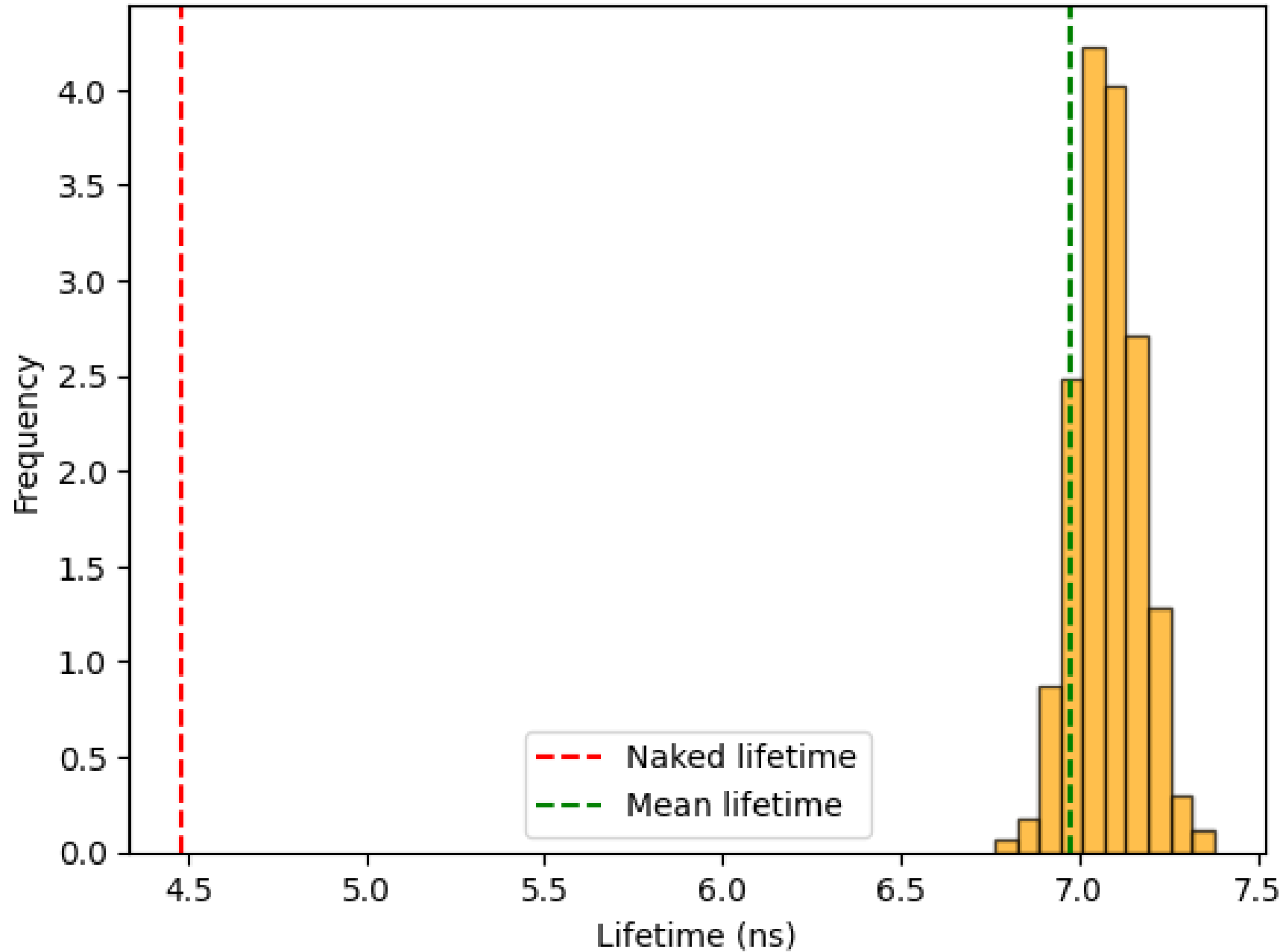
    # store
    all_values.append(time)
```

Histogram of Lifetimes



Distribution of  
10,000 random  
samples

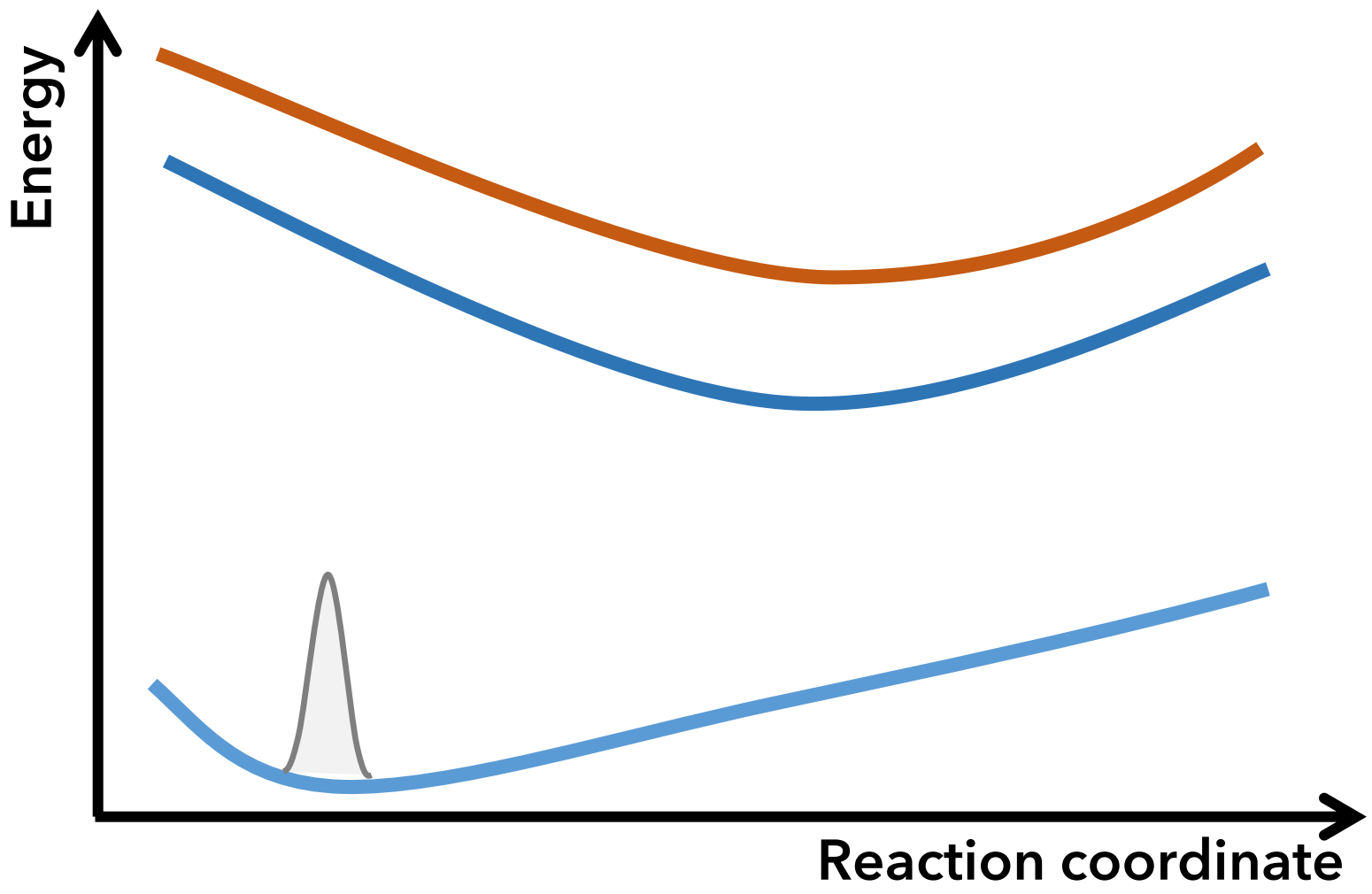
# Histogram of Mean Lifetimes

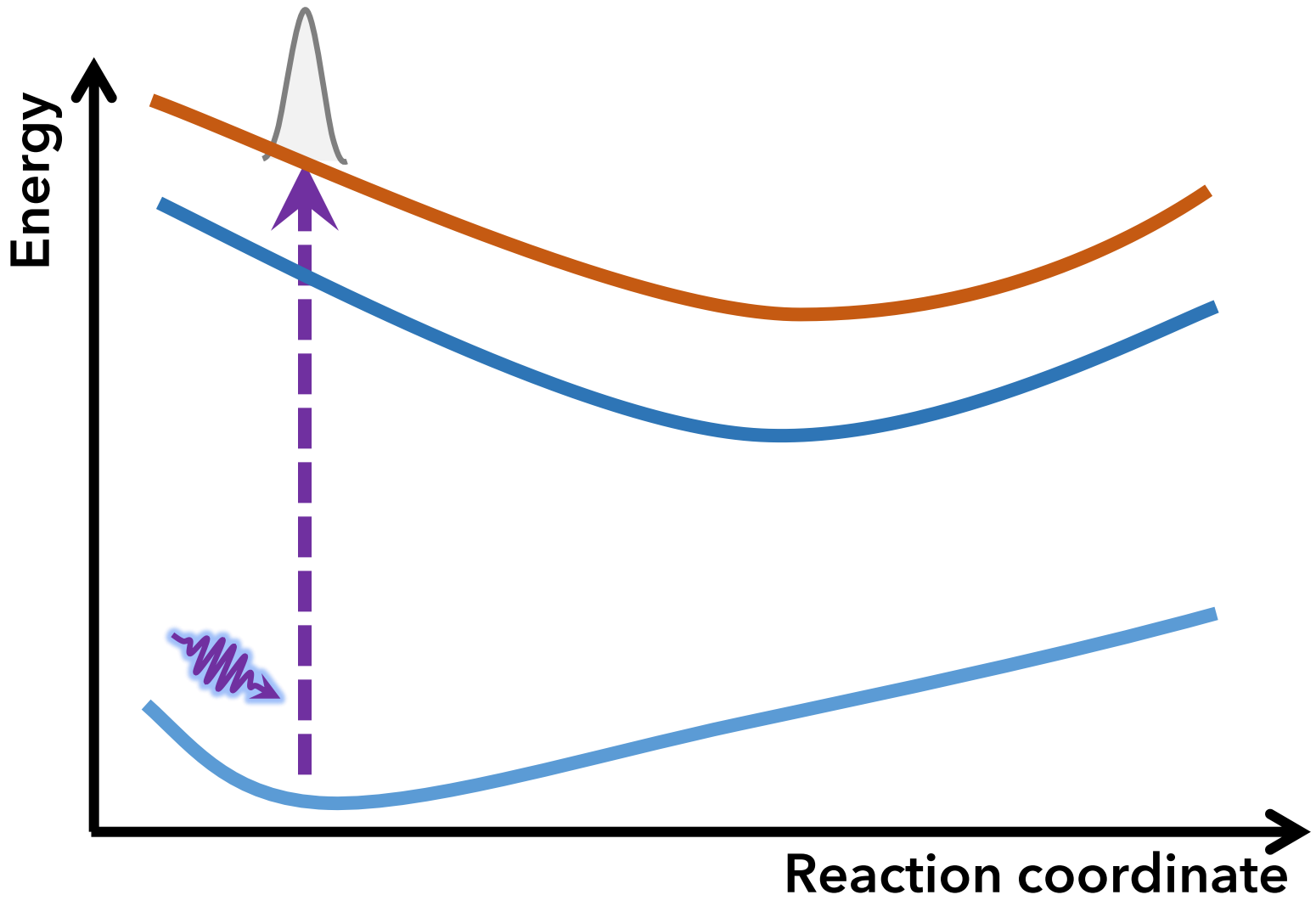


Distribution of  
1000 mean values.



**Practical use of Monte  
Carlo integration:  
Spectrum simulation**





A simple expression for the photoabsorption cross-section  $\sigma$  from the ground state (0) to state  $n$  is (in atomic units)

$$\sigma(E) = \frac{2\pi^2}{c} \frac{1}{E} \int |\chi_0(\mathbf{R})|^2 \Delta E_{0n}(\mathbf{R}) f_{0n}(\mathbf{R}) g(E - \Delta E_{0n}(\mathbf{R}), \sigma) d\mathbf{R}$$

$E$ : photon energy

$\mathbf{R}$ : nuclear coordinates

$\chi_0(\mathbf{R})$ : nuclear wave function of the ground state

$\Delta E_{0n}(\mathbf{R})$ : potential energy gap  $0 \rightarrow n$

$f_{0n}(\mathbf{R})$ : oscillator strength  $0 \rightarrow n$

$g(E - \Delta E, \sigma)$ : Normalized Gaussian in  $E$ ,

centered in  $\Delta E$  and standard deviation  $\sigma$

A simple expression for the photoabsorption cross-section  $\sigma$  from the ground state (0) to state  $n$  is (in atomic units)

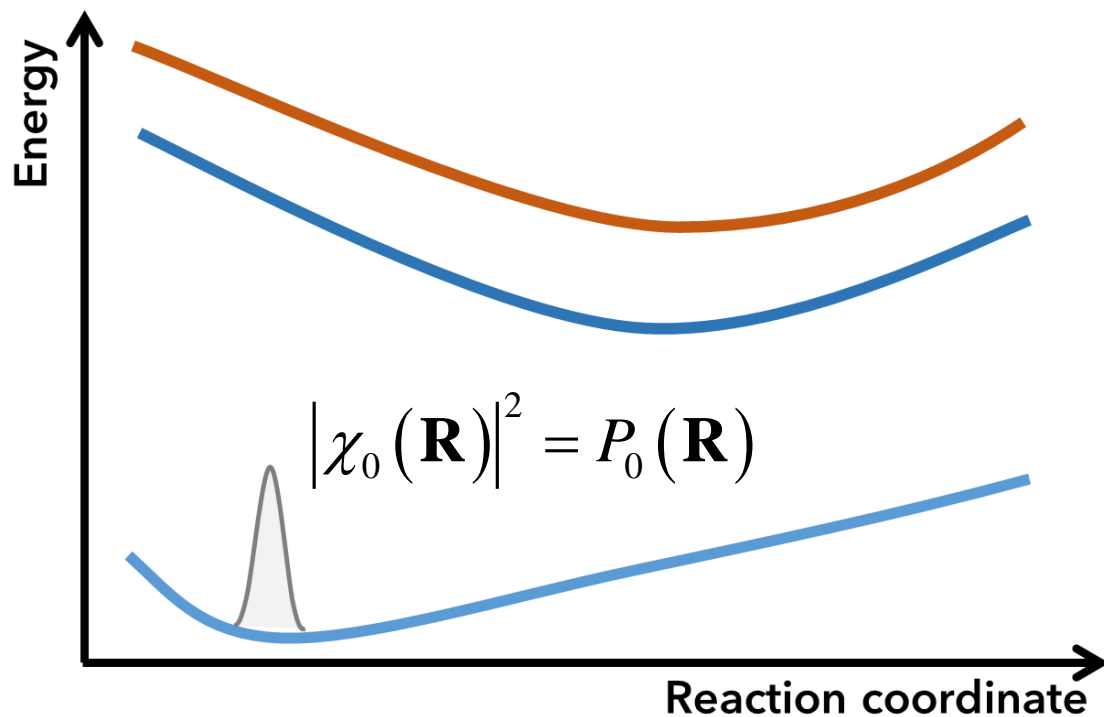
$$\sigma(E) = \frac{2\pi^2}{c} \frac{1}{E} \int |\chi_0(\mathbf{R})|^2 \Delta E_{0n}(\mathbf{R}) f_{0n}(\mathbf{R}) g(E - \Delta E_{0n}(\mathbf{R}), \sigma) d\mathbf{R}$$

$\sigma$  has unity of area and it is given in *bohr*<sup>2</sup>

We can convert it to extinction coefficient  $\varepsilon$  (in L.mol<sup>-1</sup>.cm<sup>-1</sup>) via

$$\varepsilon(E) = 7321.2134 \sigma(E)$$

$$\sigma(E) = \frac{2\pi^2}{c} \frac{1}{E} \int |\chi_0(\mathbf{R})|^2 \Delta E_{0n}(\mathbf{R}) f_{0n}(\mathbf{R}) g(E - \Delta E_{0n}(\mathbf{R}), \sigma) d\mathbf{R}$$



$P_0(\mathbf{R})$  is the **probability density** of finding the molecule in the ground state with geometry  $\mathbf{R}$ .

$P_0(\mathbf{R}).d\mathbf{R}$  is the **probability** of finding the molecule in the ground state with geometry  $\mathbf{R}$  within the volume  $d\mathbf{R}$ .

$$\sigma(E) = \int P_0(\mathbf{R}) \underbrace{\left[ \frac{2\pi^2}{c} \frac{1}{E} \Delta E_{0n}(\mathbf{R}) f_{0n}(\mathbf{R}) g(E - \Delta E_{0n}(\mathbf{R}), \sigma) \right]}_{h(E, \mathbf{R})} d\mathbf{R}$$

$$\sigma(E) = \int P_0(\mathbf{R}) h(E, \mathbf{R}) d\mathbf{R}$$

We use Monte Carlo to integrate this expression in  $\mathbf{R}$

$$\sigma(E) = \int P_0(\mathbf{R}) h(E, \mathbf{R}) d\mathbf{R}$$

1. Sample  $N_p$  random geometries  $\mathbf{R}_i$  following the distribution  $P_0(\mathbf{R})$
2. Compute  $h(\mathbf{R}_i)$  for each geometry
3. Estimate the integral as

$$\sigma(E) \approx \frac{1}{N_p} \sum_{i=1}^{N_p} h(E, \mathbf{R}_i)$$

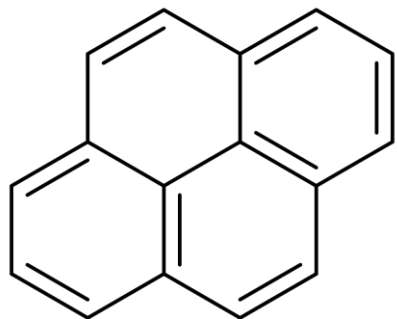


The photoabsorption cross section is

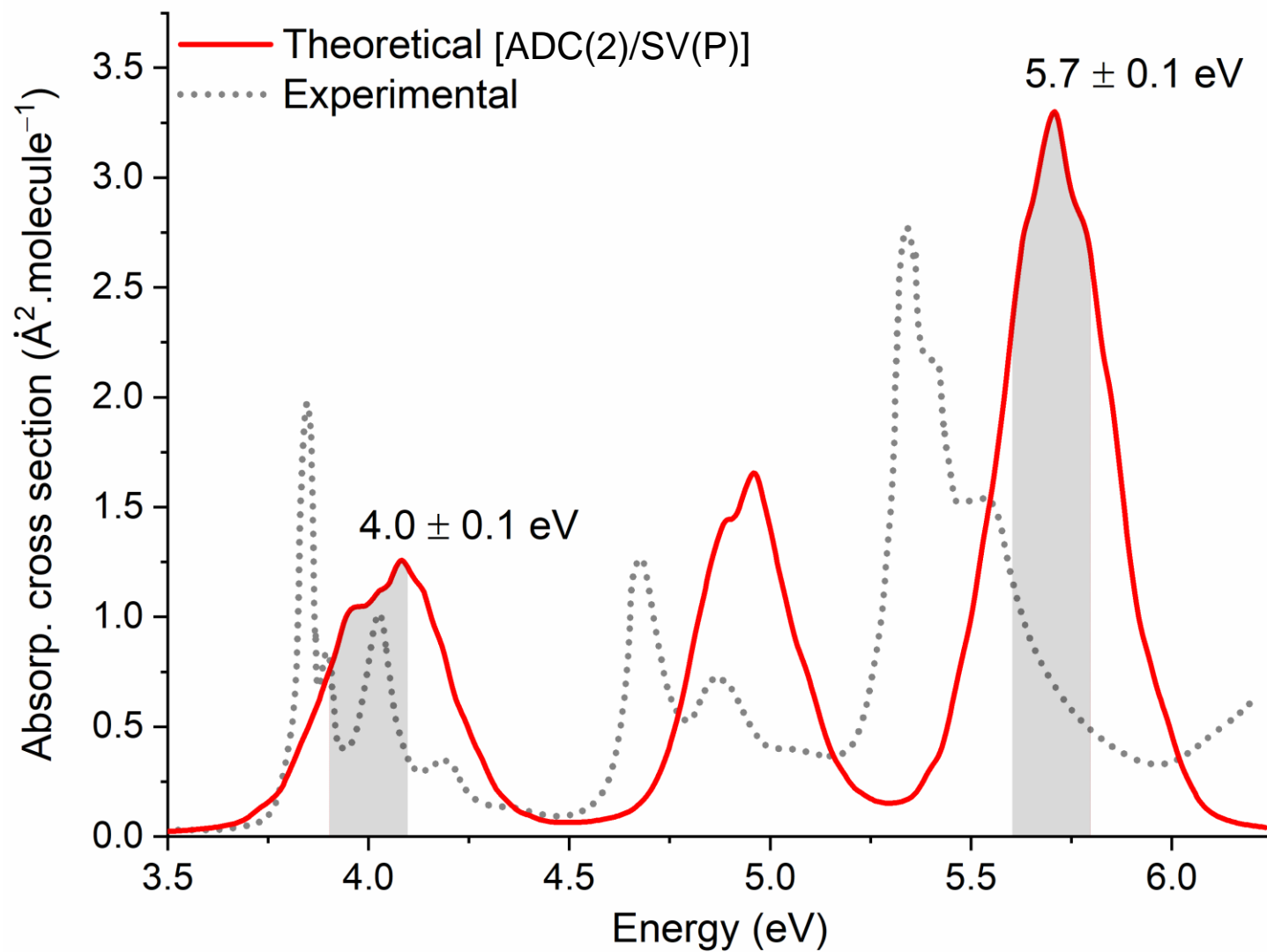
$$\sigma(E) = \frac{2\pi^2}{c} \frac{1}{N_p E} \sum_{i=1}^{N_p} \Delta E_{0n}(\mathbf{R}_i) f_{0n}(\mathbf{R}_i) g(E - \Delta E_{0n}(\mathbf{R}_i), \sigma)$$

where  $\mathbf{R}_i$  are nuclear geometries sampled from the probability density function  $|\chi_0(\mathbf{R})|^2$ .

- $\Delta E_{0n}(\mathbf{R}_i)$  and  $f_{0n}(\mathbf{R}_i)$  are computed with quantum chemistry (TDDFT, for example)
- The distribution  $|\chi_0(\mathbf{R})|^2$  can be taken from
  - ✓ a trajectory in the ground state
  - ✓ or from a Wigner sampling of the harmonic oscillator

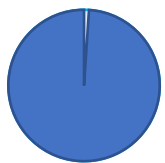
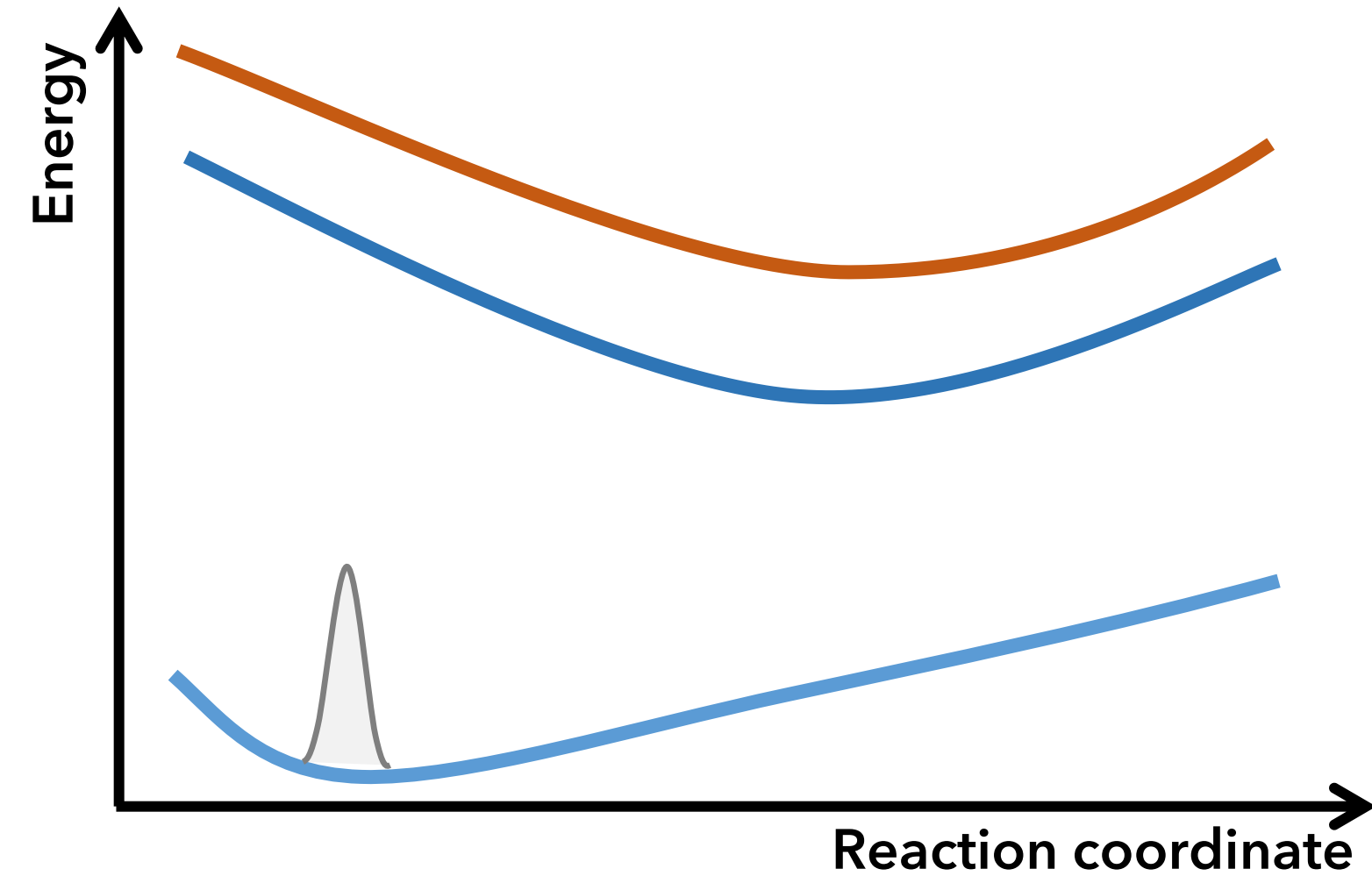


pyrene

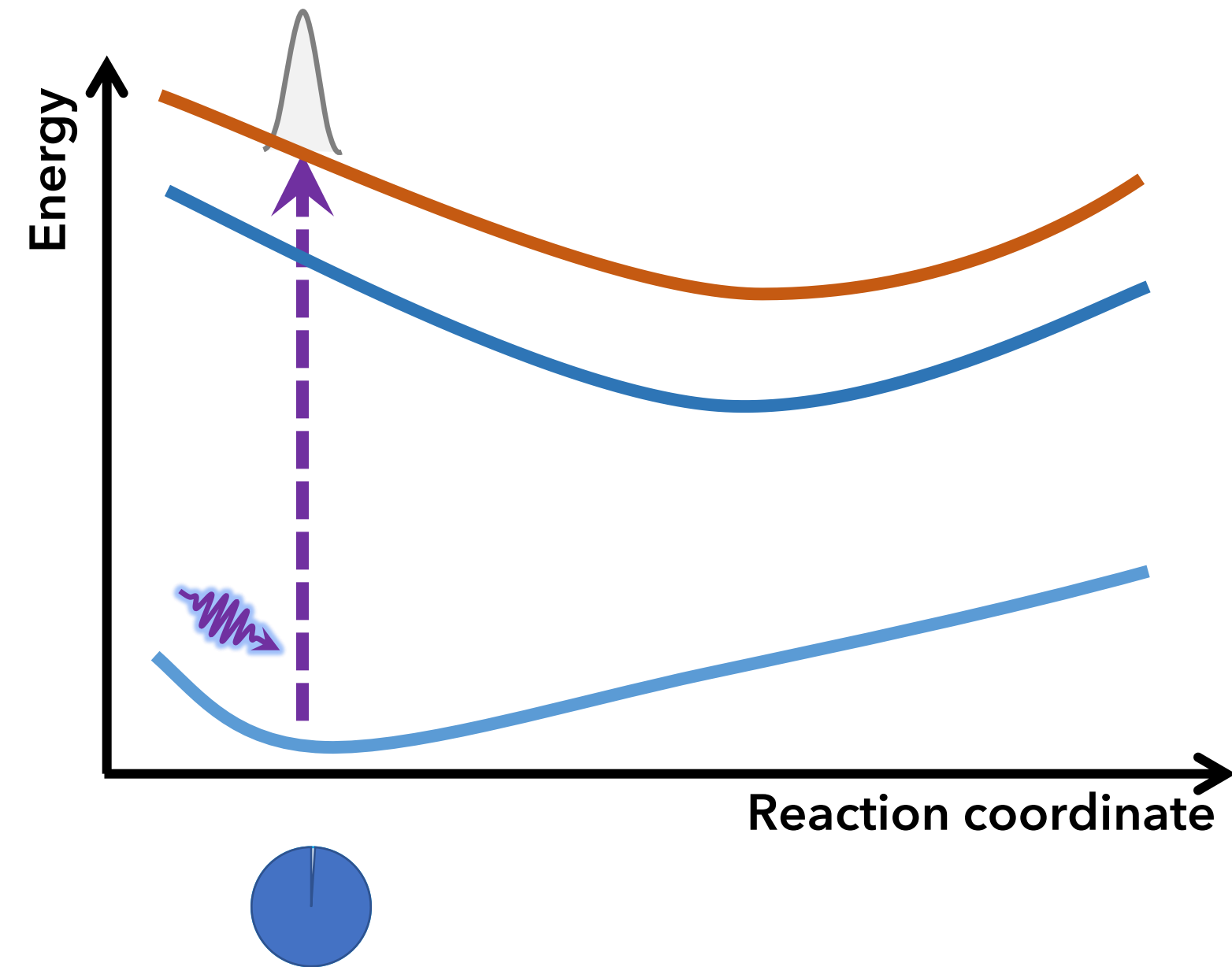


**Case study:  
Non-Kasha fluorescence  
of pyrene**

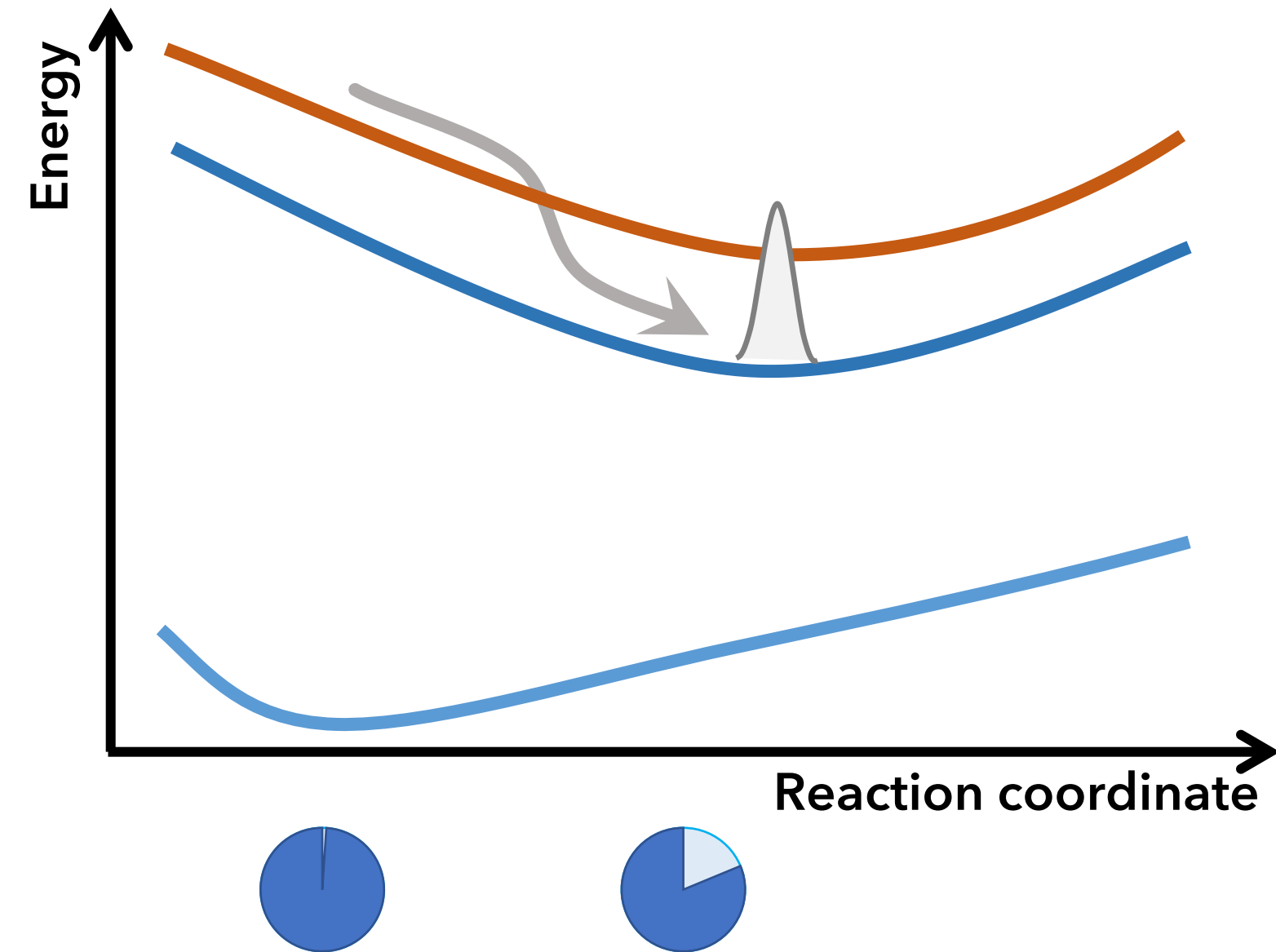
# Nonadiabatic dynamics



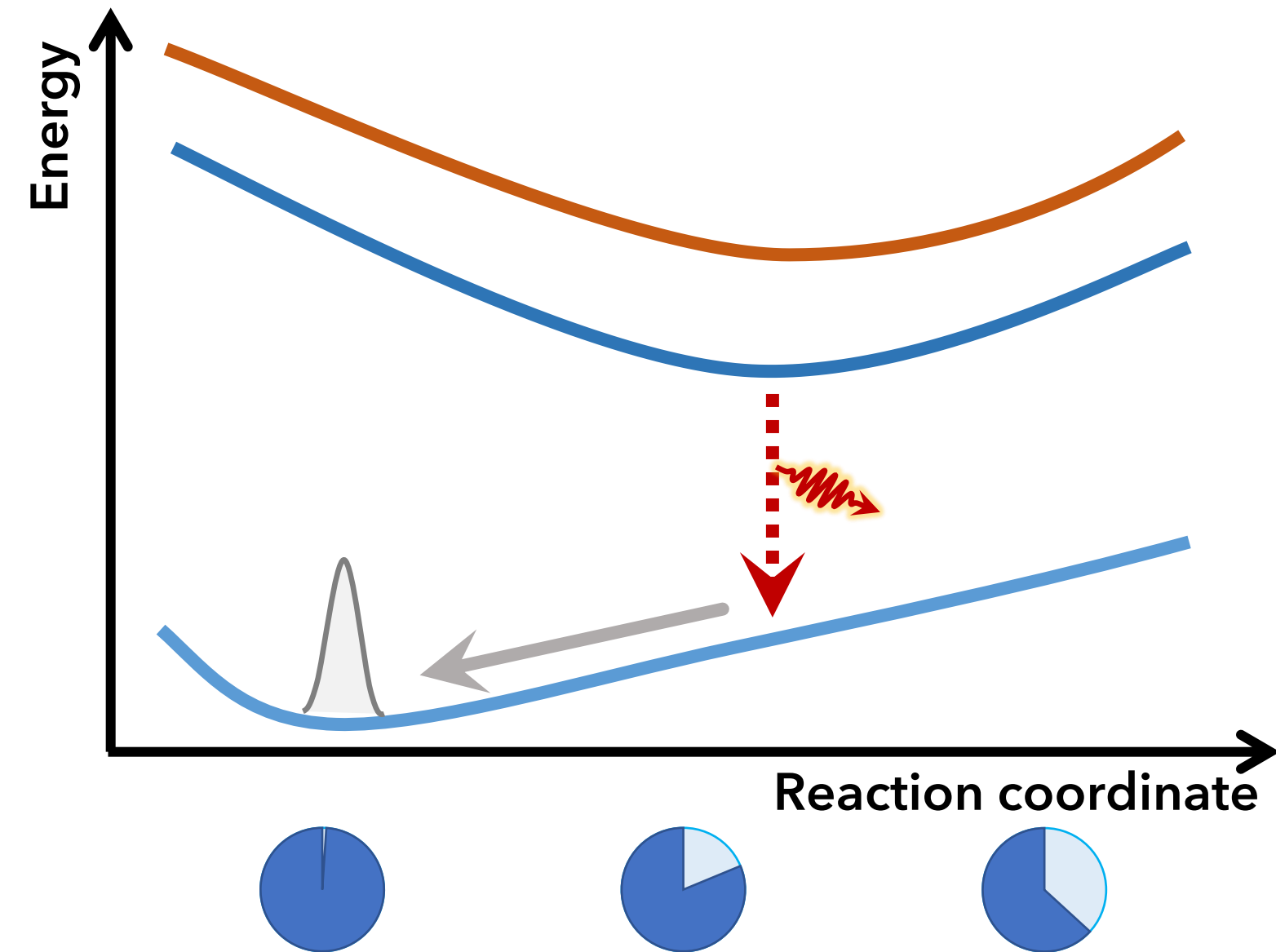
# Photoabsorption



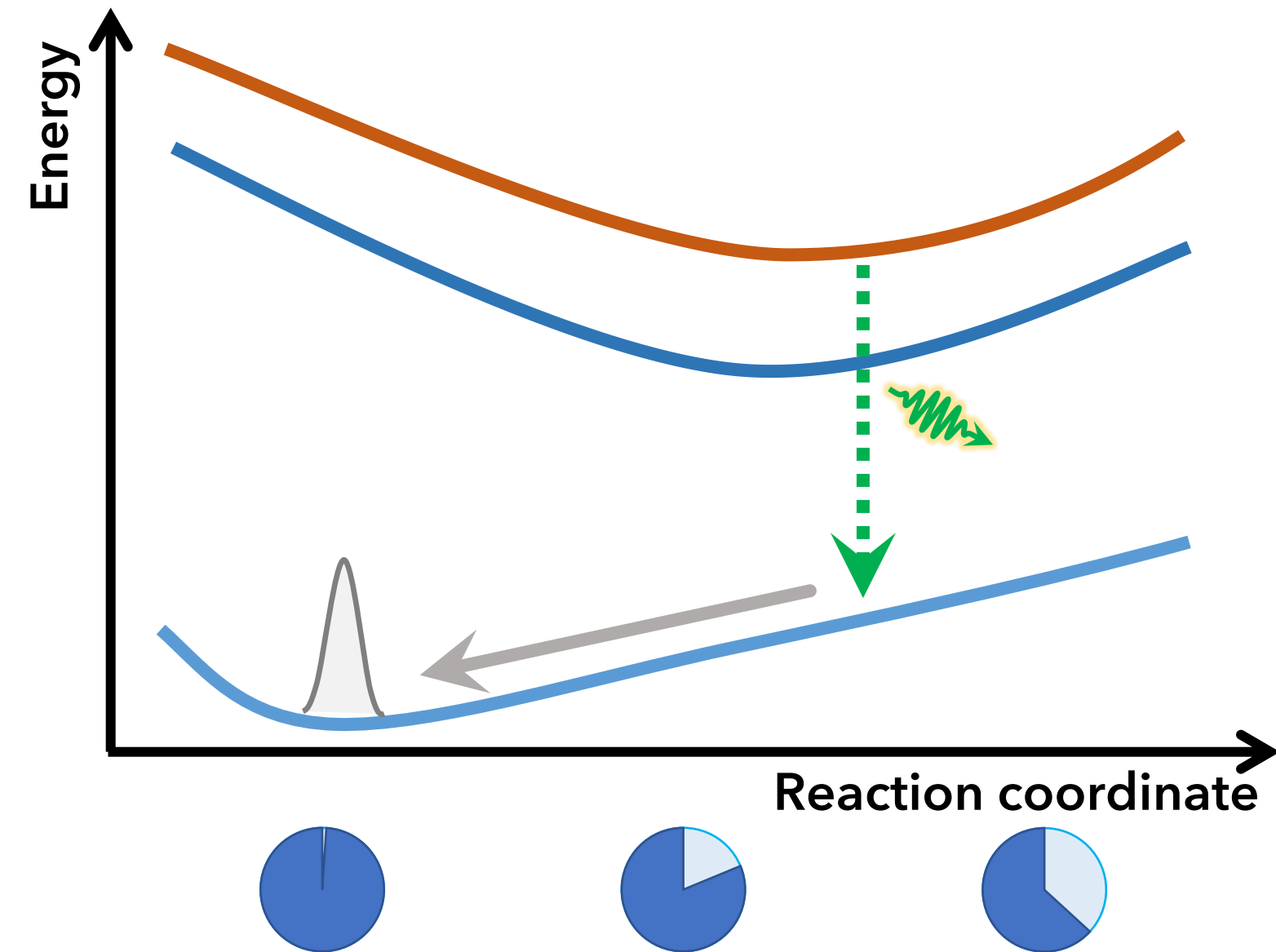
# Kasha rule



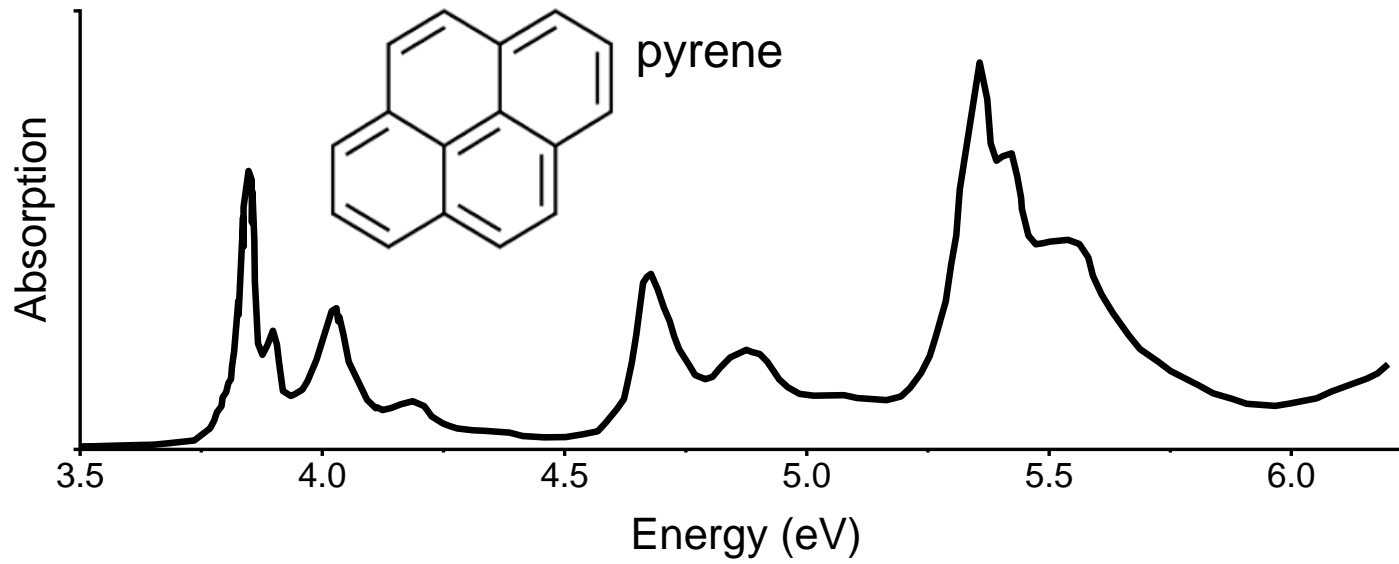
# Fluorescence

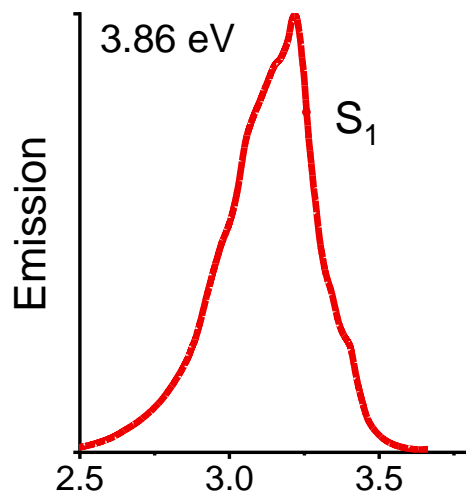
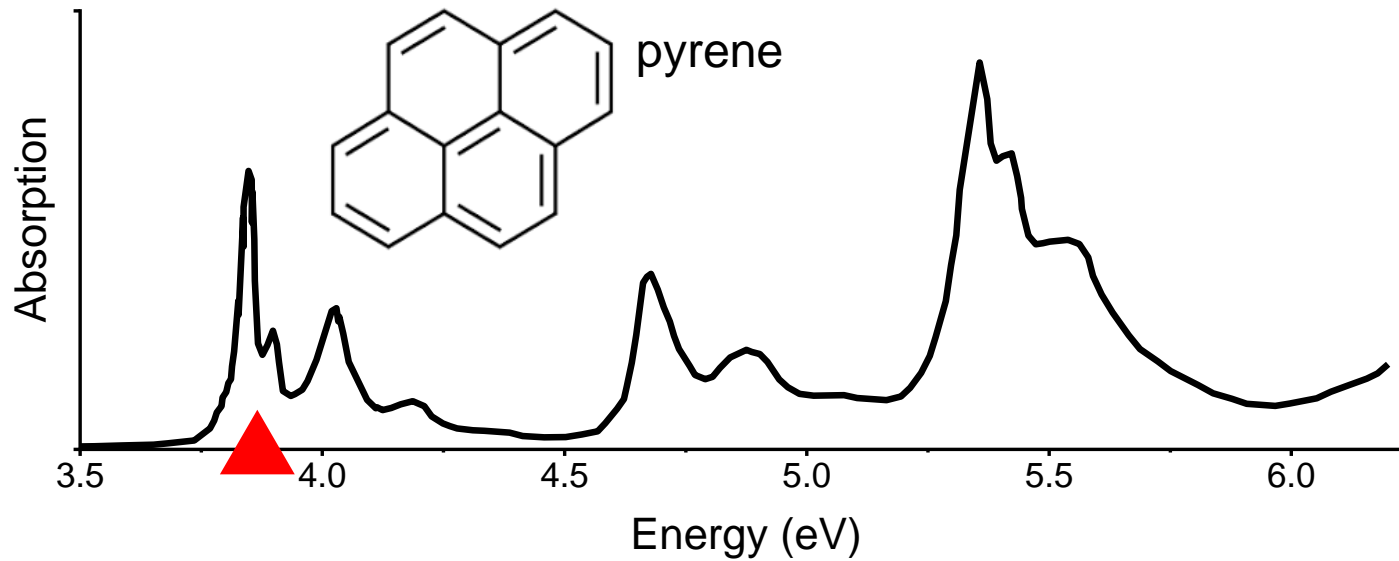


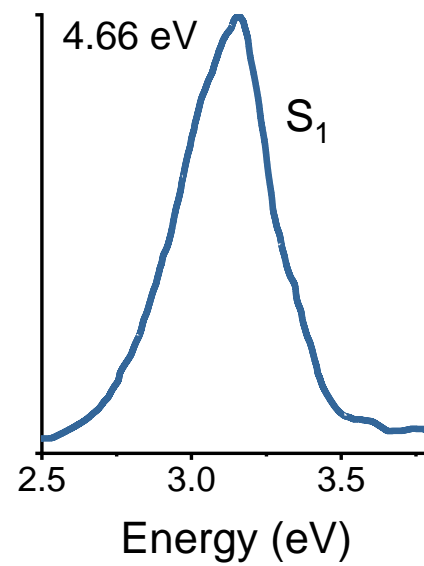
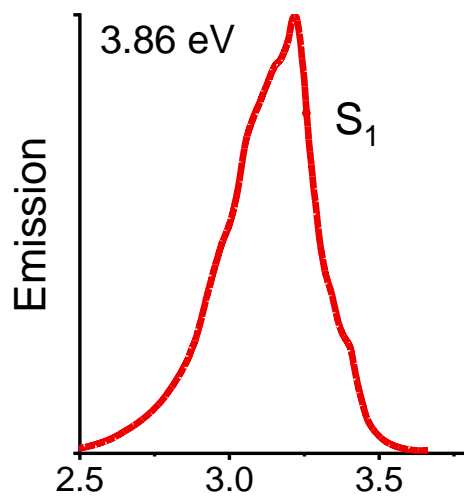
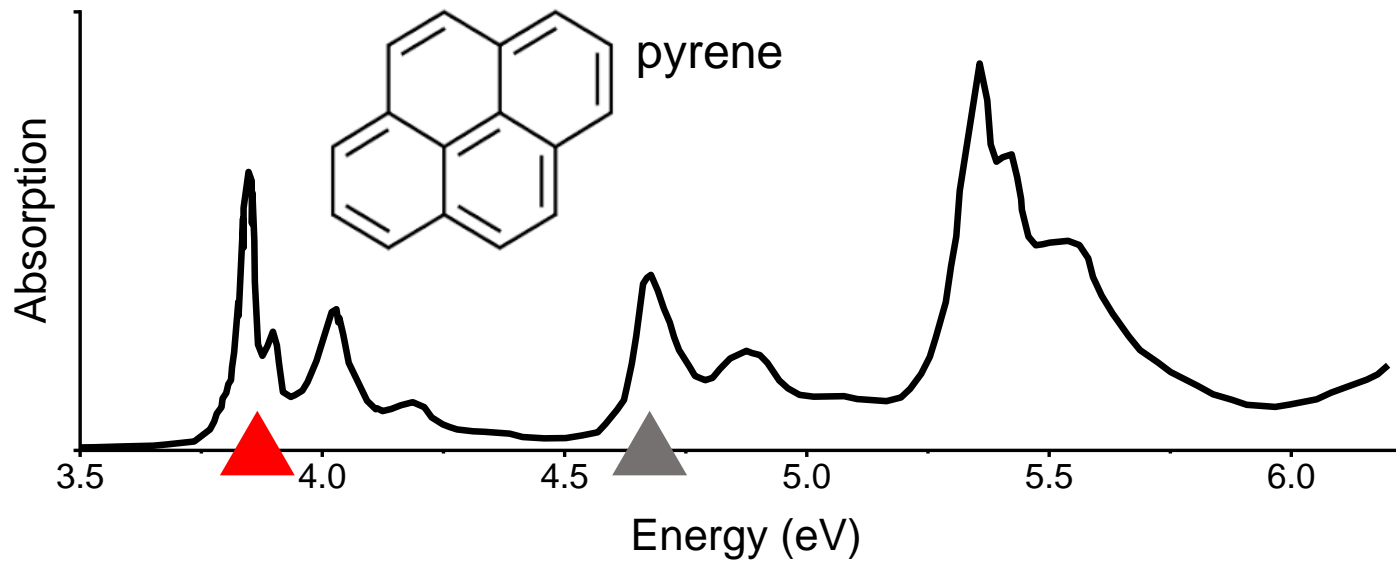
# Non-Kasha fluorescence

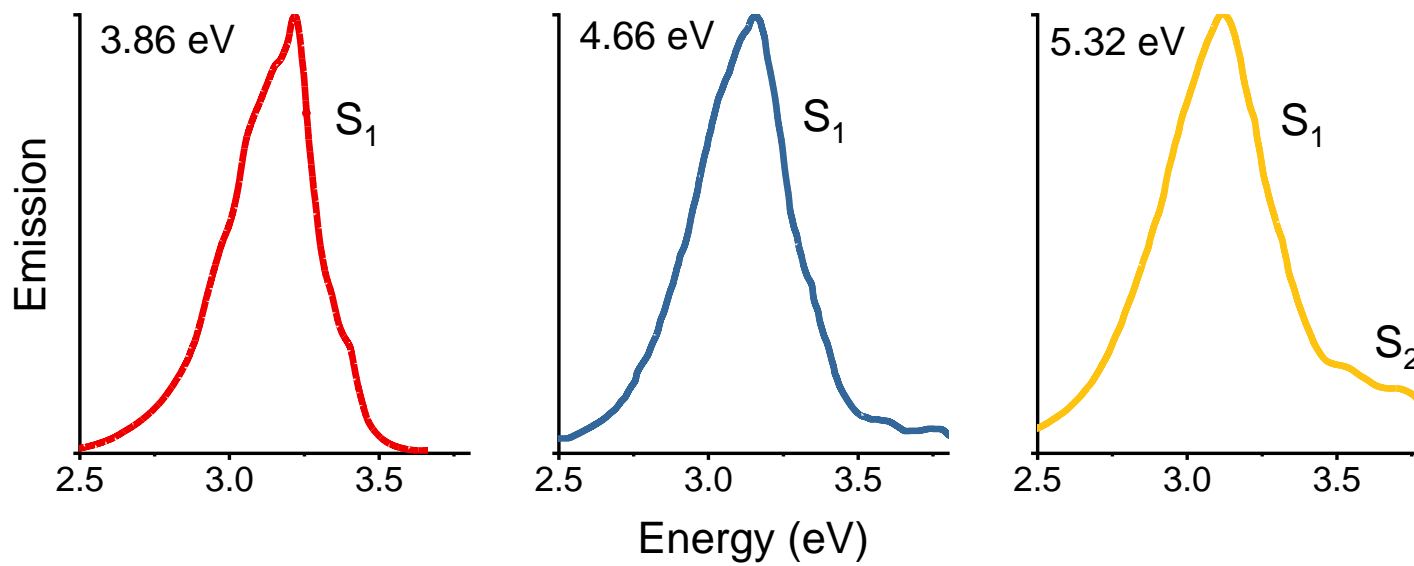
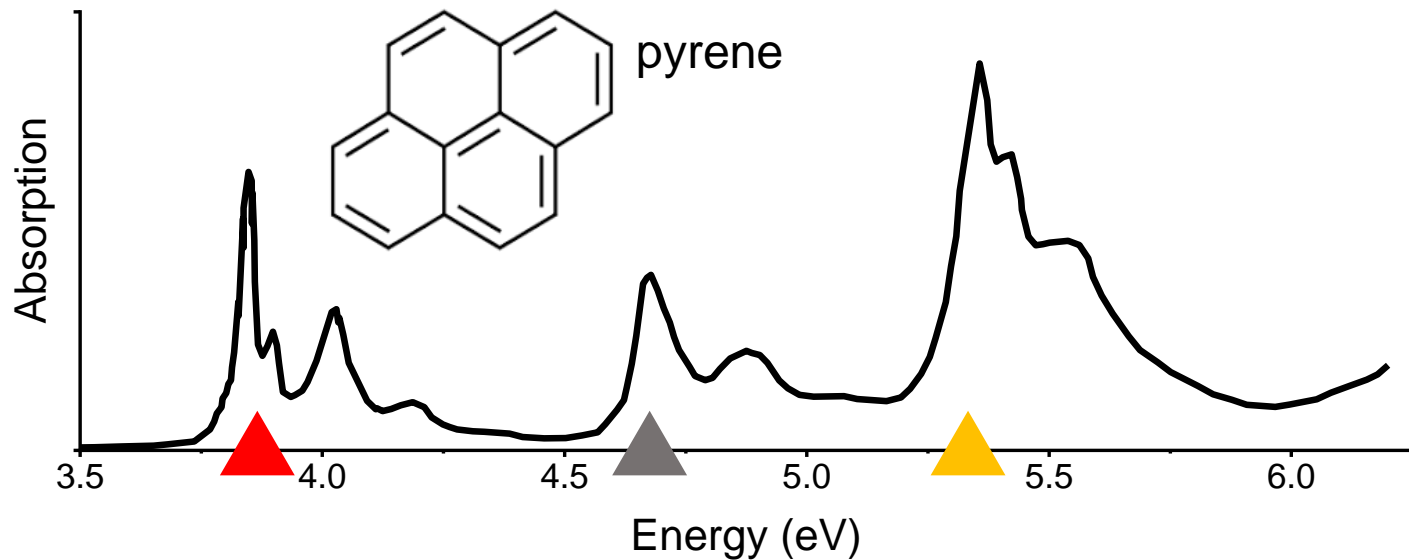


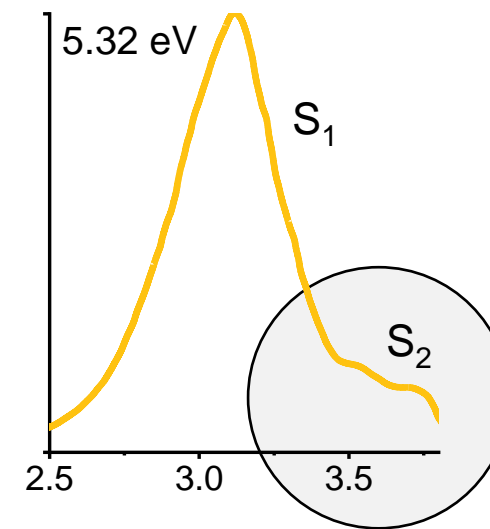
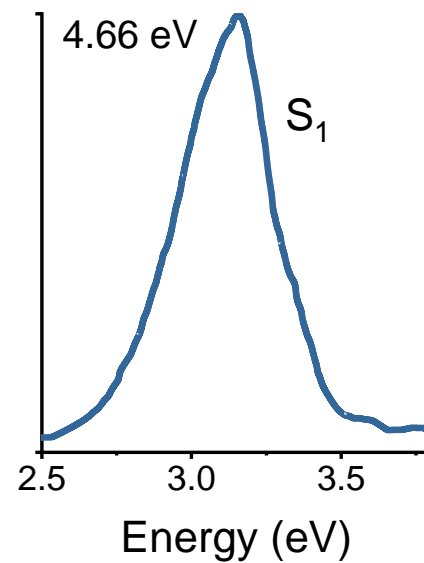
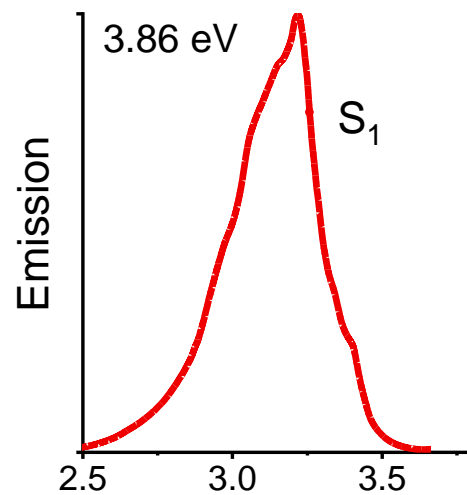
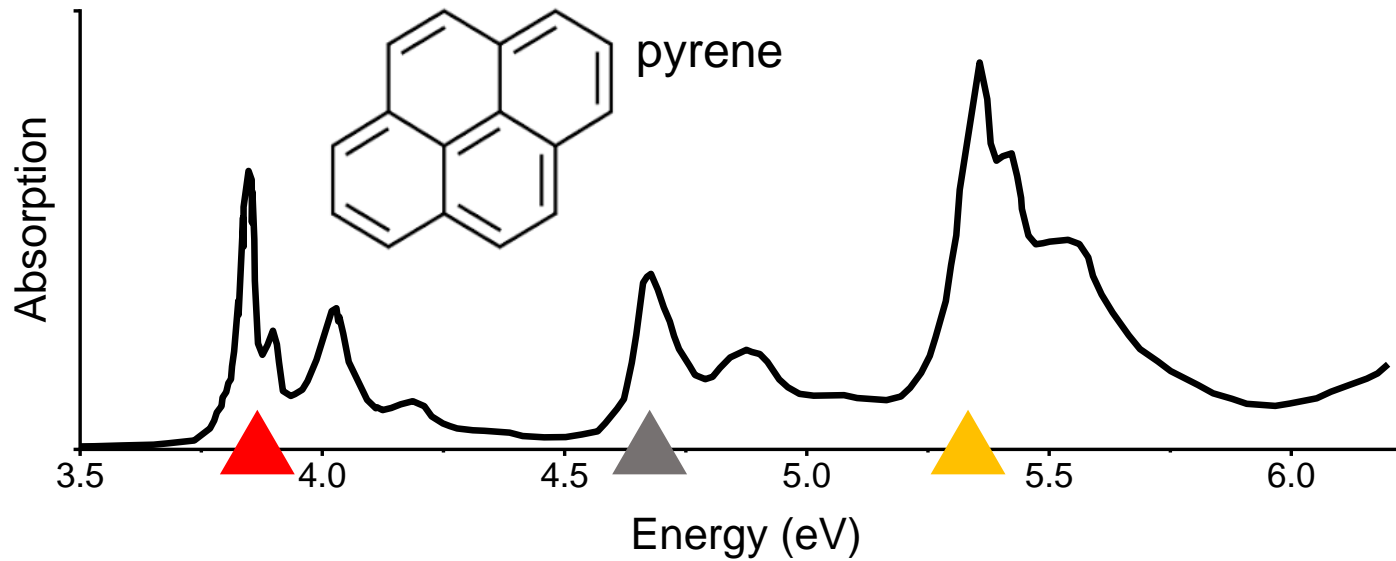








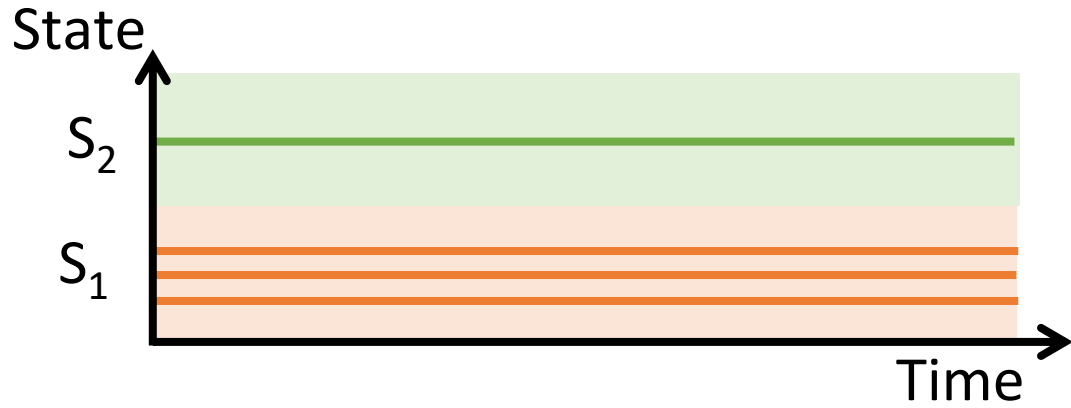




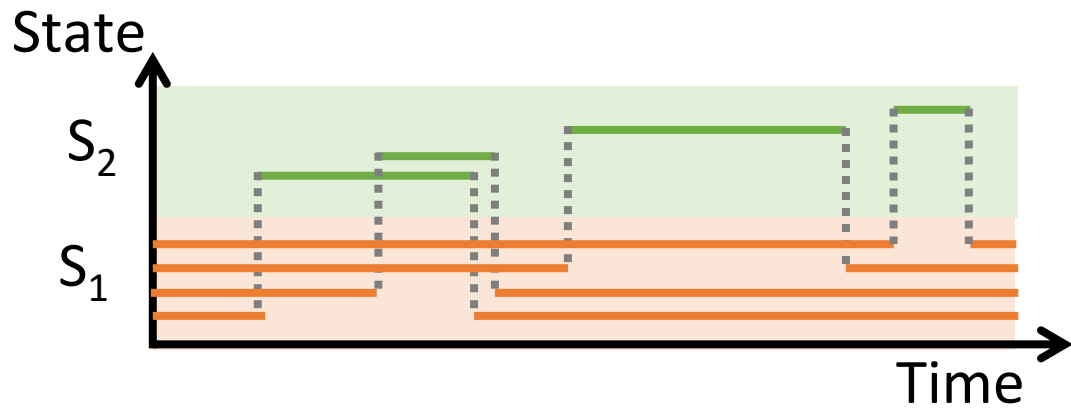
Non-Kasha emission?

Numata et al. *J Photochem Photobiol A* **2012**, 237, 49: YES  
 del Valle; Catalán. *PCCP* **2019**, 21, 10061: NO

## Static equilibrium



## Dynamic equilibrium



If the peak is non-Kasha,  
which kind of non-Kasha?

# Non-Kasha fluorescence of pyrene emerges from a dynamic equilibrium between excited states

Cite as: *J. Chem. Phys.* **157**, 154305 (2022); doi: [10.1063/5.0113908](https://doi.org/10.1063/5.0113908)

Submitted: 25 July 2022 • Accepted: 15 September 2022 •

Published Online: 19 October 2022



View Online



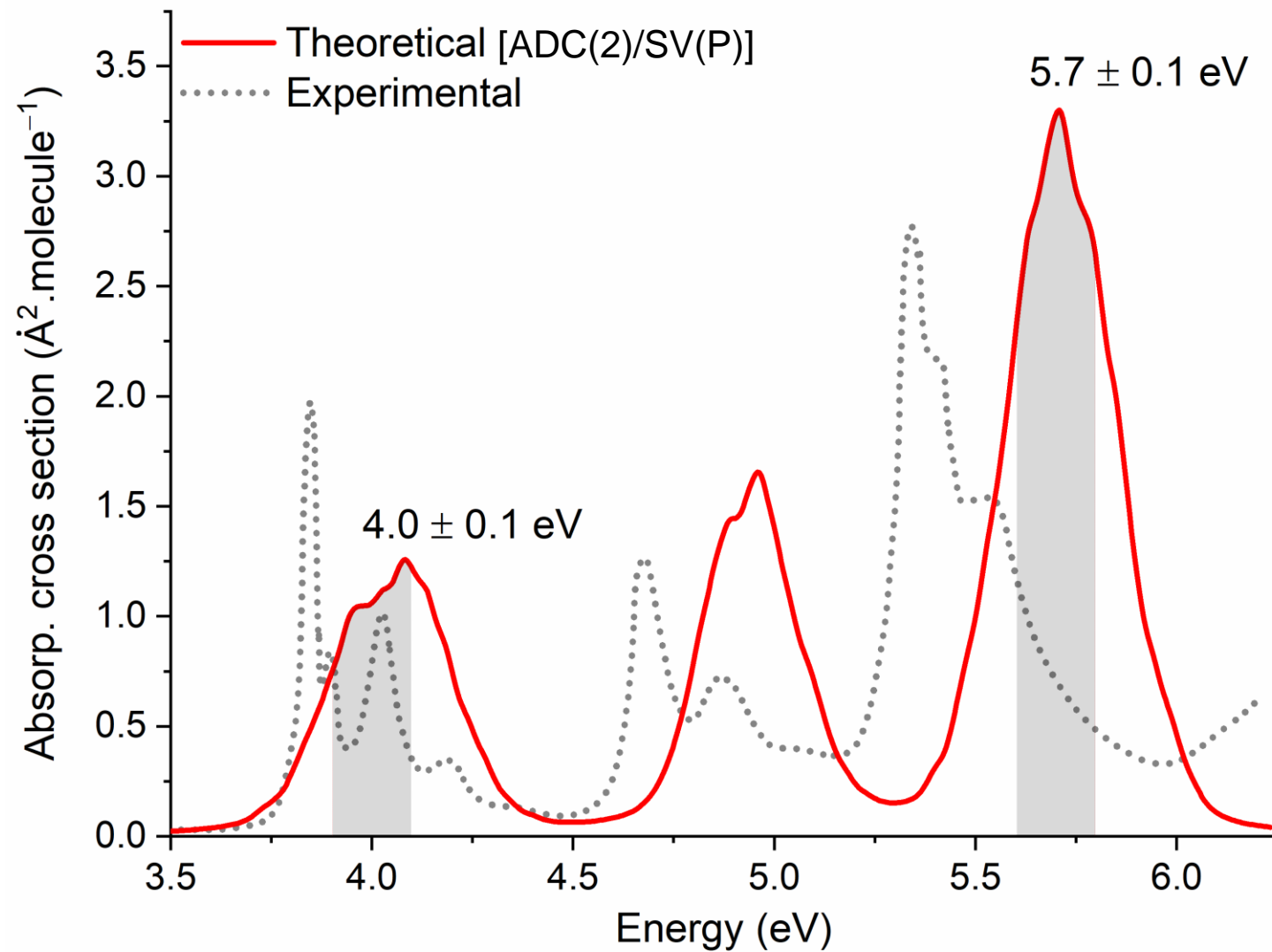
Export Citation



CrossMark

Gabriel Braun,<sup>1</sup> Itamar Borges, Jr.,<sup>1,a)</sup> Adélia J. A. Aquino,<sup>2</sup> Hans Lischka,<sup>3,a)</sup> Felix Plasser,<sup>4</sup>   
Silmar A. do Monte,<sup>5</sup> Elizete Ventura,<sup>5,a)</sup> Saikat Mukherjee,<sup>6</sup> and Mario Barbatti<sup>6,7,a)</sup>

# Dynamics from two excitation windows





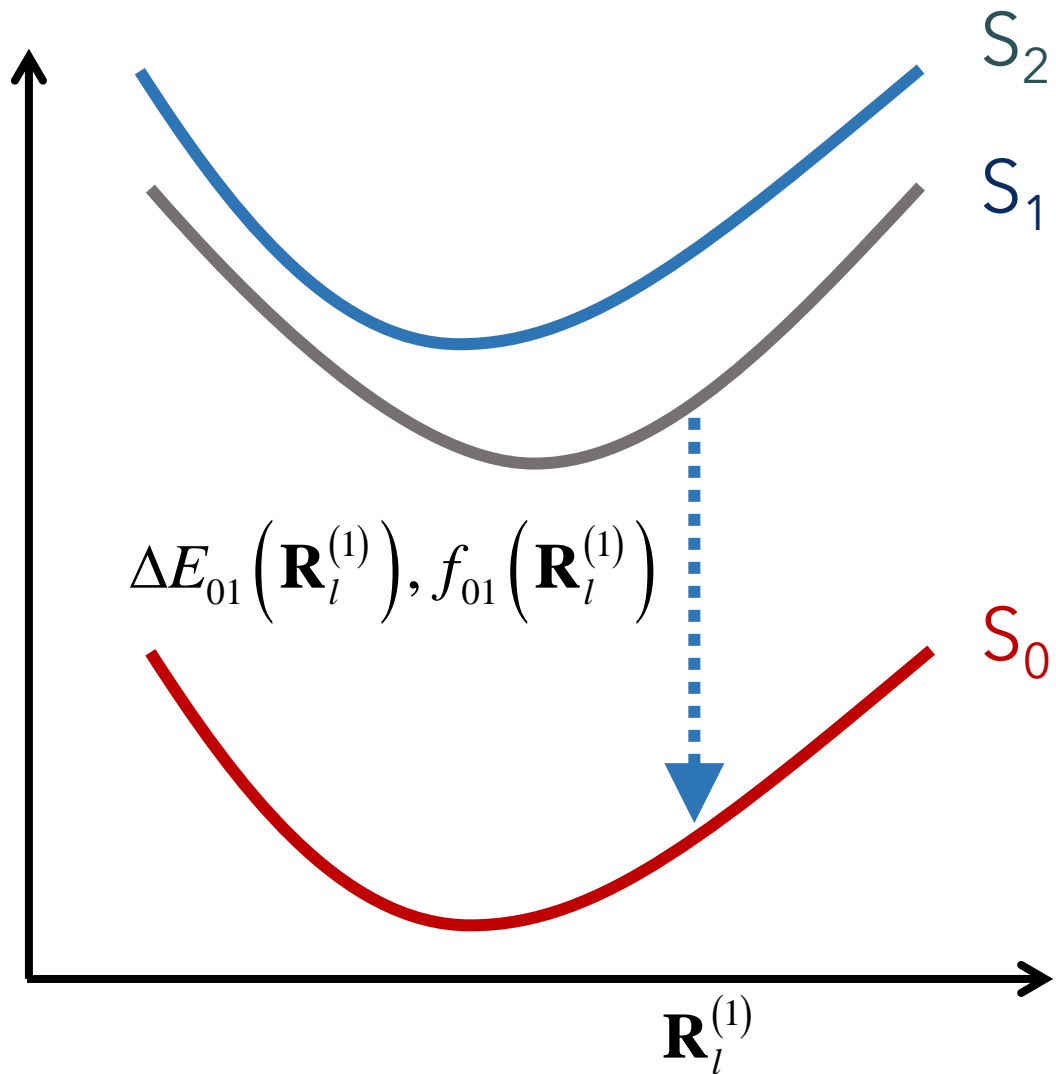
# Simulating non-Kasha fluorescence

$$\Gamma_{TOT}(E) = \sum_J \Gamma_{J \rightarrow 0}(E) \rho_{1J}$$

$\sum_J ( )$  Sum over electronic states  $J$

$\Gamma_{J \rightarrow 0}(E)$  Differential emission rate from  $S_J$  to  $S_0$

$\rho_{1J}$  Distribution of excited-state population  $J$   
(relative to  $S_1$ )



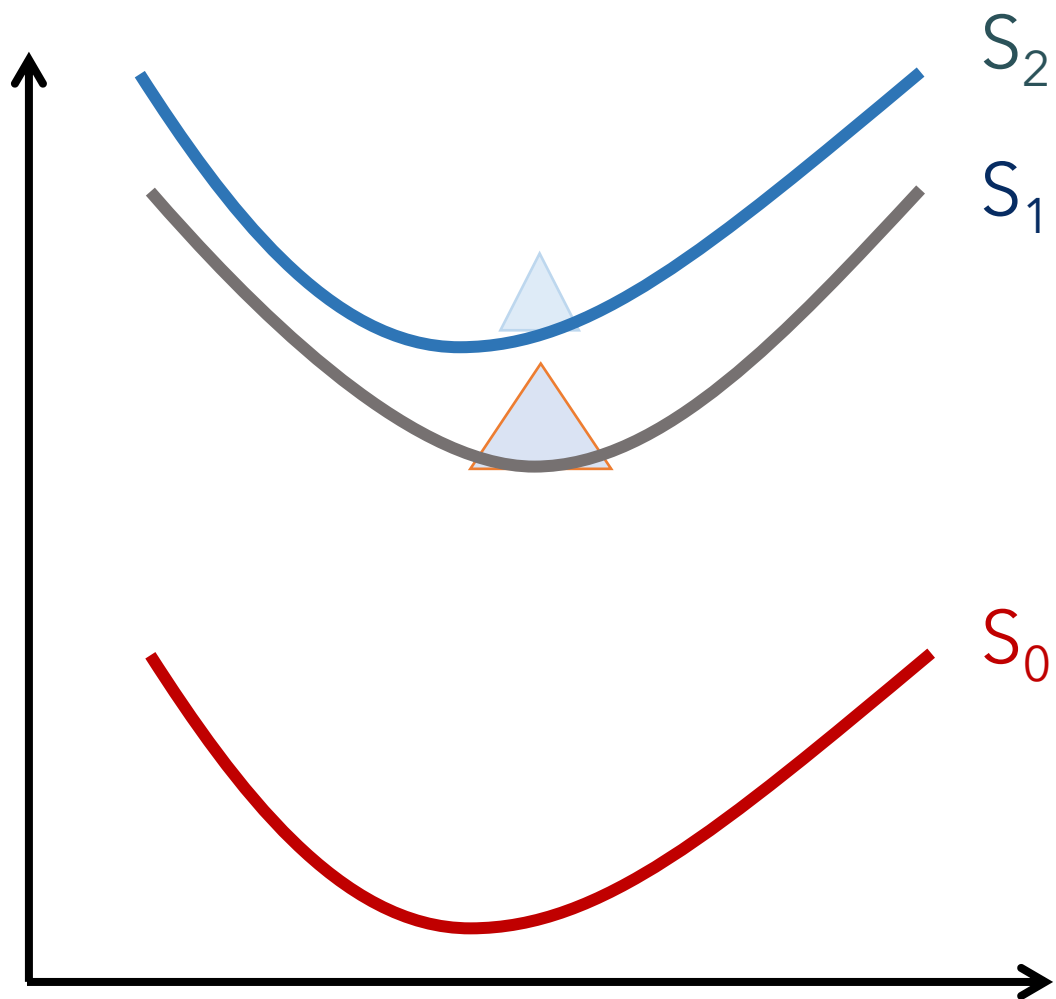
$$\Gamma_{TOT}(E) = \sum_J \Gamma_{J \rightarrow 0}(E) \rho_{1J}$$

In atomic units:

$$\begin{aligned} \Gamma_{J \rightarrow 0}(E) = & \frac{2}{c^3 N_p^{(J)}} \sum_l^{N_p^{(J)}} \Delta E_{0J}^2(\mathbf{R}_l^{(J)}) f_{0J}(\mathbf{R}_l^{(J)}) \\ & \times [1 - H(E - E_a)] \\ & \times g(E - \Delta E_{0J}(\mathbf{R}_l^{(J)}), \delta_J) \end{aligned}$$

$E_a$  - absorption energy

$H$  - Heaviside step function



$$\Gamma_{TOT}(E) = \sum_J \Gamma_{J \rightarrow 0}(E) \rho_{1J}$$

$$\rho_{1J} = \frac{N_p^{(J)}}{N_T}$$

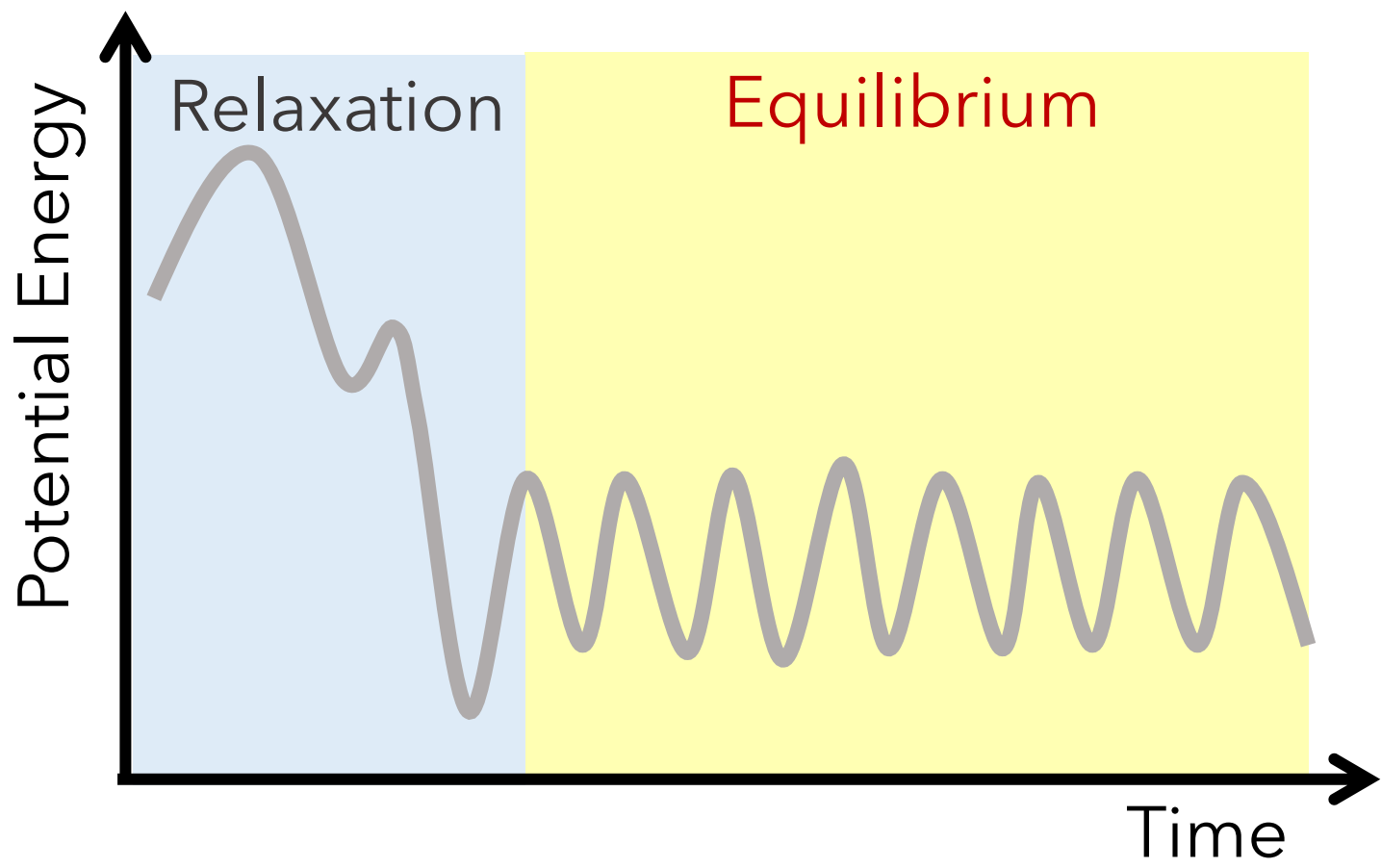
Number of points in  
state  $J$  during  
dynamics

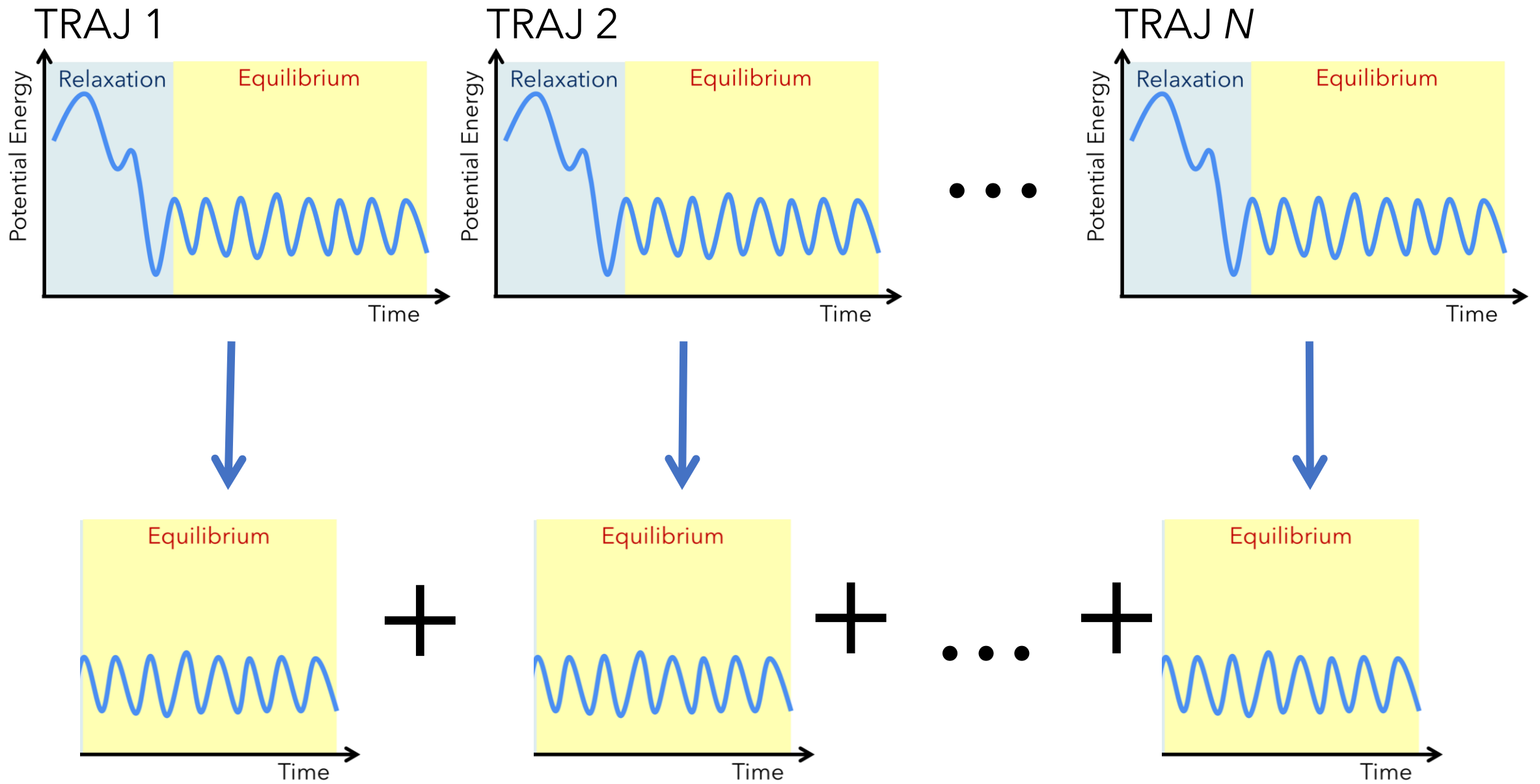
# Building the ensemble

Pyrene fluorescence: 100 ns - Dynamics is unaffordable

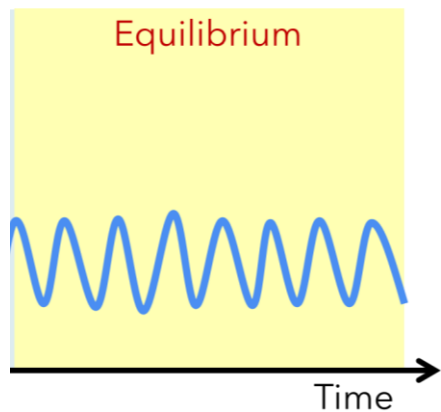
Ergodic protocol:

- Simulate many short surface hopping trajectories
- Discard initial relaxation
- Build the ensemble with the relaxed part of all trajectories
- Suppose this total cumulative time is representative of phase space occupation



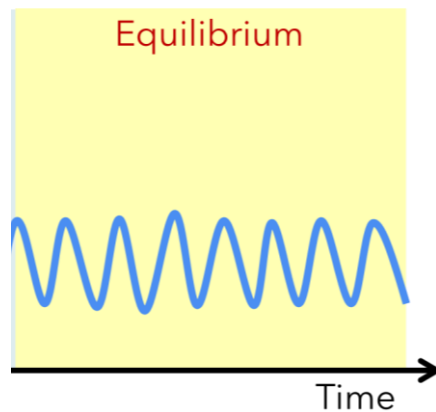


TRAJ 1



+

TRAJ 2

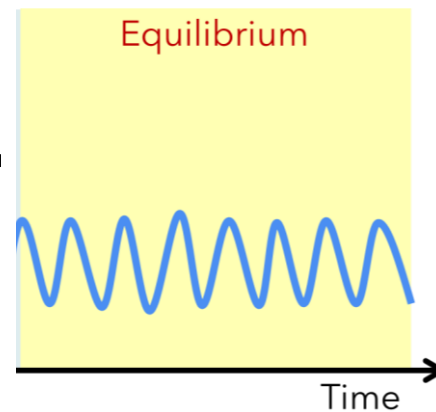


+

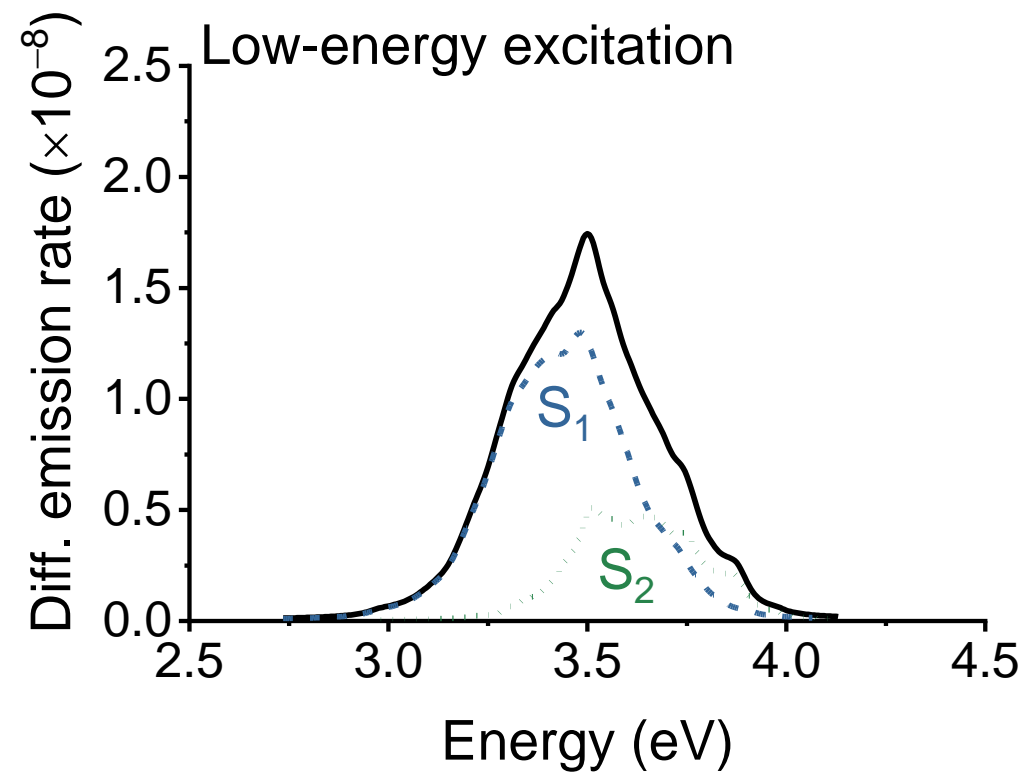
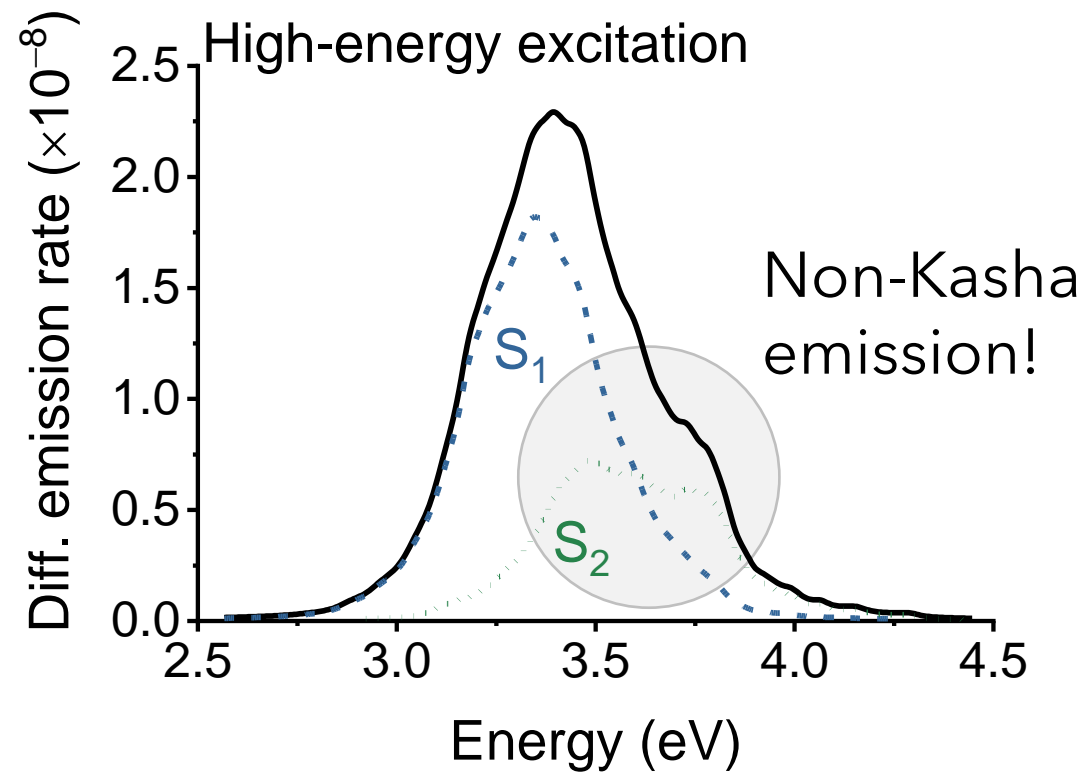
...

+

TRAJ N

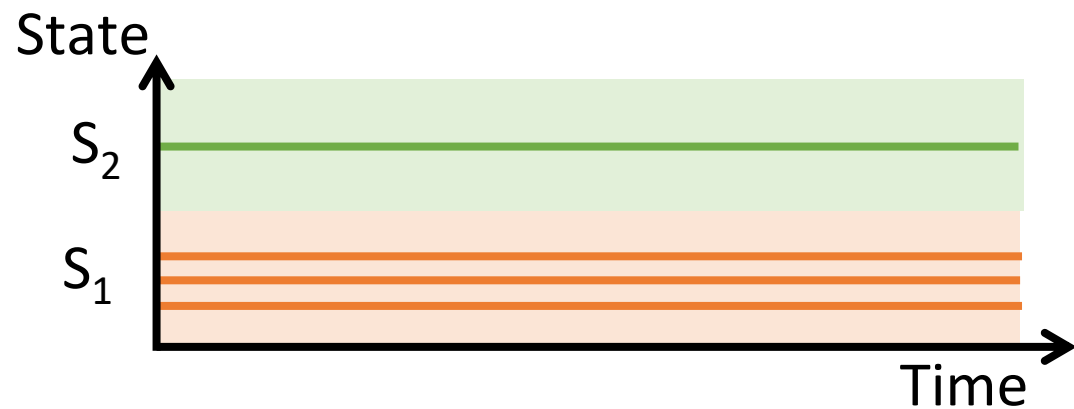


	Cumulative time
High-energy band	21 ps
Low-energy band	23 ps

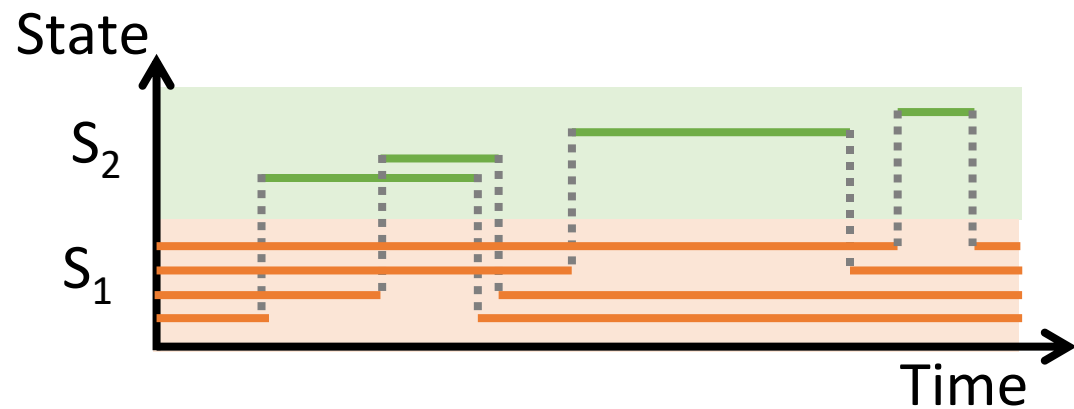




## Static equilibrium



## Dynamic equilibrium



$S_2$  occupation during dynamics:

- High-energy - 3.4%
- Low-energy - 1.6%

To know more:

Margins of error

- [tinyurl.com/kahnstat](https://tinyurl.com/kahnstat)
- [tinyurl.com/blognormal](https://tinyurl.com/blognormal)

To truly understand Gaussian errors: central limit theorem

- [www.youtube.com/watch?v=zeJD6dqJ5lo](https://www.youtube.com/watch?v=zeJD6dqJ5lo)

Statistical errors in MD

- Schiferl; Wallace. *J Chem Phys* **1985**, 83, 5203

The most important statistical ideas in the last 50 years

- Gelman; Vehtari. *J Am Stat Assoc* **2021**, 116, 2087

Spectrum simulations

- Eric J Heller, *The semiclassical way*, **2018**. Ch 17

Papers available for download at:

[amubox.univ-amu.fr/s/xXAiMZrDPb9RMRX](https://amubox.univ-amu.fr/s/xXAiMZrDPb9RMRX) (Ask me for the password)