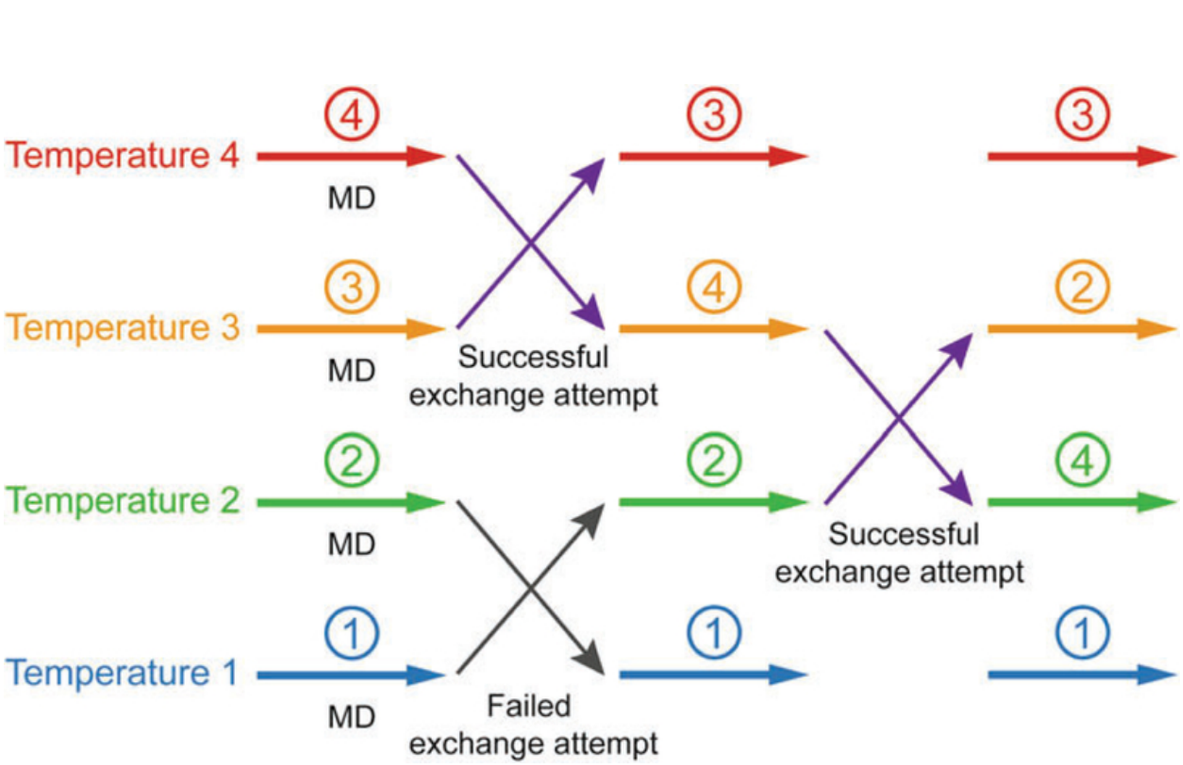
Project 4

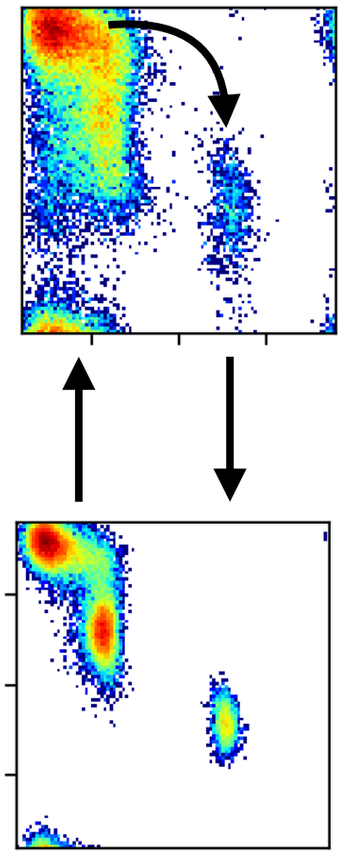
REMD: the replica exchange MD simulations

Assignments

- Write the code to do replica exchange in REMD
  - You only need to write the code to do exchanges
- Do REMD simulations with your own code for alanine dipeptide in vacuum
  - Combine your replica-exchange code with GROMACS (or any other MD package)
- Compute the free energy landscape

The replica exchange MD





Exchange:

$$p_n^{[j]'} = \sqrt{\frac{T_n}{T_m}} p_m^{[i]}$$

$$p_m^{[i]'} = \sqrt{\frac{T_m}{T_n}} p_n^{[j]}$$

$$x_n^{[j]'} = x_m^{[i]}$$

$$x_m^{[i]'} = x_n^{[j]}$$

$$\frac{w(X \to X')}{w(X' \to X)} \equiv \frac{w(x_m^{[i]} \to x_n^{[j]})}{w(x_n^{[j]} \to x_m^{[i]})} = e^{-(\beta_n - \beta_m)(V_i - V_j)}$$

Metropolis criterion: $P_{\text{exchange}} = \min\{1, e^{-(\beta_n - \beta_m)(V_i - V_j)}\}$

R. Qi, G. Wei, B. Ma, R. Nassinov, ***Methods in Molecular Biology***, Peptide Self-Assembly 101–119 (2019)
R. Zhou, ***Methods in Molecular Biology***, Protein Folding Protocols 205-223 (2007)
Alexandra Patriksson and David van der Spoel, ***PCCP*** **10**, 2073–2077 (2008)

Note: Qi's and Patriksson's papers were correct. Zhou's Δ were wrong, should be $-\Delta$

Replica-exchange code based on GROMACS

```
project4-remd/remd$ cat do-remd.sh
bash step0/run-md.sh >& /dev/null
for i in `seq 0 999`; do
    step=$i ./do-exchange.sh >> remd.log
    bash step$((i+1))/run-md.sh >& /dev/null
done

project4-remd/remd$ cat do-exchange.sh
#!/bin/bash
gmx=$HOME/bin/gromacs-2022/bin/gmx
number_of_replicas=8
...
...
for i in `seq 1 $number_of_replicas`; do
    if [ ! -e step$((step+1))/$i.tpr ]; then
        $gmx grompp -c step$((step+1))/ex.$i.gro -p ../src/ala2.top -f ../tprs/remd-$i.mdp -o step$
((step+1))/$i.tpr >& /dev/null
    fi
done
ls step$((step+1))/*tpr | sed s/.tpr//g | while read fn; do
    echo $gmx mdrun -deffnm $fn
    echo echo 10 11 \| ~/bin/gromacs-2022/bin/gmx energy -f $fn.edr -o $fn.xvg
done > step$((step+1))/run-md.sh
...

project4-remd/remd$ cat step213/run-all.sh
gmx mdrun -deffnm step213/1
echo 10 11 | gmx energy -f step213/1.edr -o step213/1.xvg
gmx mdrun -deffnm step213/2
echo 10 11 | gmx energy -f step213/2.edr -o step213/2.xvg
...
```

Tools in GROMACS for this project

Generate TPR to run MD simulations:

```
gmx grompp -f md.mdp -c conform.gro -p topol.top -o md.tpr
```

Run MD simulations with replica_1.tpr**:**

```
gmx mdrun -deffnm replica_1
```

Combine trajectories to a single trajectory:

```
gmx trjcat -f step?/1.xtc step??/1.xtc step???/1.xtc -o traj1.xtc -cat
```

Compute the Ramachandran angles from MD trajectory replica_1.xtc**:**

```
gmx rama -s md.tpr -f replica_1.xtc -o replica_1.xvg
```

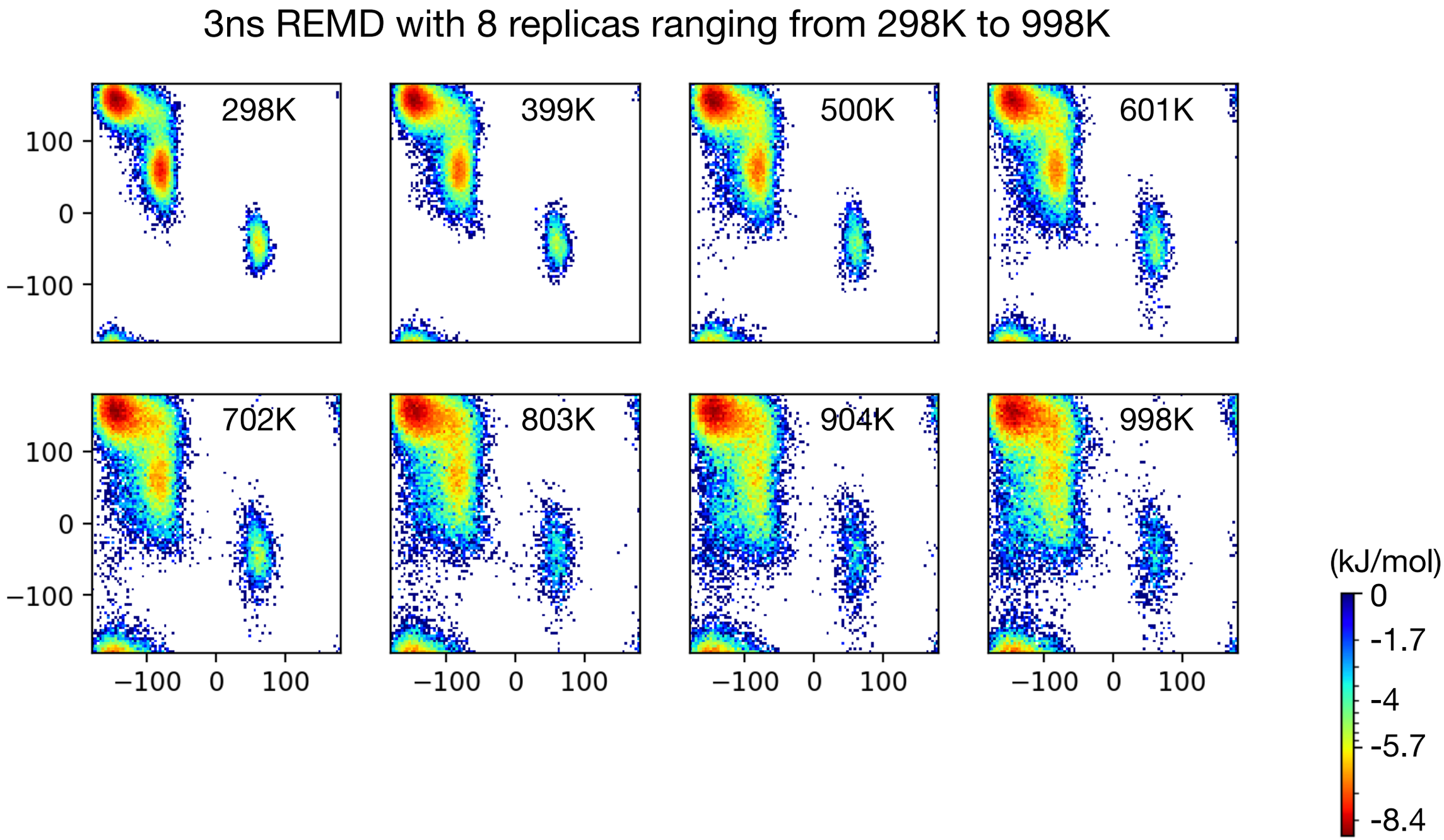
Scale velocities in a GRO file:

```
Protein
   22
    1ACE   HH31    1   1.429   1.465   1.117 -0.2640  1.8826 -1.4574
    1ACE    CH3    2   1.457   1.391   1.042 -0.5754  0.1883 -0.0009
... ...
    3NME   HH32   21   0.834   1.247   1.367  0.6916 -0.3557 -1.7130
    3NME   HH33   22   0.774   1.092   1.317 -0.6368 -0.0776 -0.0320
   3.00000   3.00000   3.00000

awk: substr($0,1,44) : index, molecule, atom and xyz coordinates
awk: substr($0,45,8) : vx in %8.4 format
awk: substr($0,53,8) : vy in %8.4 format
awk: substr($0,61,8) : vz in %8.4 format
```

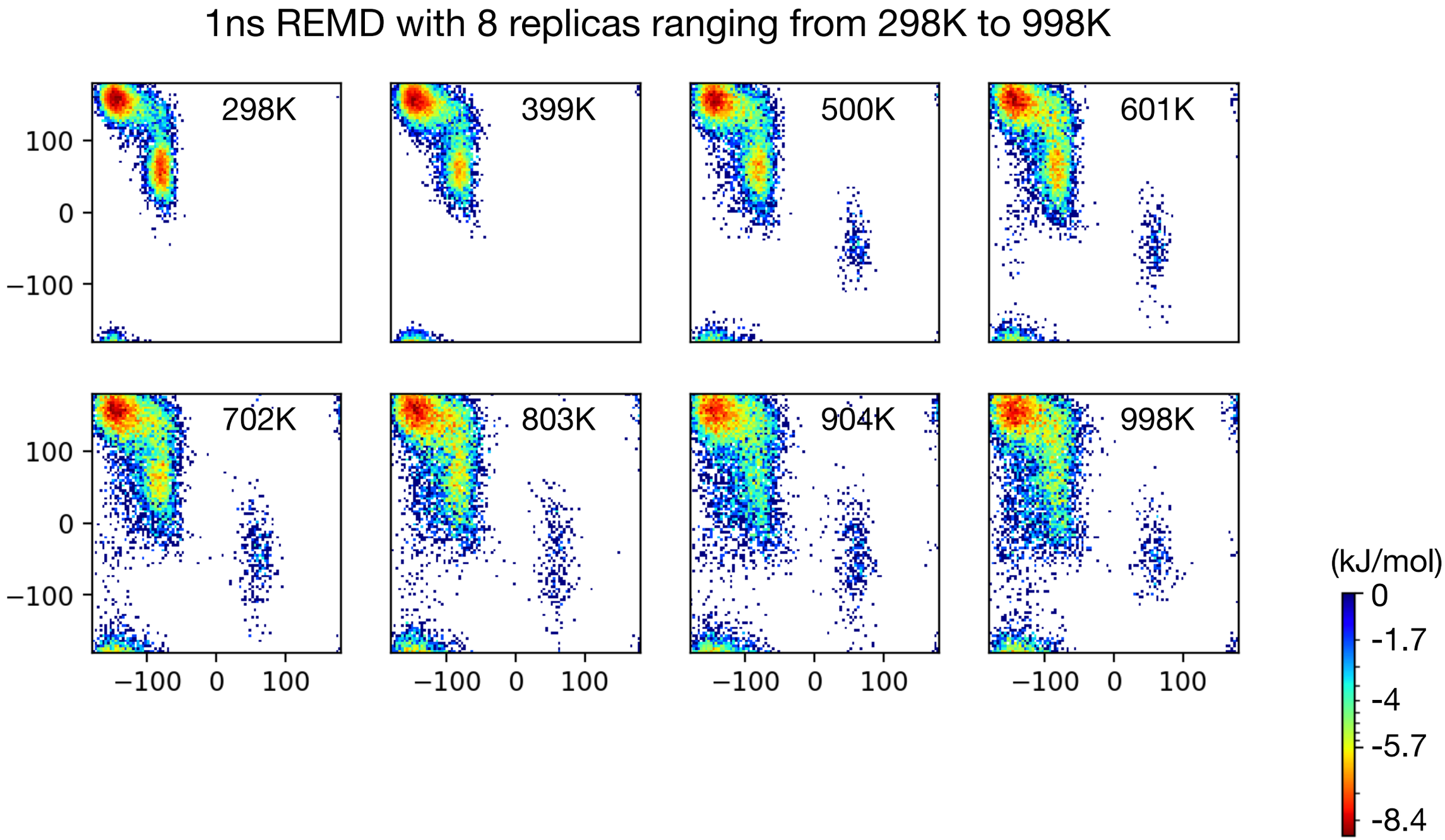
Replica-exchange for alanine dipeptide in vacuum

- A well sampled REMD can efficiently find all metastable states



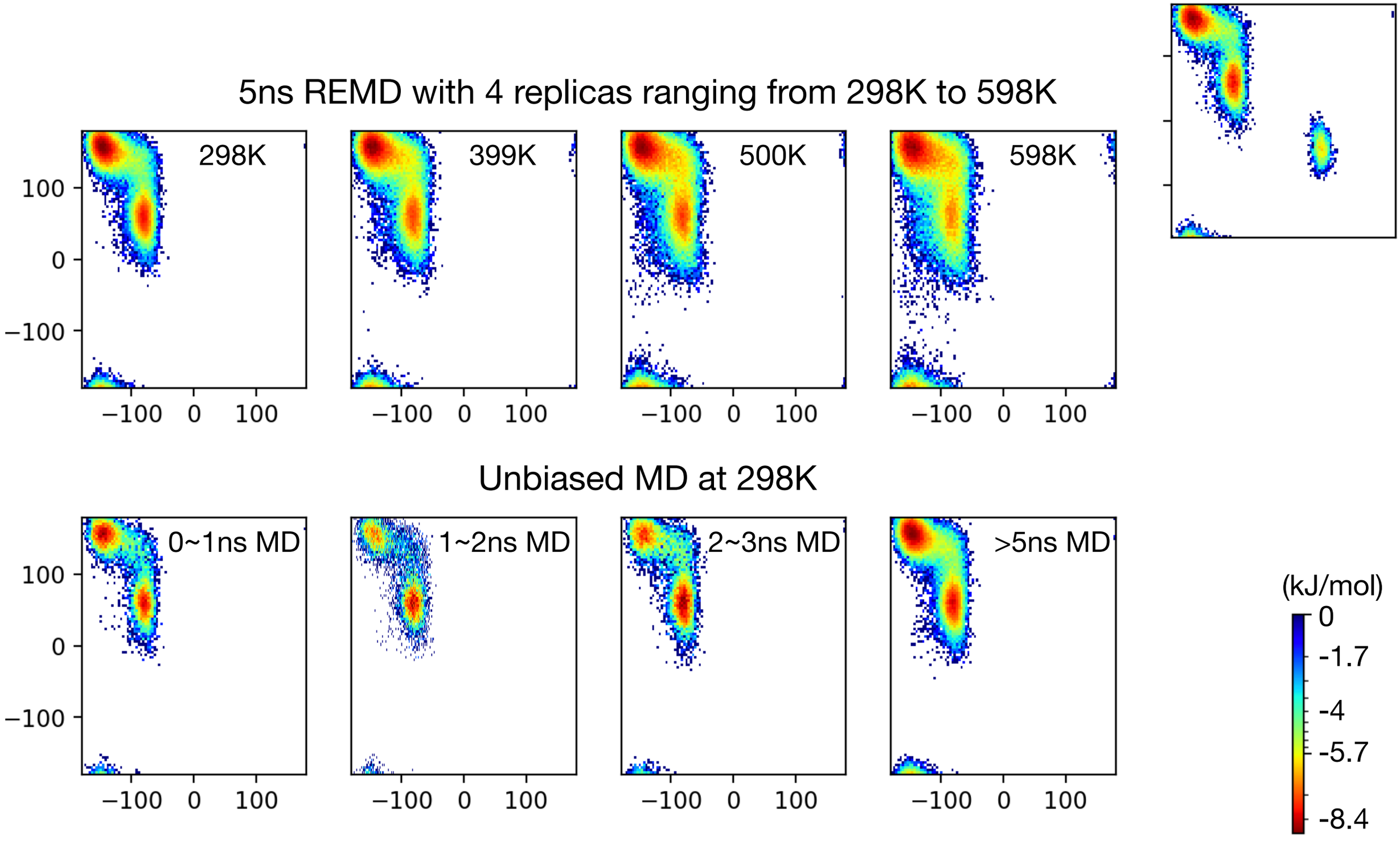
Replica-exchange for alanine dipeptide in vacuum

- A well sampled REMD can efficiently find all metastable states



Replica-exchange for alanine dipeptide in vacuum

- A insufficient sampling REMD will miss important states



** Advanced topic: temperatures of REMD

- Average probability of exchange:

$$\langle P(T_1 \leftrightarrow T_2)\rangle = \int_{-\infty}^{\infty} P(T_1 \leftrightarrow T_2)\rho_{U_1-U_2}(u)du = \int_{-\infty}^{0} \rho_{U_1-U_2}(u)du + \int_{0}^{\infty} e^{(\beta_1-\beta_2)u}\rho_{U_1-U_2}(u)du$$

$$\rho_{U_1-U_2}(u) = \frac{1}{\sigma_{12}\sqrt{2\pi}} e^{-(u-\mu_{12})^2/2\sigma_{12}^2} \qquad \mu_{12} = \mu_1 - \mu_2 \qquad \sigma_{12} = \sqrt{\sigma_1^2 + \sigma_2^2}$$

$$\langle P(T_1 \leftrightarrow T_2)\rangle = \frac{1}{2}\left[1 + \mathrm{erf}\left(-\frac{\mu_{12}}{\sigma_{12}\sqrt{2}}\right)\right] + \frac{1}{2} e^{(\beta_1-\beta_2)\mu_{12}+(\beta_1-\beta_2)^2\sigma_{12}^2/2}\left[1 + \mathrm{erf}\left(\frac{\mu_{12}+(\beta_1-\beta_2)\sigma_{12}^2}{\sigma_{12}\sqrt{2}}\right)\right]$$

Exchange probability:		Tolerance:	1e-4
Lower temperature limit:		Upper temperature limit:	
Number of water molecules:		Constraints in water:	Fully Flexible
Number of protein atoms:		Constraints in the protein:	Fully Flexible
Hydrogens in protein:	All H	Virtual sites in protein:	None
Simulation type:	NPT		

Alexandra Patriksson and David van der Spoel, ***PCCP* 10**, 2073–2077 (2008)
https://virtualchemistry.org/remd-temperature-generator/tgenerator.php