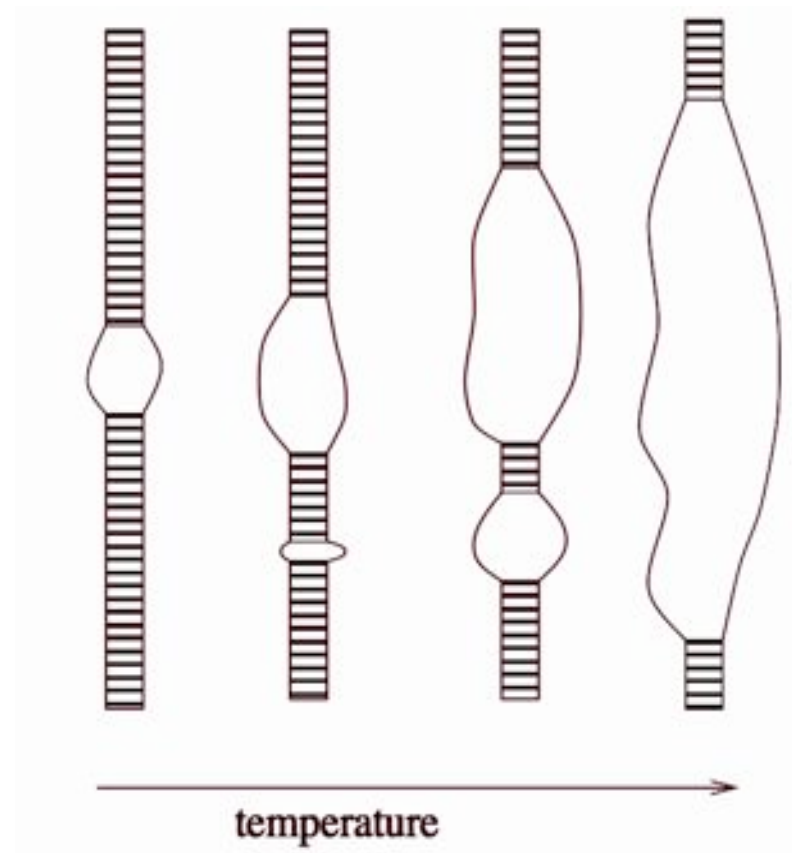


cours 2 : thermodynamique de l'ADN

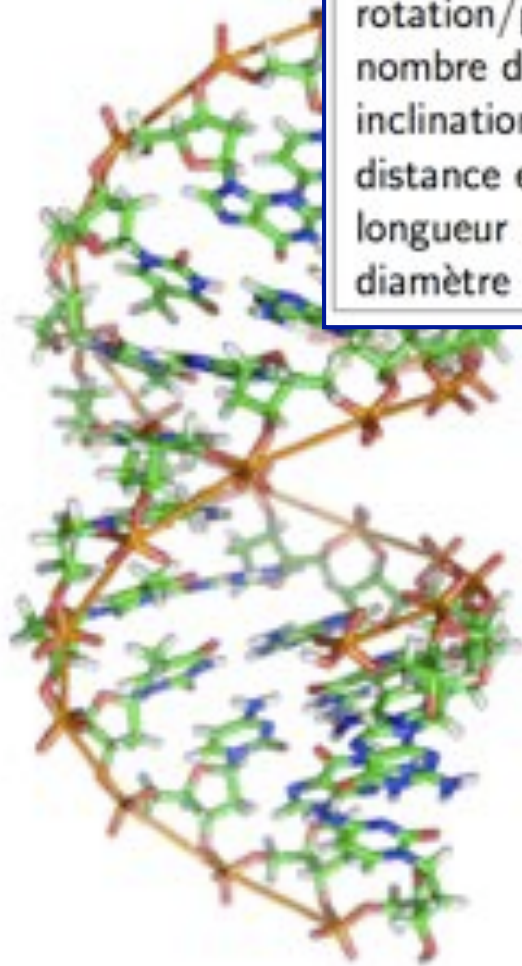


I - la molécule d'ADN



1. l'ADN en double hélice

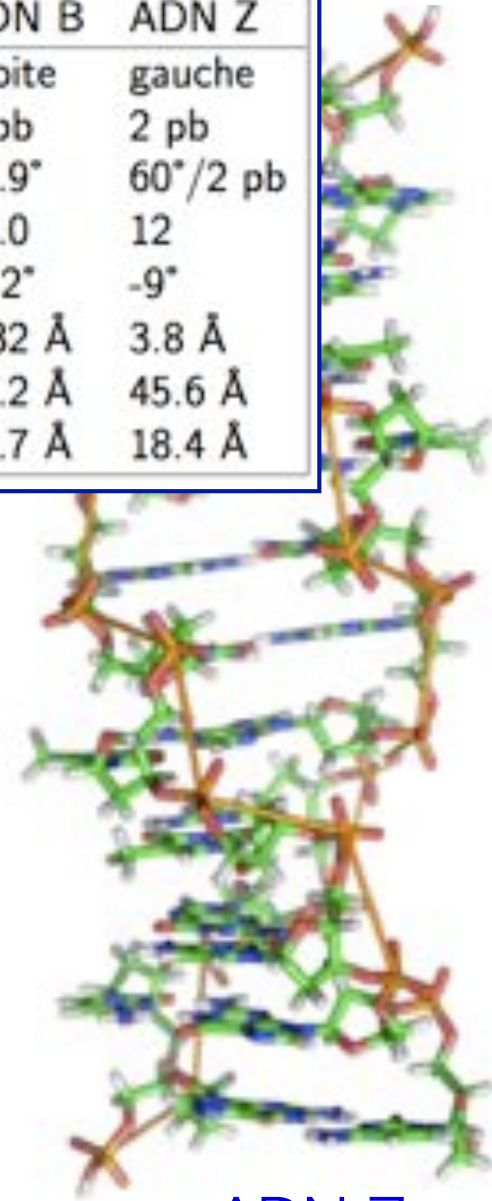
Caractéristique géométrique	ADN A	ADN B	ADN Z
hélicité	droite	droite	gauche
unité de répétition	1 pb	1 pb	2 pb
rotation/pb	33.6°	35.9°	60°/2 pb
nombre de pb par période	10.7	10.0	12
inclination pb par rapport à l'axe	+19°	-1.2°	-9°
distance entre pb le long de l'axe (<i>rise</i>)	2.3 Å	3.32 Å	3.8 Å
longueur par période (<i>pitch</i>)	24.6 Å	33.2 Å	45.6 Å
diamètre	25.5 Å	23.7 Å	18.4 Å



ADN A



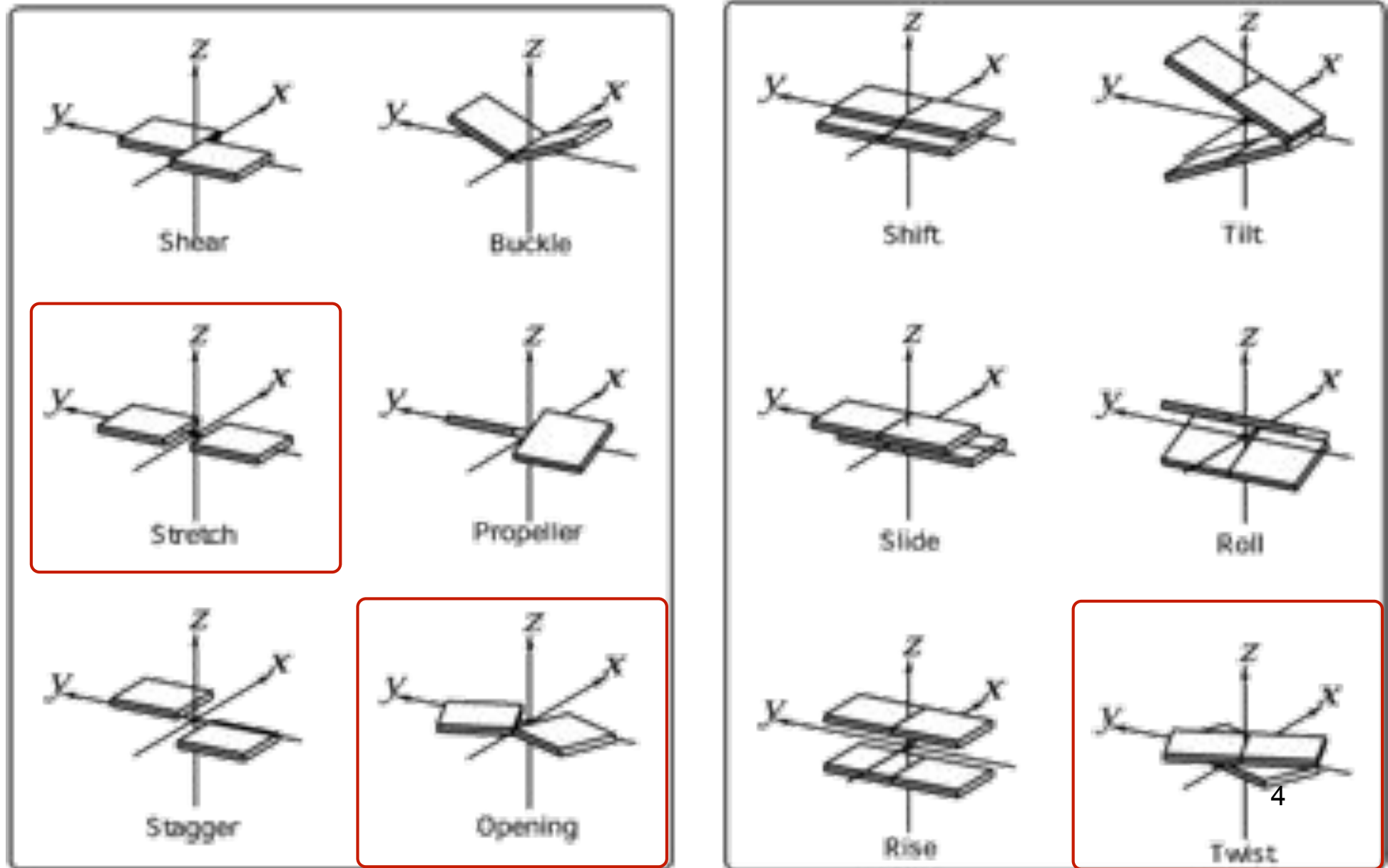
ADN B



ADN Z 3

1. l'ADN en double hélice

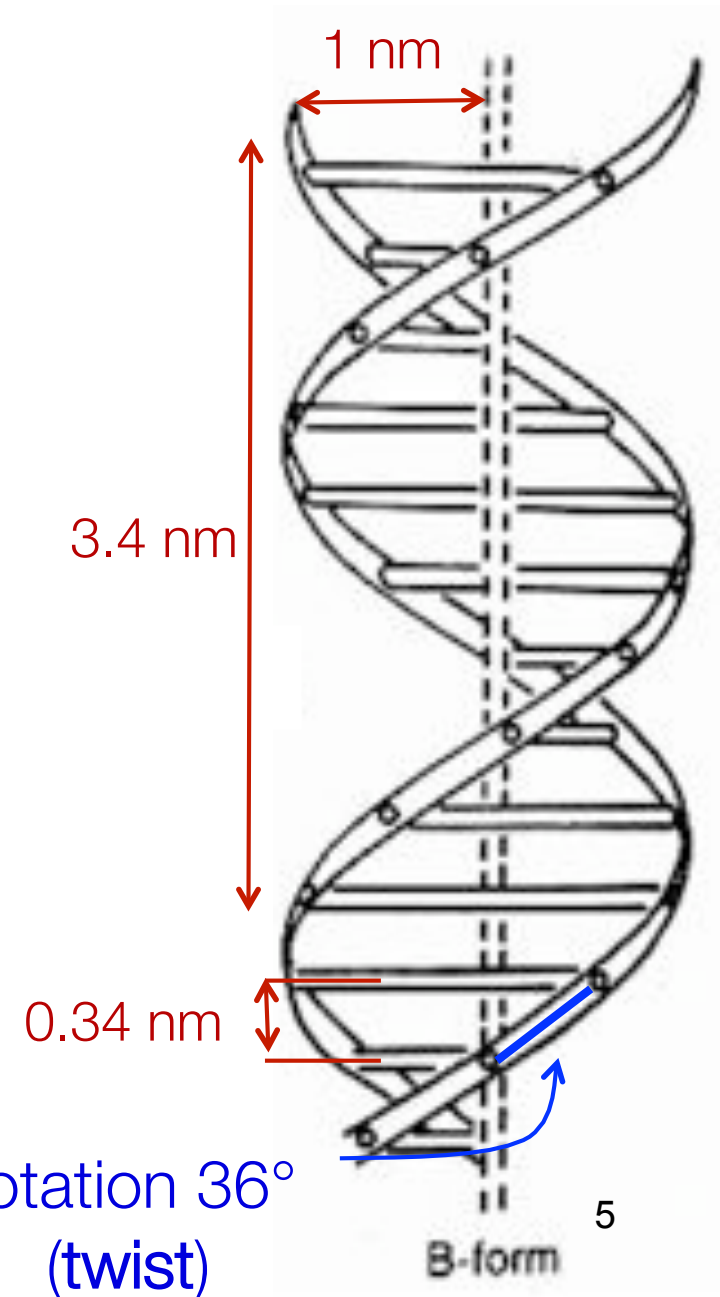
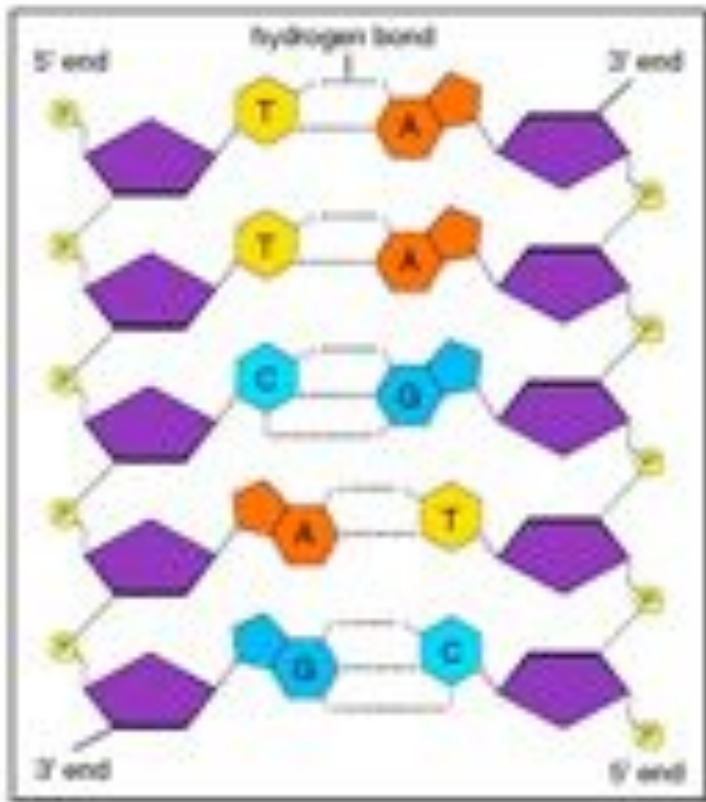
géométrie des paires de bases - la nomenclature officielle



1. l'ADN en double hélice

interactions d'empilement ([stacking](#))
(hydrophobicité des bases +...)

→ structure en double hélice

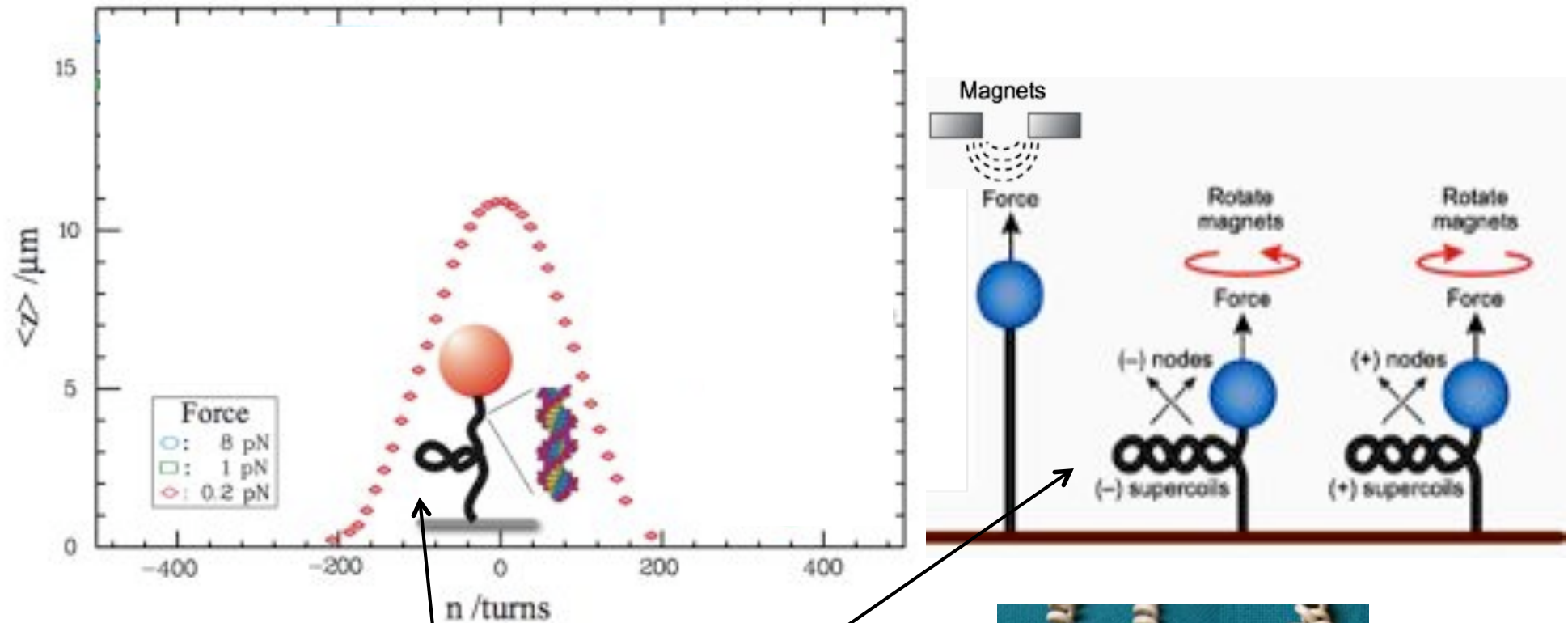


II - la dénaturation de l'ADN



Dénaturation mécanique de l'ADN

1. Tordre l'ADN avec des pinces magnétiques

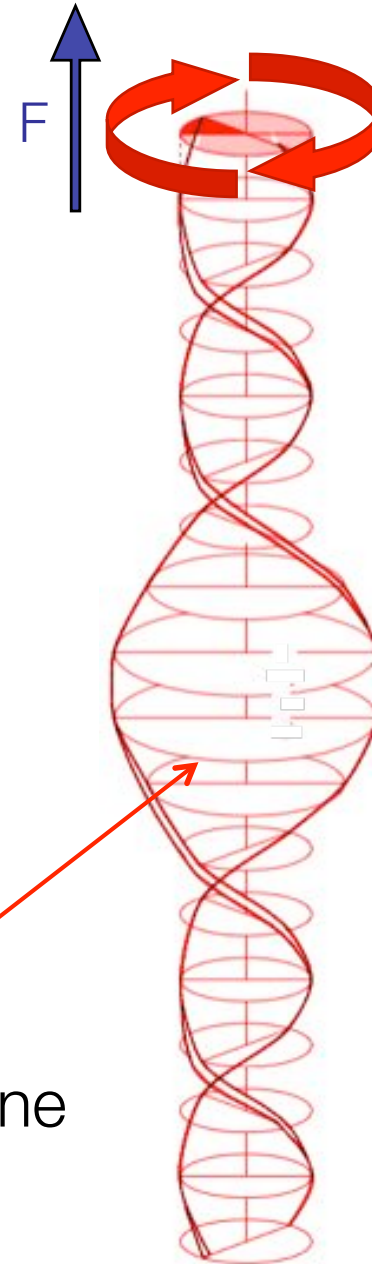
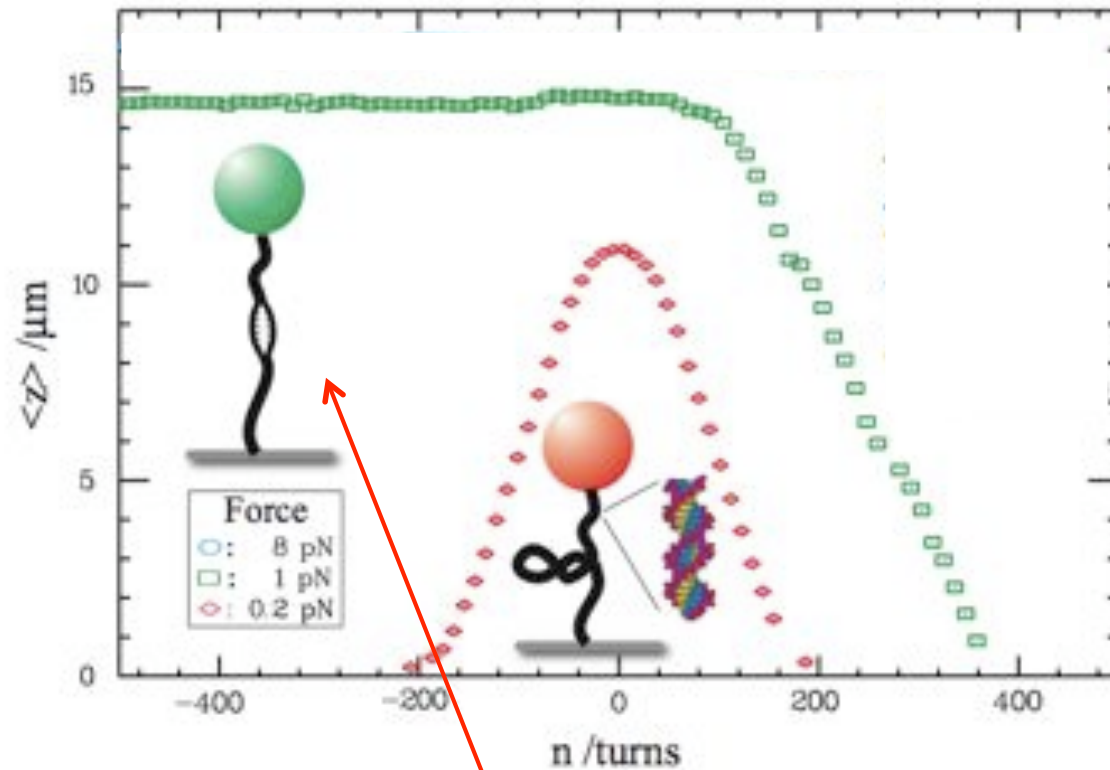


formation de plectonemes =



Dénaturation mécanique de l'ADN

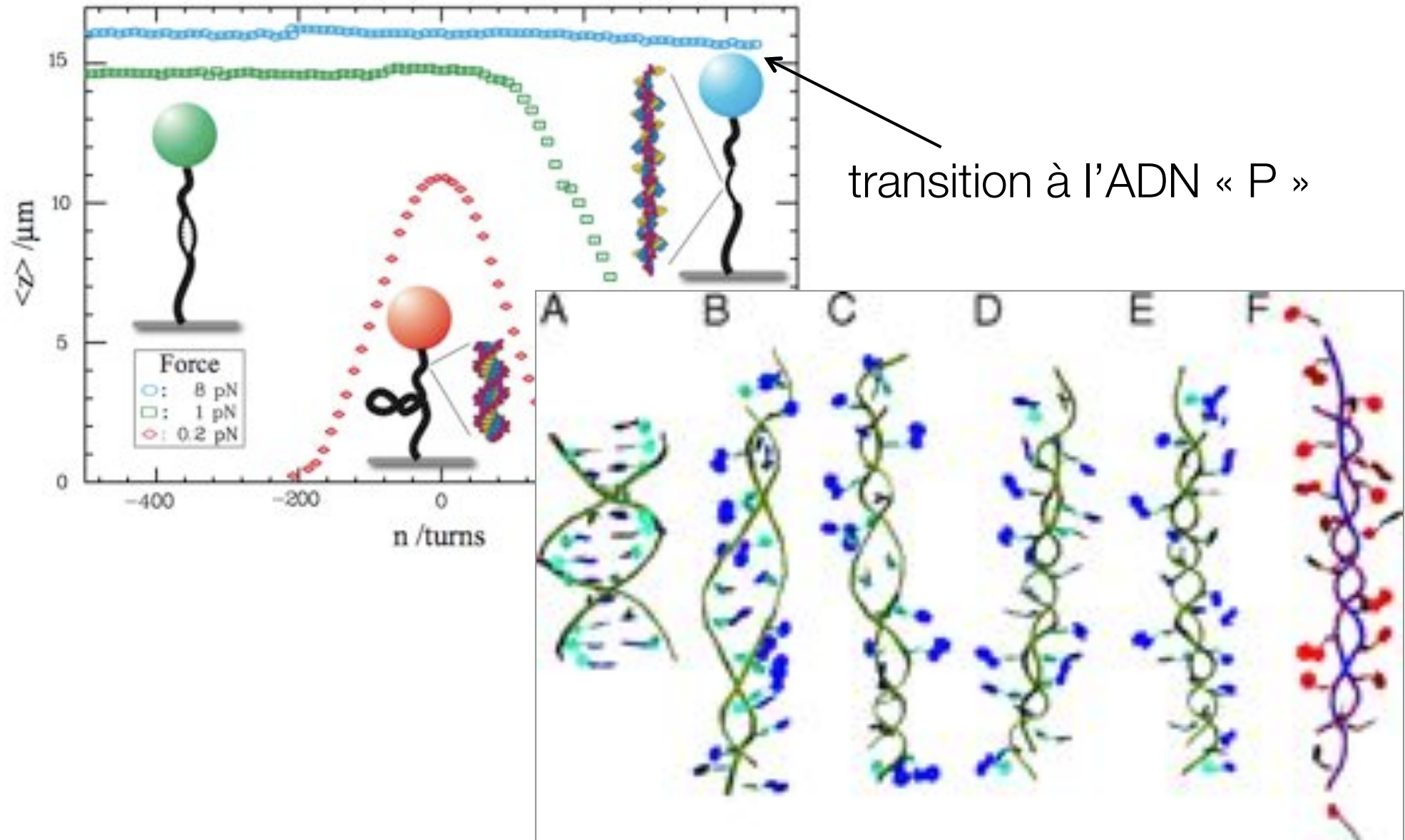
2. Torsion négative et force élevée :



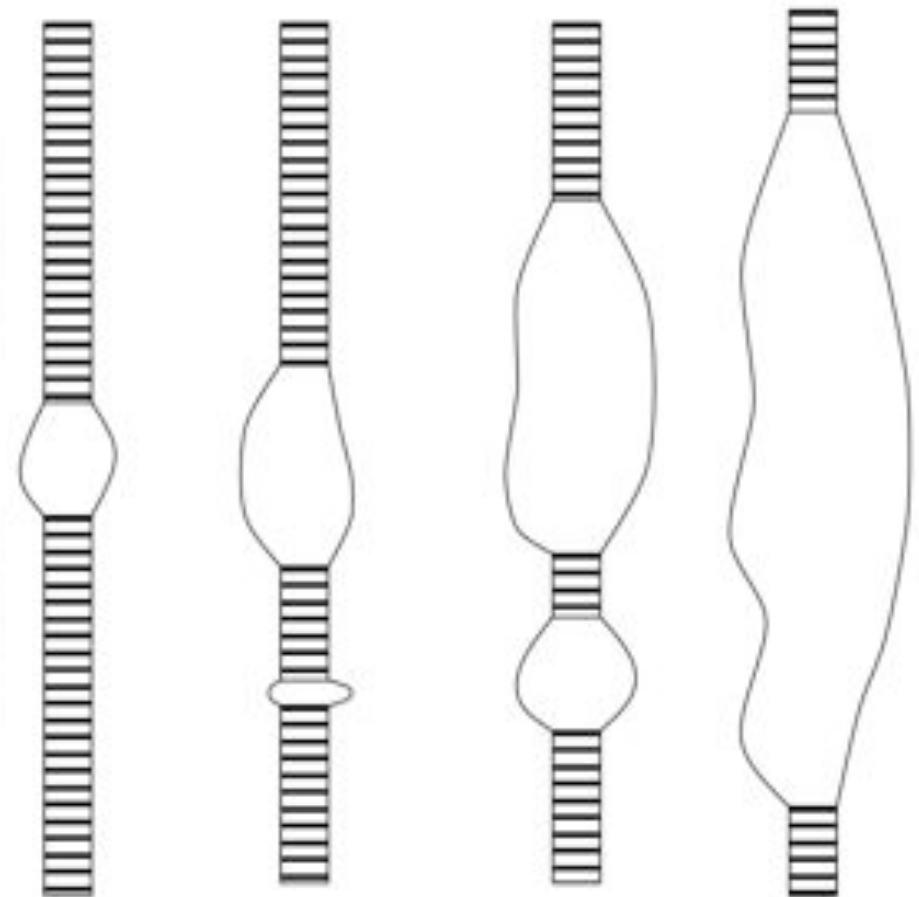
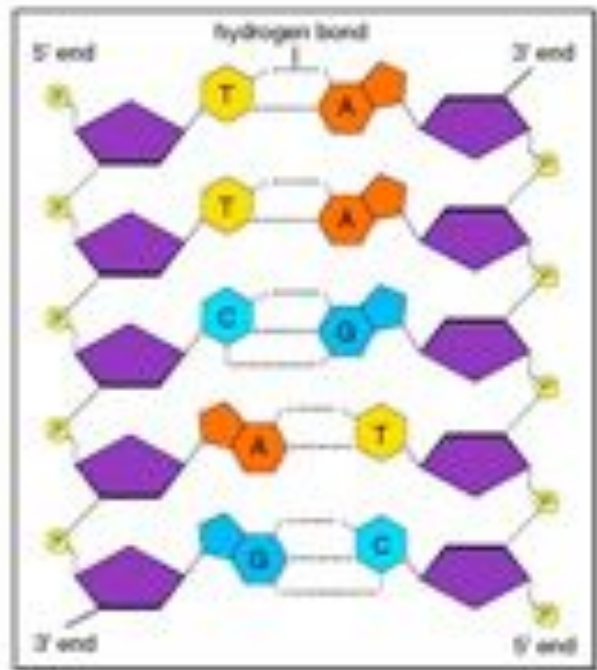
la torsion exercée sur l'ADN étiré est suffisante pour casser les liaisons hydrogène et séparer les 2 brins

Dénaturation mécanique de l'ADN

3. (Force très élevée) :



Dénaturation thermique de l'ADN



énergie d'une liaison hydrogène :
de l'ordre de $1 k_B T$

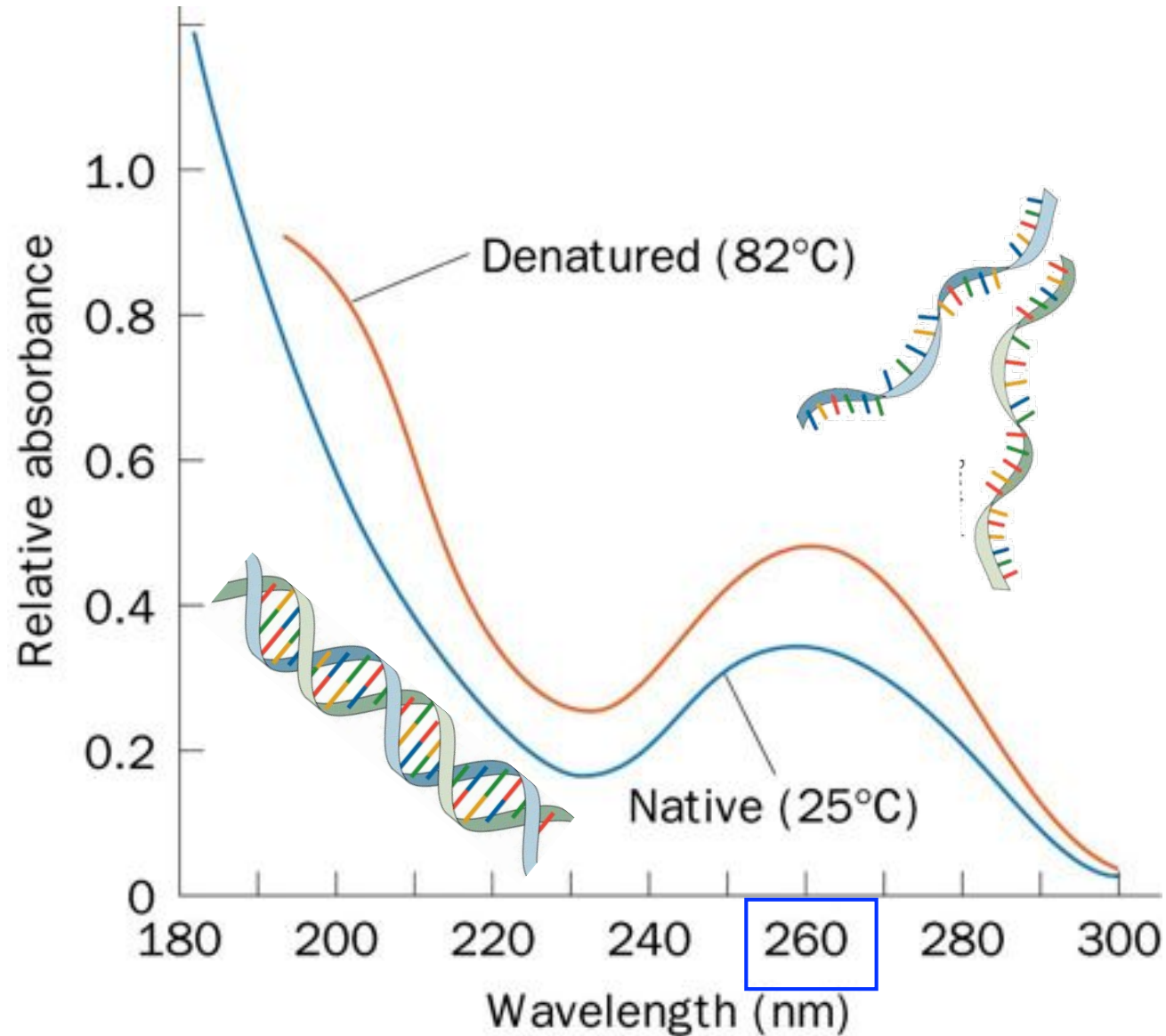
(= 0.0256 eV = 0.6 kcal/mol)

temperature

10

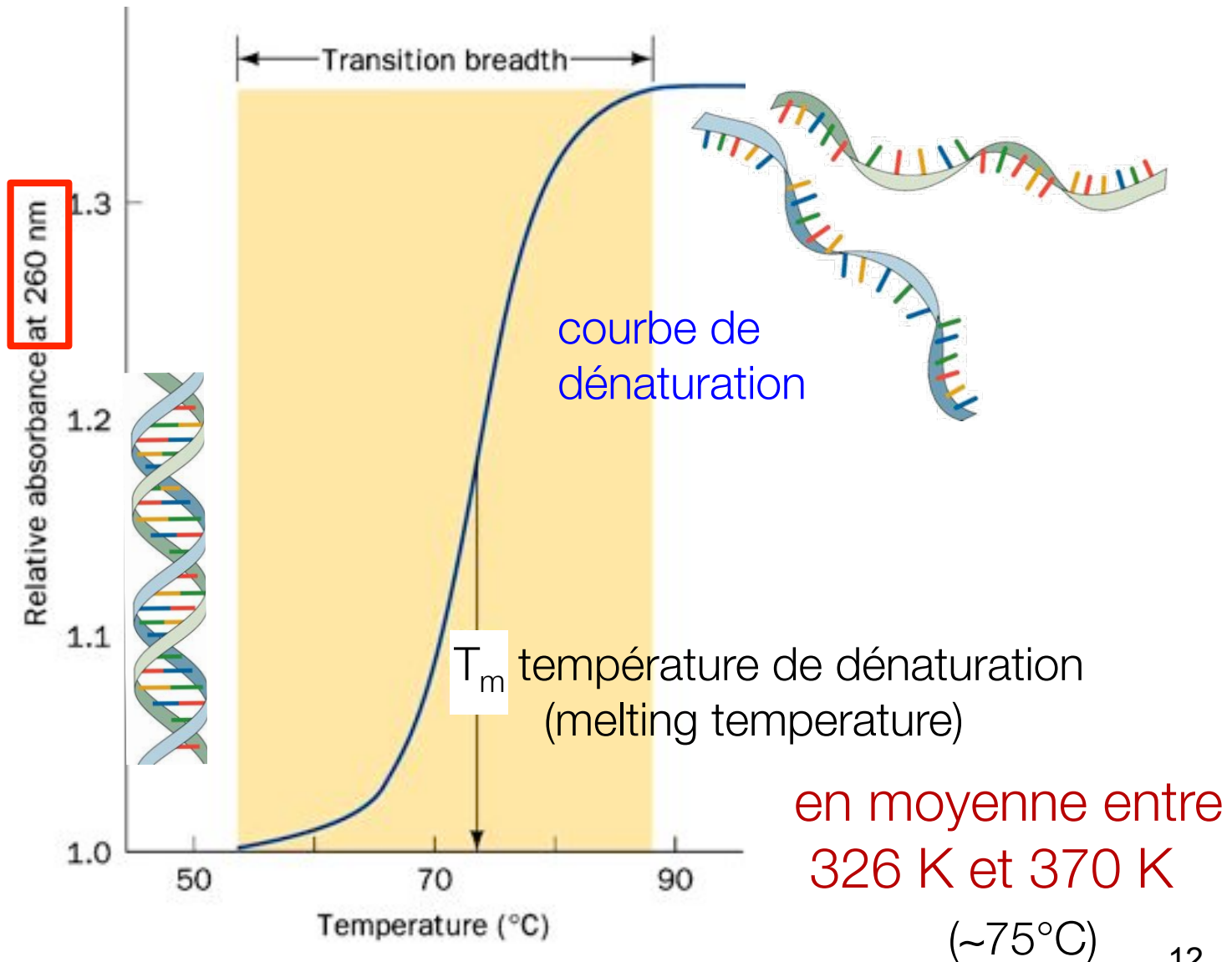
courbe de dénaturation

absorbance UV d'une solution d'ADN



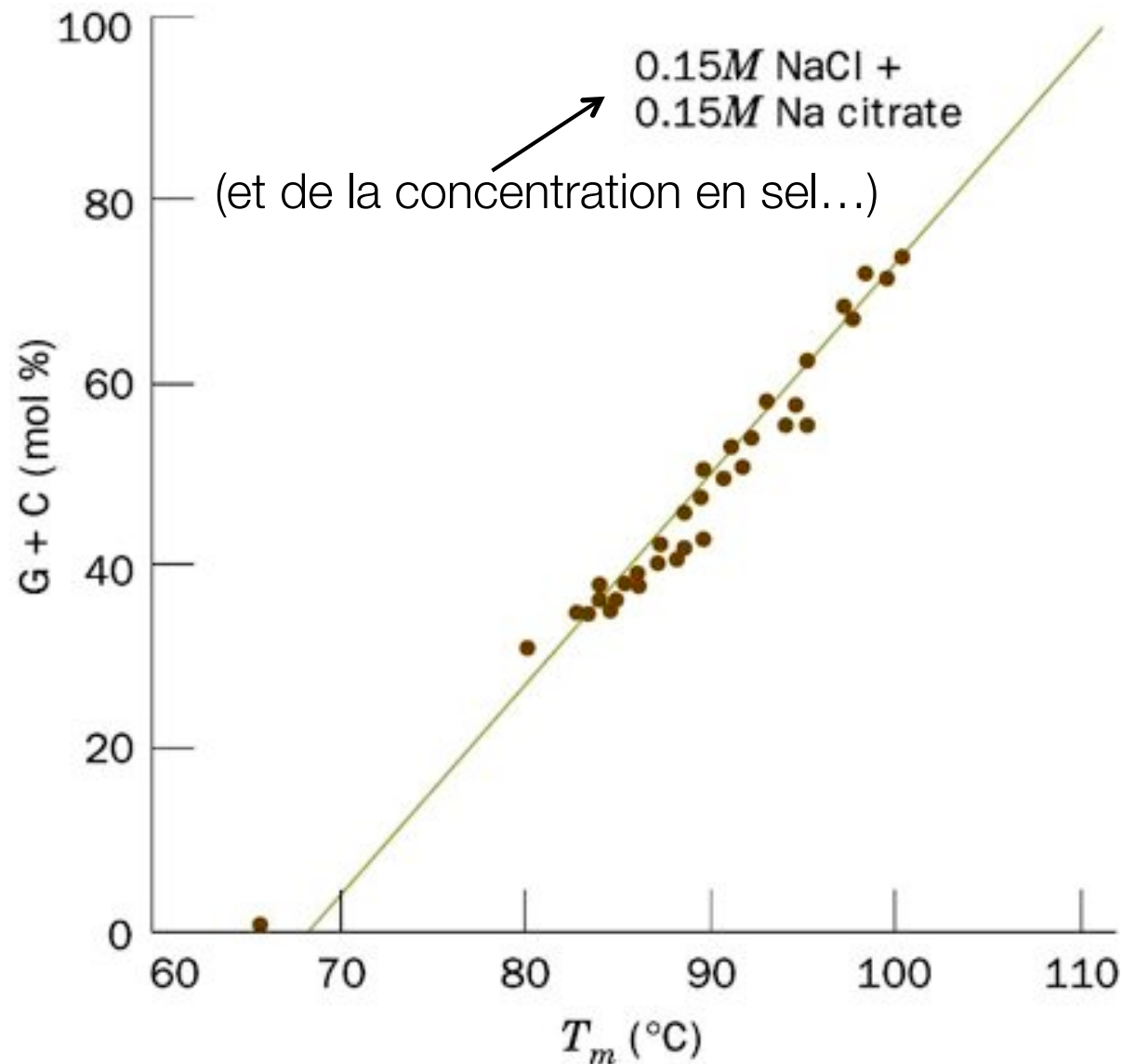
courbe de dénaturation

absorbance UV d'une solution d'ADN



courbe de dénaturation

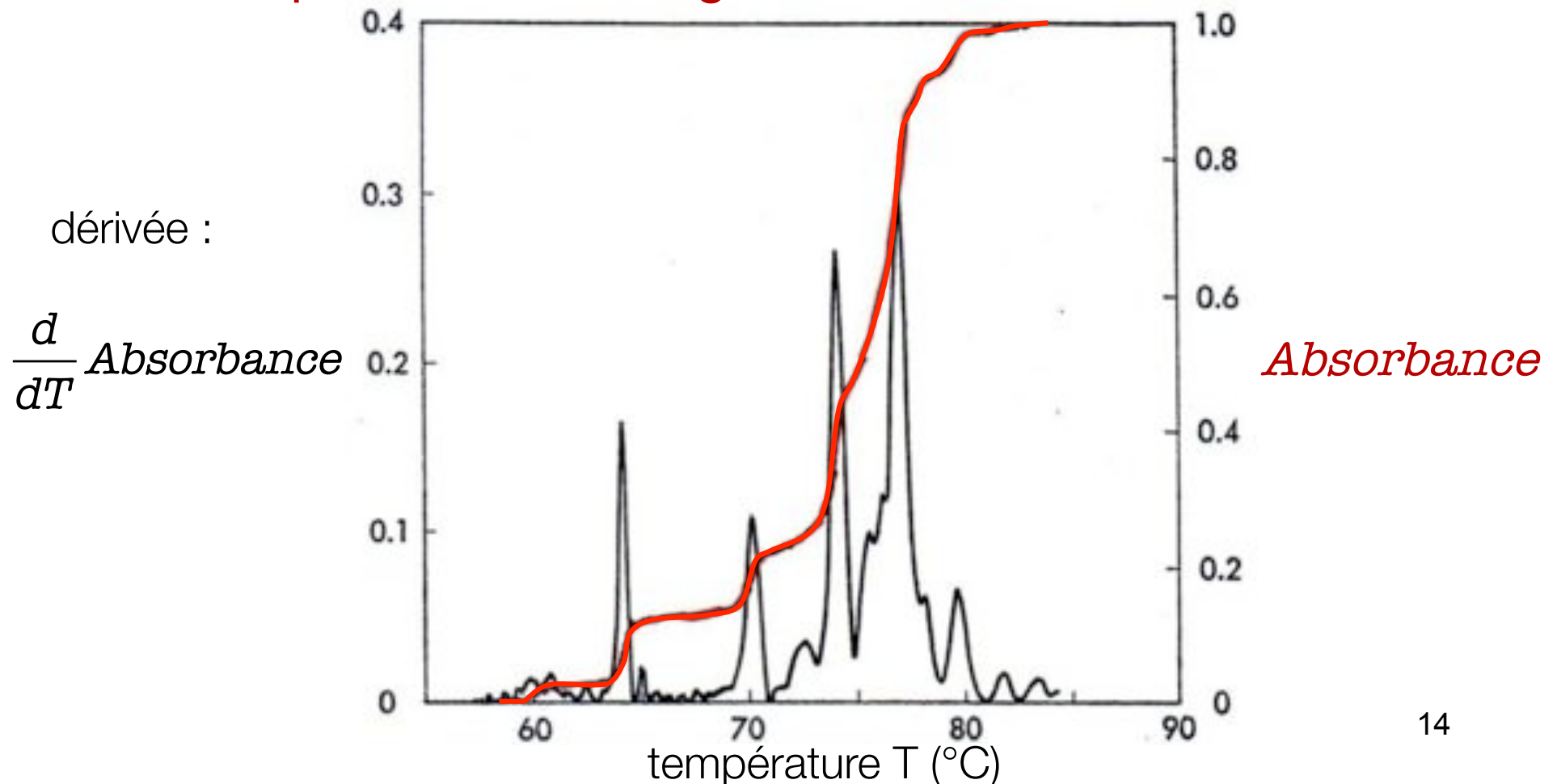
la température de dénaturation dépend
de la **concentration en GC** (triple liaison)



courbe de dénaturation

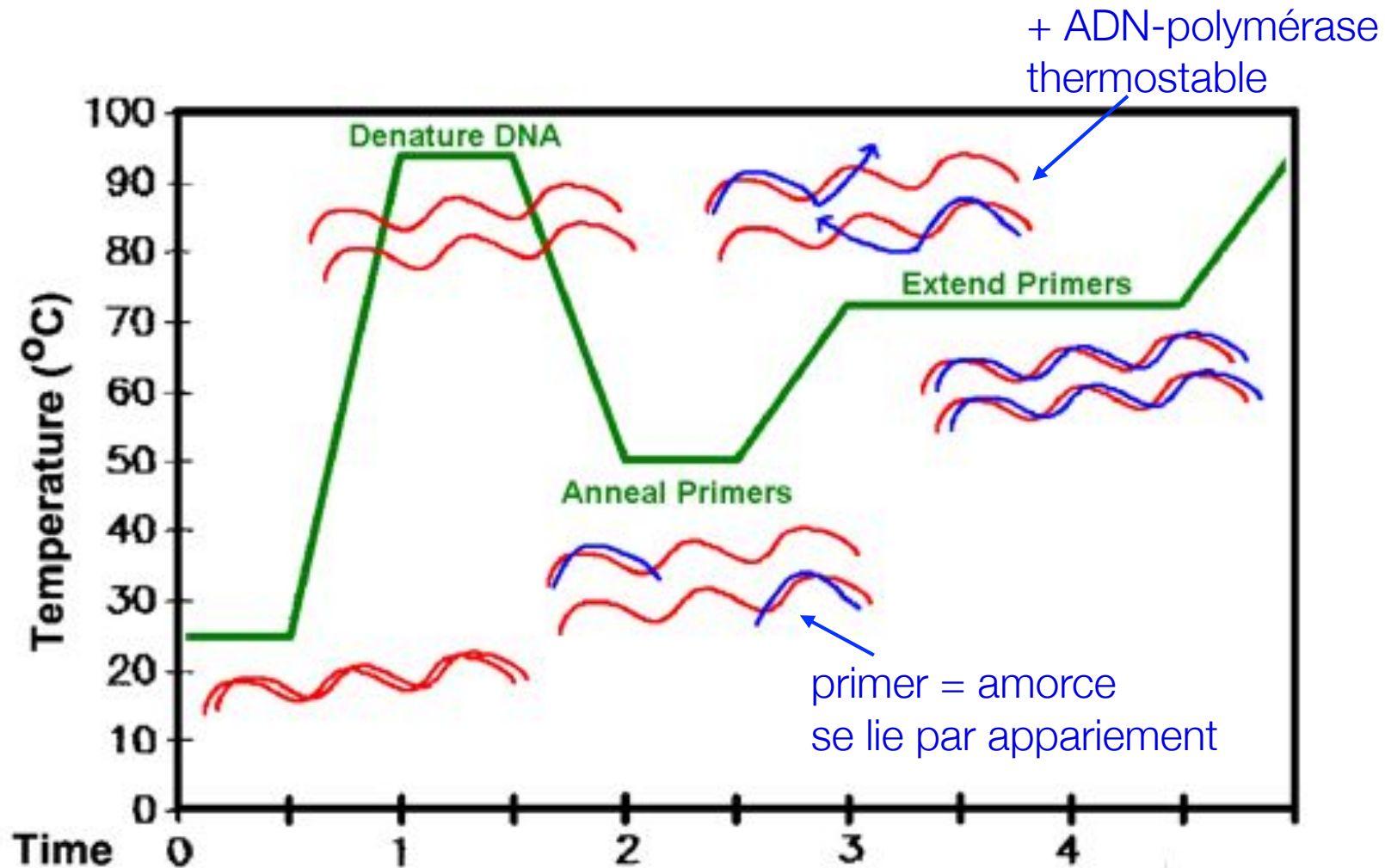
la température de dénaturation dépend
de la **concentration en GC** (triple liaison)

et la dénaturation se fait donc de manière inhomogène,
« par sauts », les régions riches en AT d'abord



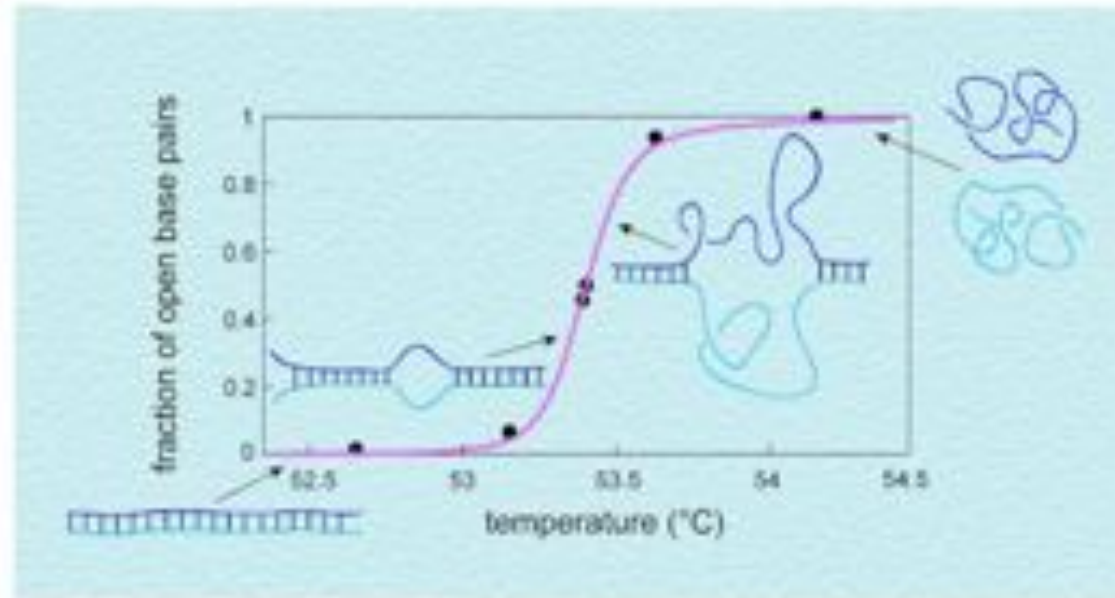
une technique incontournable : la PCR

polymerase chain reaction
(réaction en chaîne par polymérase)



30 - 40 cycles $\rightarrow 2^{30} = 1$ milliard de copies !

III - la dénaturation comme transition de phase

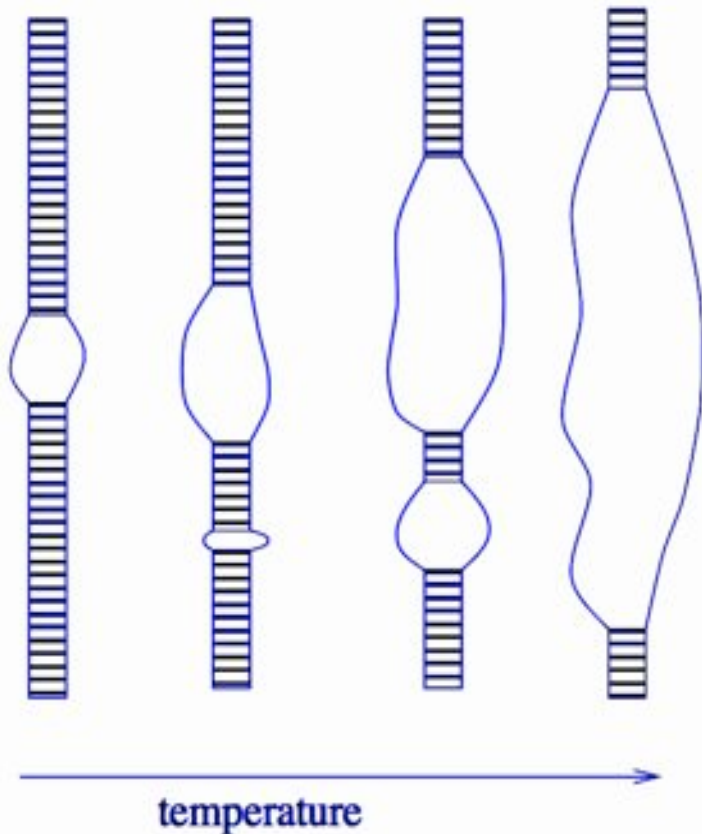


la dénaturation comme transition de phase

deux mécanismes :

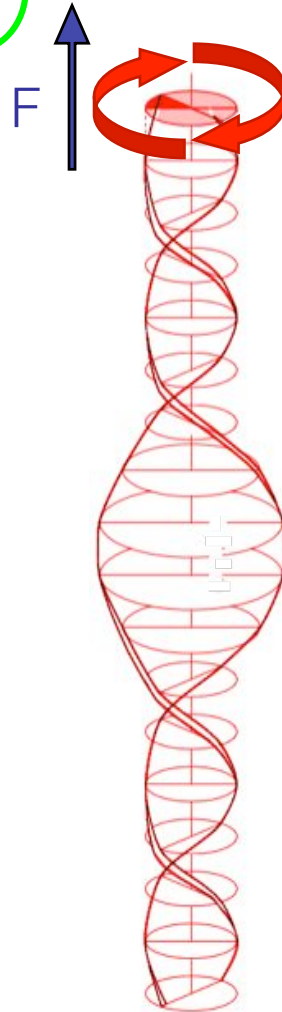
1. élévation de la température T

⇒ ouverture des paires de bases
⇔ déroulement hélice



2. application d'un couple Γ

⇒ ouverture des paires de bases
⇔ déroulement hélice



le résultat :

la fraction $\rho = \frac{n_d}{N}$

de paires de bases ouvertes augmente

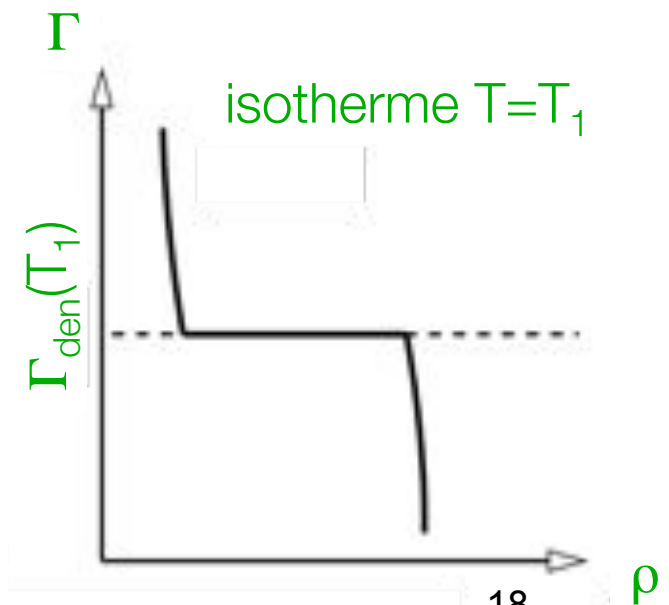
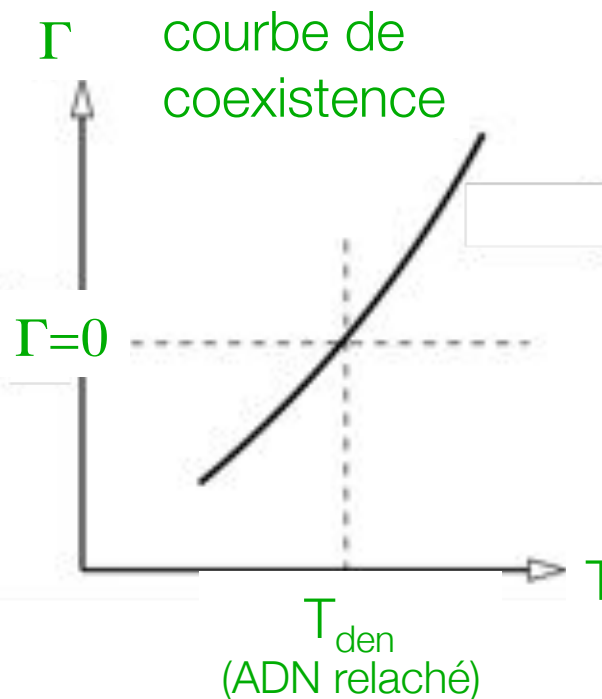
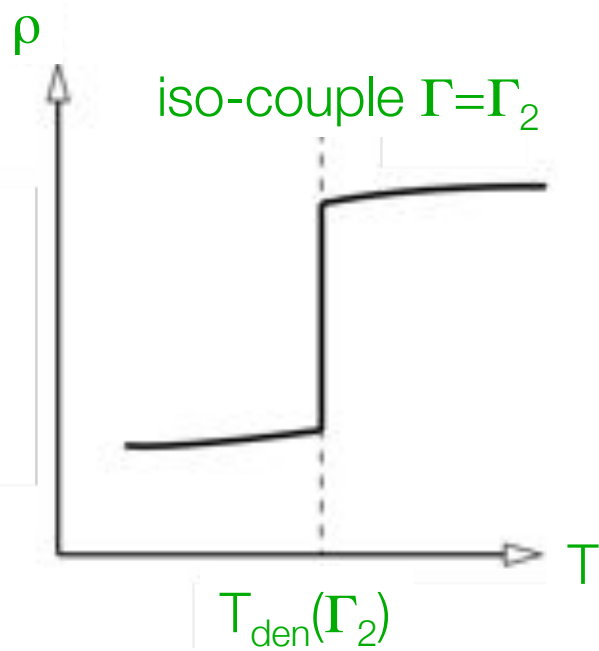
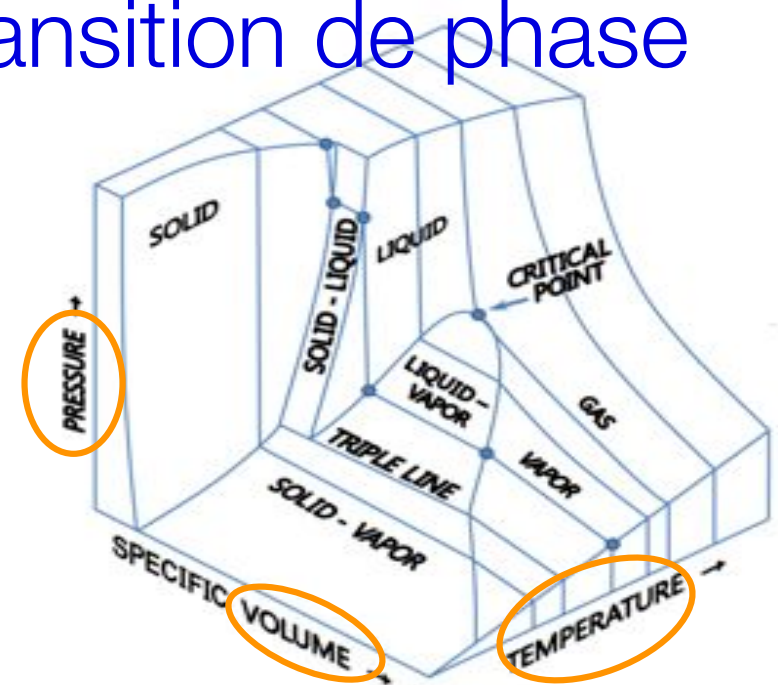
on remarque l'analogie avec un phénomène mieux connu..

la dénaturation comme transition de phase

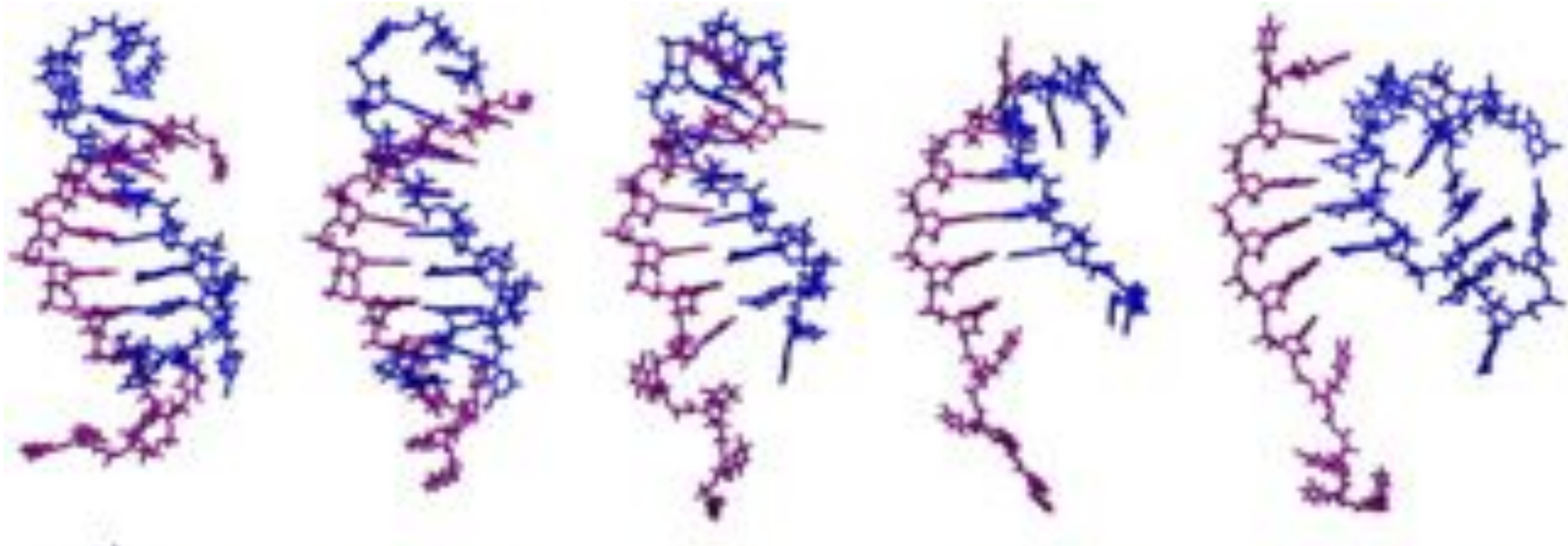
transition de phase liquide-gaz :

P	\leftrightarrow	Γ	couple
V	\leftrightarrow	ρ	fraction bps ouvertes
T	\leftrightarrow	T	température

dénaturation de l'ADN :



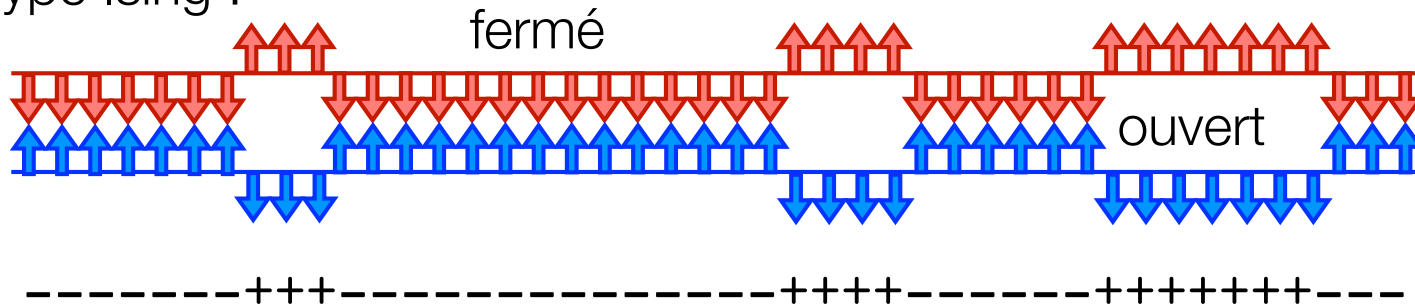
IV – modéliser la dénaturation de l'ADN



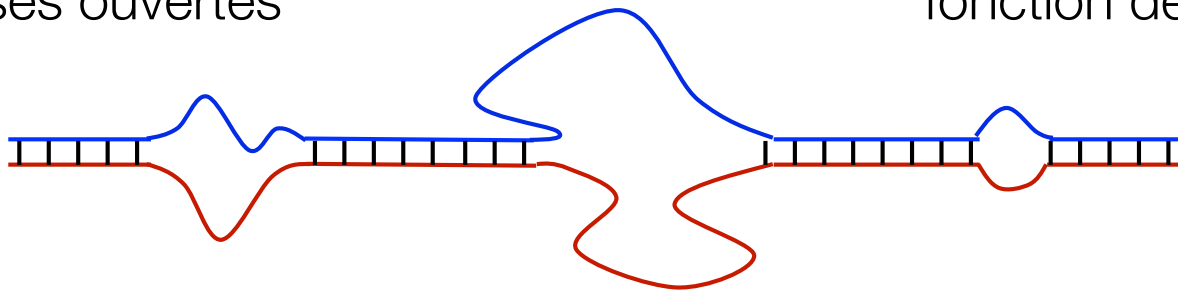
100-ns molecular dynamics simulations

a. les modèles statistiques

modèles type Ising :



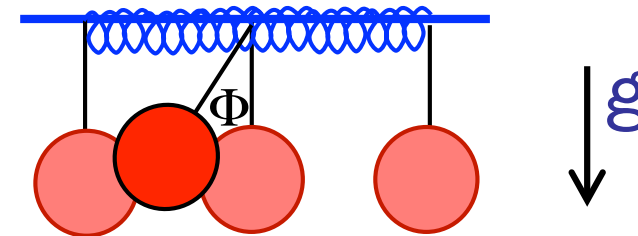
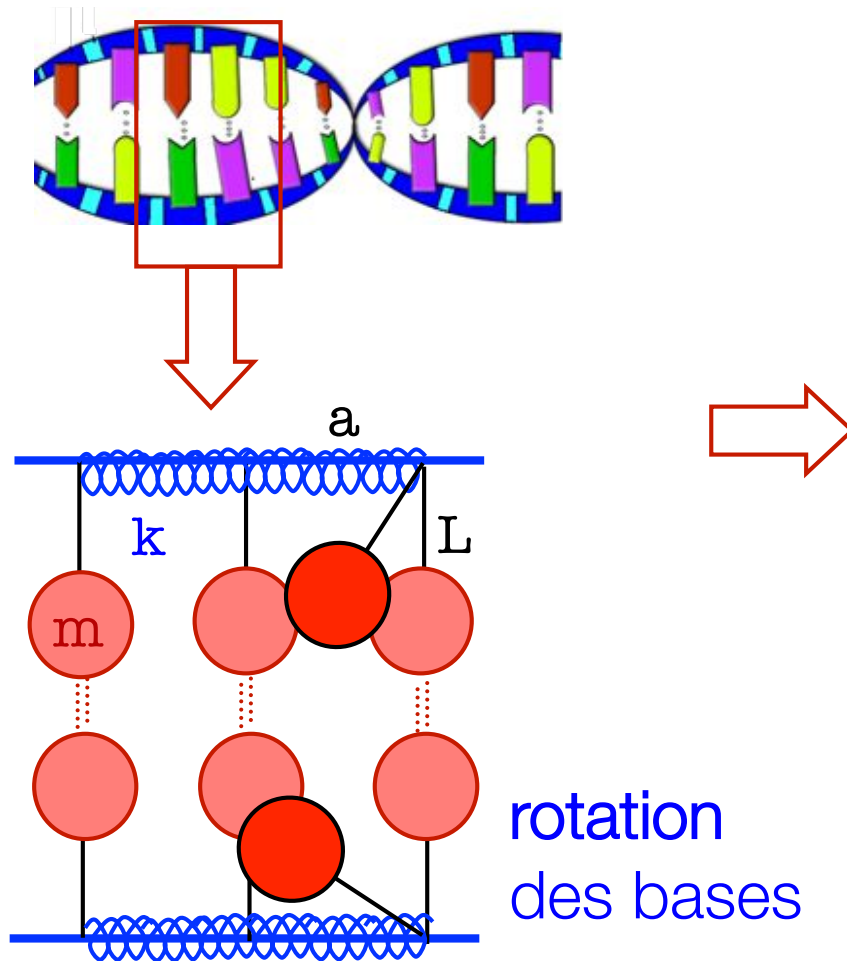
- pb ouverte = coût énergétique E : $Z(N) = \sum C_N^m e^{-mE / k_B T}$
si m bases ouvertes
fonction de partition



- boucle de taille i = gain entropique $S(i)$:
 - compétition entre énergie et désordre
 - modèles statistiques plus complexes

b. les modèles dynamiques

exemple 1 : des pendules couplés - modèle de Yakushevich

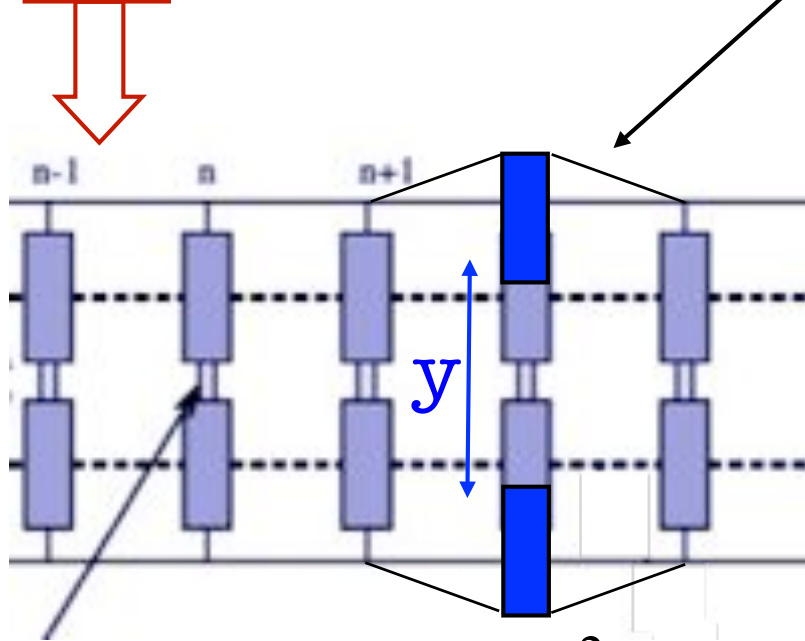
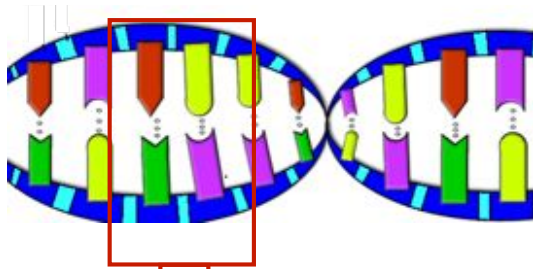


équation de sine Gordon

$$\frac{\partial^2 \Phi}{\partial x^2} - \frac{a^2 k}{I} \frac{\partial^2 \Phi}{\partial t^2} = \frac{mgL}{I} \sin(\Phi)$$

b. les modèles dynamiques

exemple 2 : des masses liées - **modèle de Peyrard-Bishop (TD)**



$$W(y_n, y_{n-1}) = K(y_n - y_{n-1})^2$$

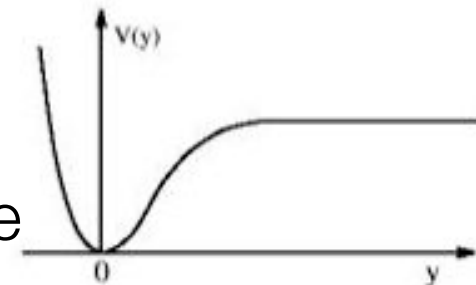
couplage harmonique...

$$W(y_n, y_{n-1}) = S(y_n - y_{n-1})^2 \left[1 + \rho e^{-b(y_n + y_{n-1})} \right]$$

...ou anharmonique

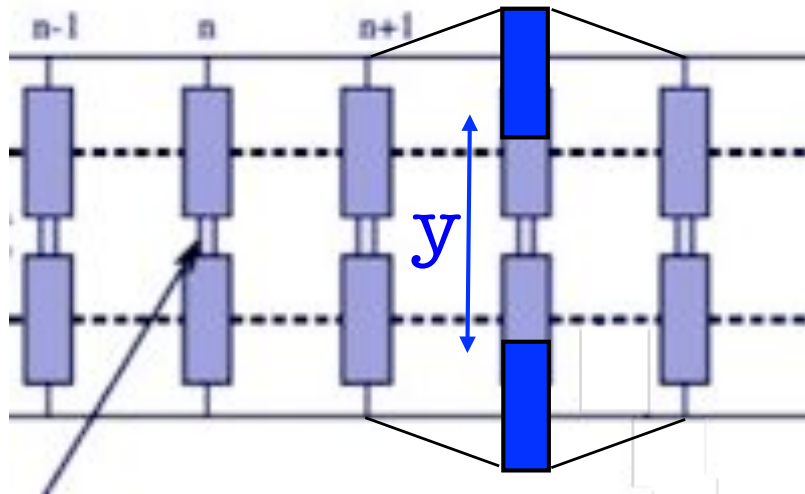
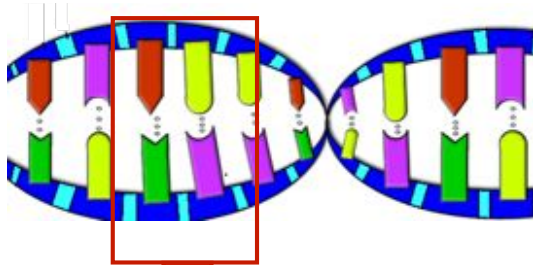
separation
des bases

$$V(y_n) = D(e^{-ay_n} - 1)^2 \text{ potentiel de Morse}$$

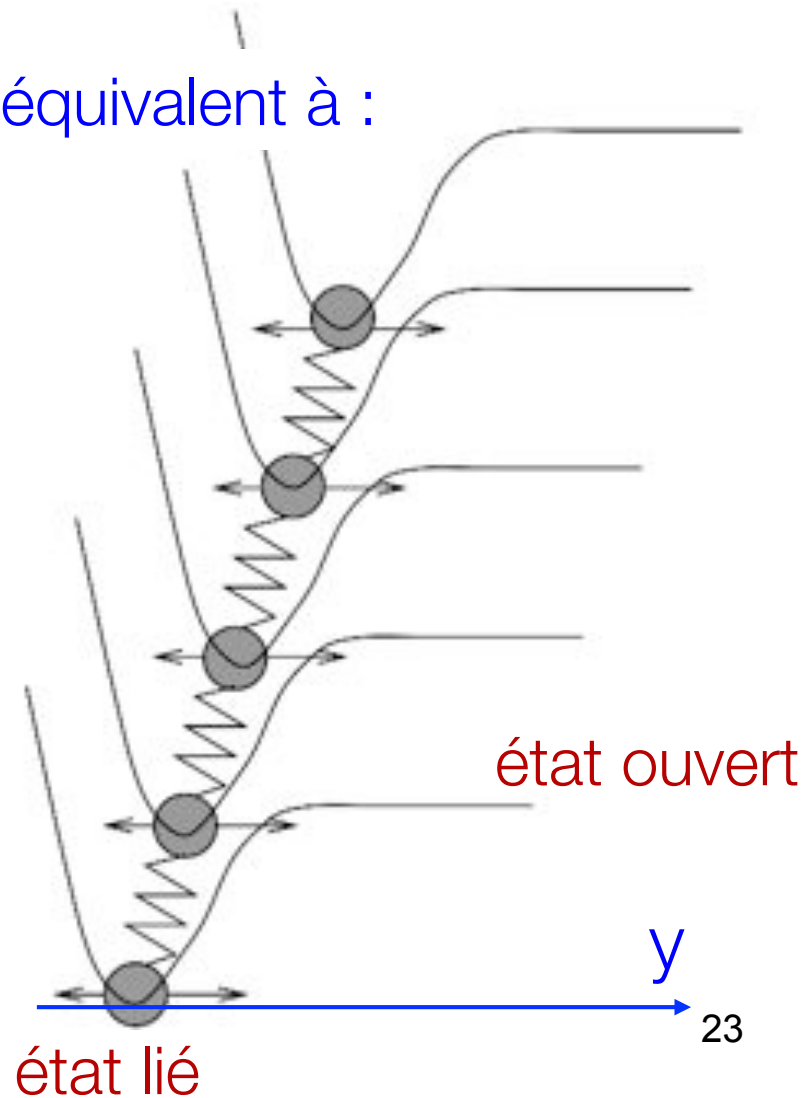


b. les modèles dynamiques

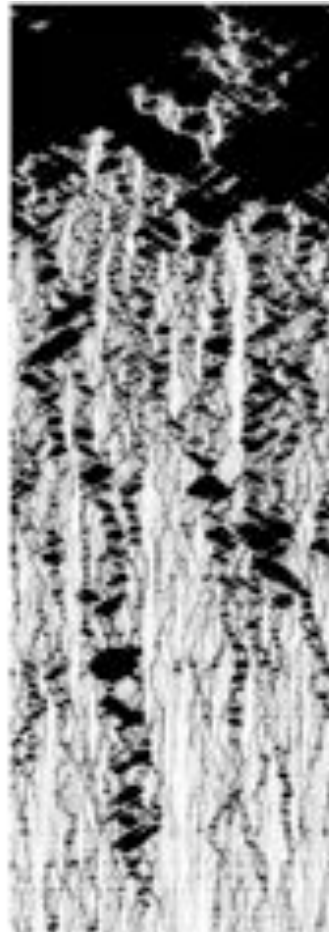
exemple 2 : des masses liées - modèle de Peyrard-Bishop (TD)



c'est équivalent à :



V – simulations de dynamique moléculaire



simulations de dynamique moléculaire

pour obtenir la dynamique du système :

il faut intégrer les équations du mouvement :

$$\begin{aligned} y &= (y_1, \dots, y_N) \text{ positions des masses} \\ v &= (v_1, \dots, v_N) \text{ vitesses associées} \end{aligned} \quad \begin{cases} y(t+\Delta t) = y(t) + v(t) \Delta t \\ v(t+\Delta t) = v(t) + (F(t)/m) \Delta t \end{cases}$$

(Euler)

- il faut donc déterminer $F_i = -\partial V_{\text{tot}} / \partial y_i$
- choisir les conditions initiales
- faire évoluer le système jusqu'à avoir de conditions stables
puis mesurer les quantités physiques d'intérêt : $T, P, E, \langle y \rangle \dots$

simulations de dynamique moléculaire

comment prendre en compte la température ?

solution 1 (simple) : simulation à énergie constante
(ensemble microcanonique)

on choisi comme conditions initiales (par exemple)

$\mathbf{y} = (0, \dots, 0)$ (position d'équilibre, $E_{pot} = 0$) \Rightarrow énergie pot. initiale = 0

\mathbf{v} tel que $E_{cin} = \sum \frac{1}{2} m v^2 = E_{totale} = 2 \frac{f}{2} N k_B T$
énergie (cinétique) initiale énergie prévue à température T

f = nombre de degrés de liberté par particule
= 1 pour le modèle Peyrard Bishop

c'est le théorème d'équipartition de l'énergie :

$$\langle e_{cin} \rangle = \langle e_{pot} \rangle = \frac{f}{2} k_B T \quad \text{pour } f \text{ degrés de liberté}$$

simulations de dynamique moléculaire

solution 2 : simulation à température constante T_0 (ens. canonique)

➔ plusieurs méthodes :

1. dynamique de Langevin :

$$v(t+\Delta t) = v(t) + (F(t)/m) \Delta t - \gamma v(t) \Delta t + R(t) \Delta t$$

viscosité du solvant

R force stochastique
simule l'effet des collisions

$$\langle R \rangle = 0$$

$$C_{RR}(\tau) = \langle R(t) R(t+\tau) \rangle$$

$$= 2 \gamma k_B T_0 m \delta(\tau)$$

c'est le théorème fluctuation dissipation : $D = \frac{k_B T}{\gamma}$

simulations de dynamique moléculaire


solution 2 : simulation à température constante T_0 (ens. canonique)

→ plusieurs méthodes :

2. thermostat de Berendsen :

$$v(t+\Delta t) = v(t) + (F(t)/m) \Delta t - \gamma v(t) \left(\frac{T_0}{T} - 1 \right) \Delta t$$

on rescale les vitesses pour
obtenir $\frac{dT}{dt} = 2\gamma(T - T_0)$



simulations de dynamique moléculaire

solution 2 : simulation à température constante T_0 (ens. canonique)

➔ plusieurs méthodes :

3. thermostat de Nosé-Hoover :

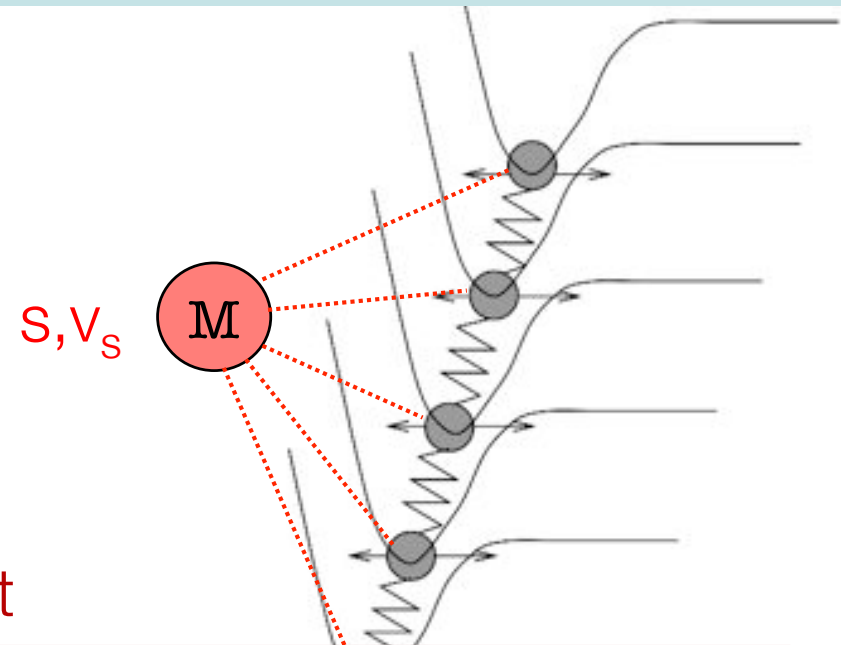
on ajoute un degré de liberté s
couplé à toutes les particules
qui agit de thermostat :

$$mv(t+\Delta t) = mv(t) + F(t)\Delta t - v_s mv(t)\Delta t$$

$$\ln s(t+\Delta t) = \ln s(t) + v_s \Delta t$$

$$Mv_s(t+\Delta t) = Mv_s(t) + [E_{\text{cin}} - N f k_B T/2] \Delta t$$

..positive ou négative selon $E_{\text{cin}} - E_{\text{cin}}^0$

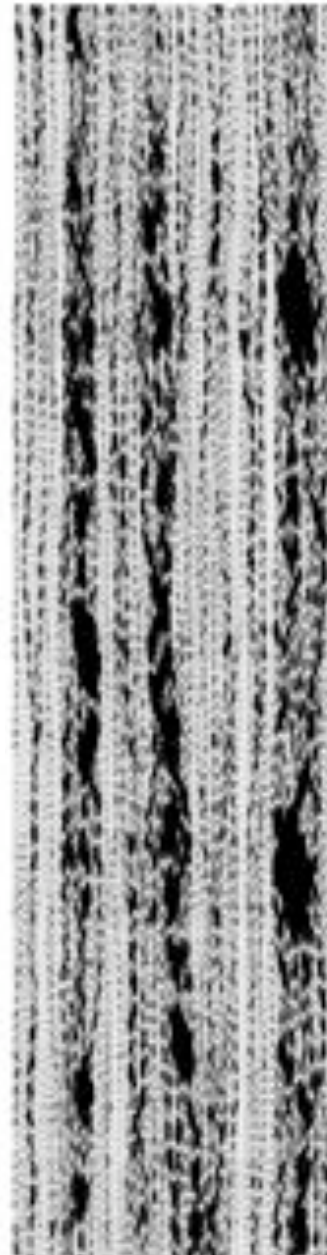


v_s agit comme une friction..

observons la dénaturation

simulations
à T fixée
(canonique)

temps \uparrow
 $T < T_D$

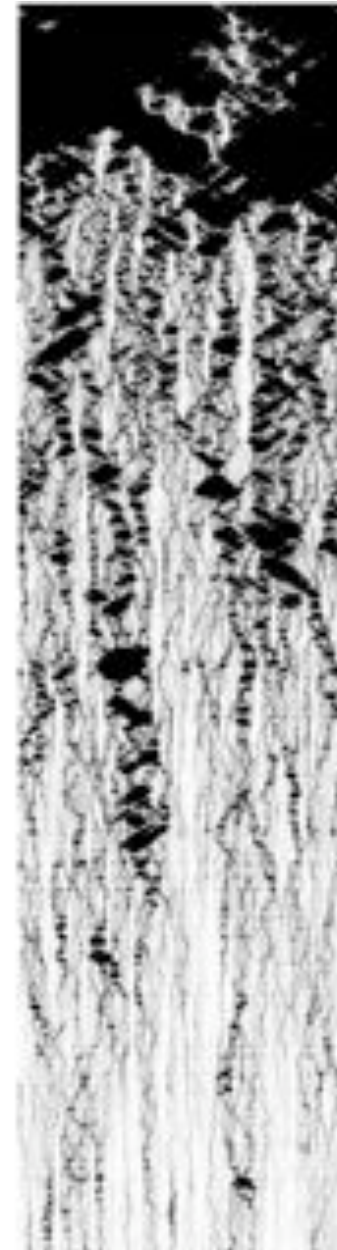


position 1...n...256

simulations
à T fixée
successives
avec $T \nearrow$

$T > T_D$

temps/température \uparrow



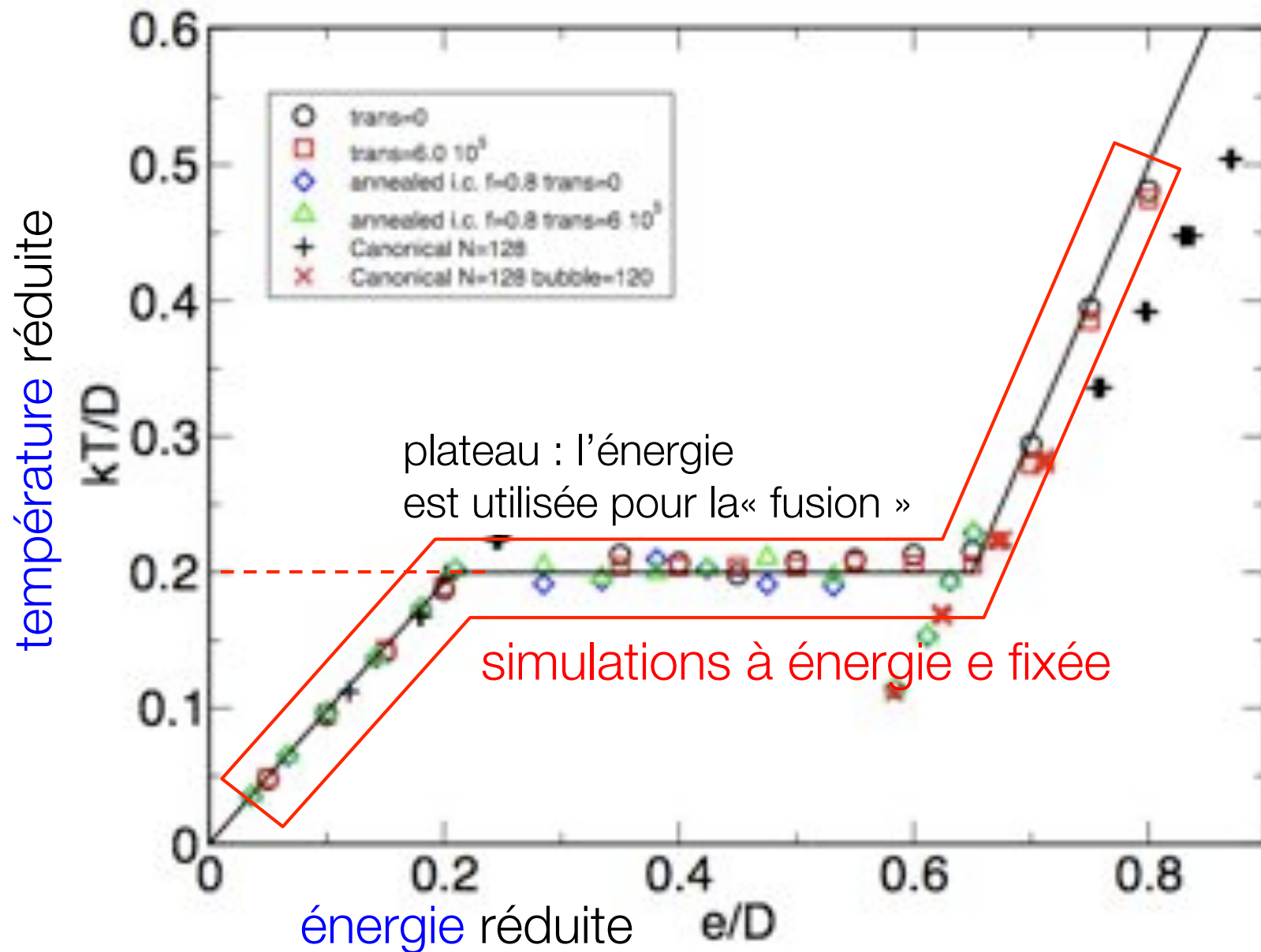
$T < T_D$

position 1...n...256

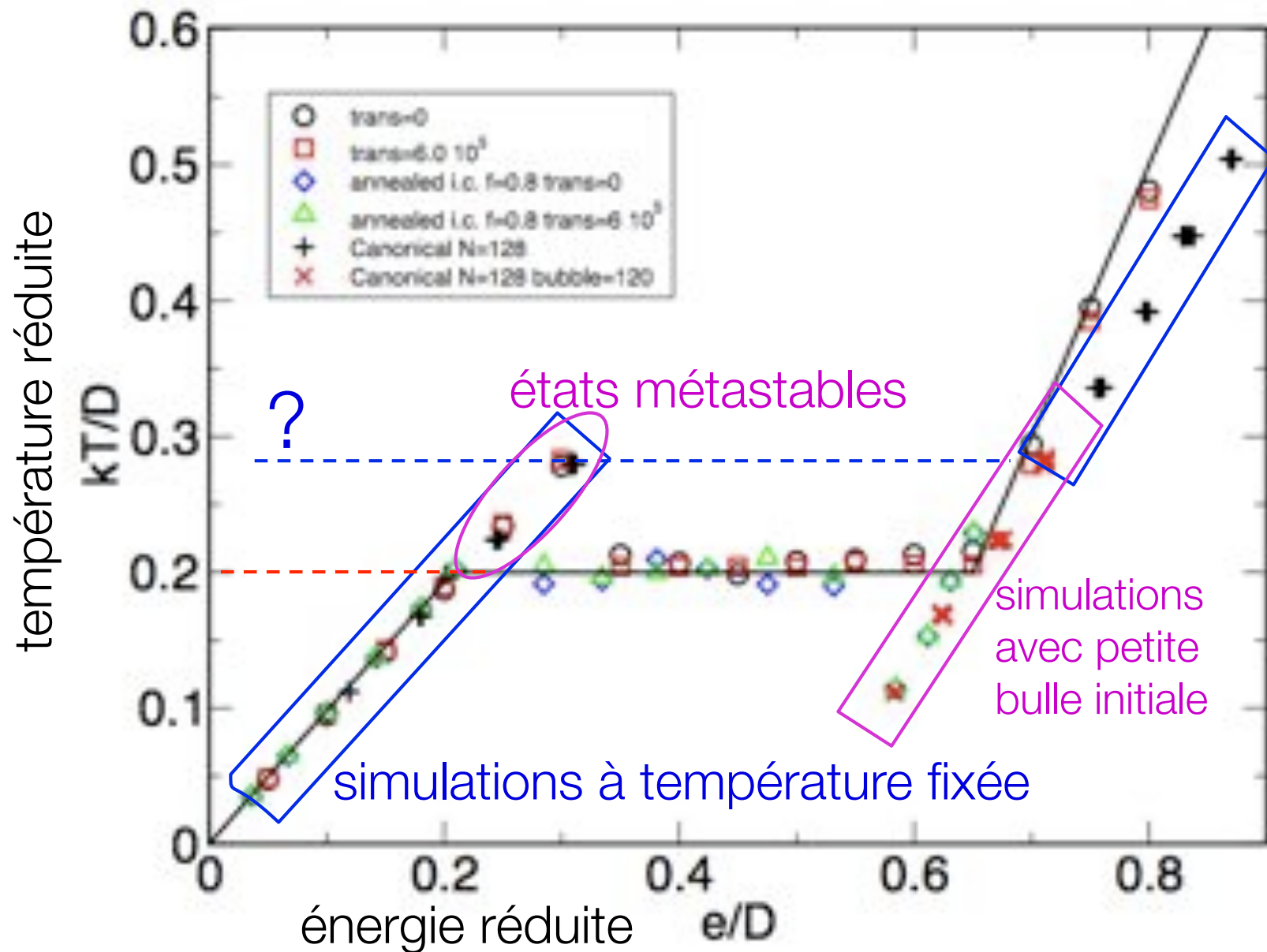
blanc=fermé

noir=ouvert

simulations sur un modèle proche de PB

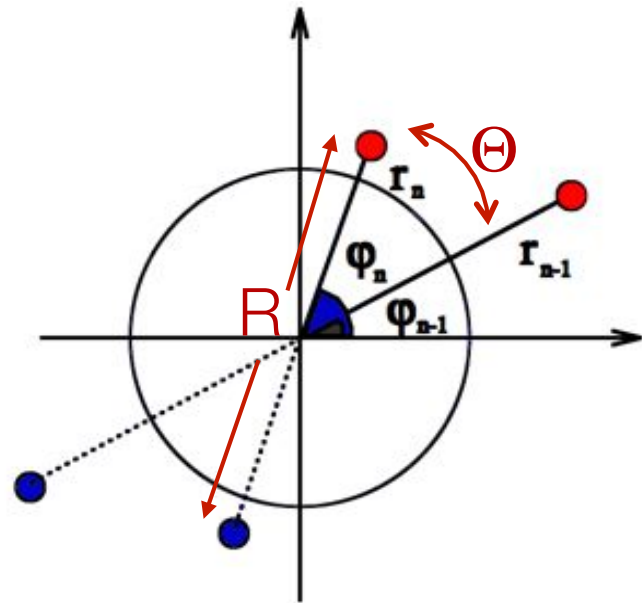


simulations sur un modèle proche de PB

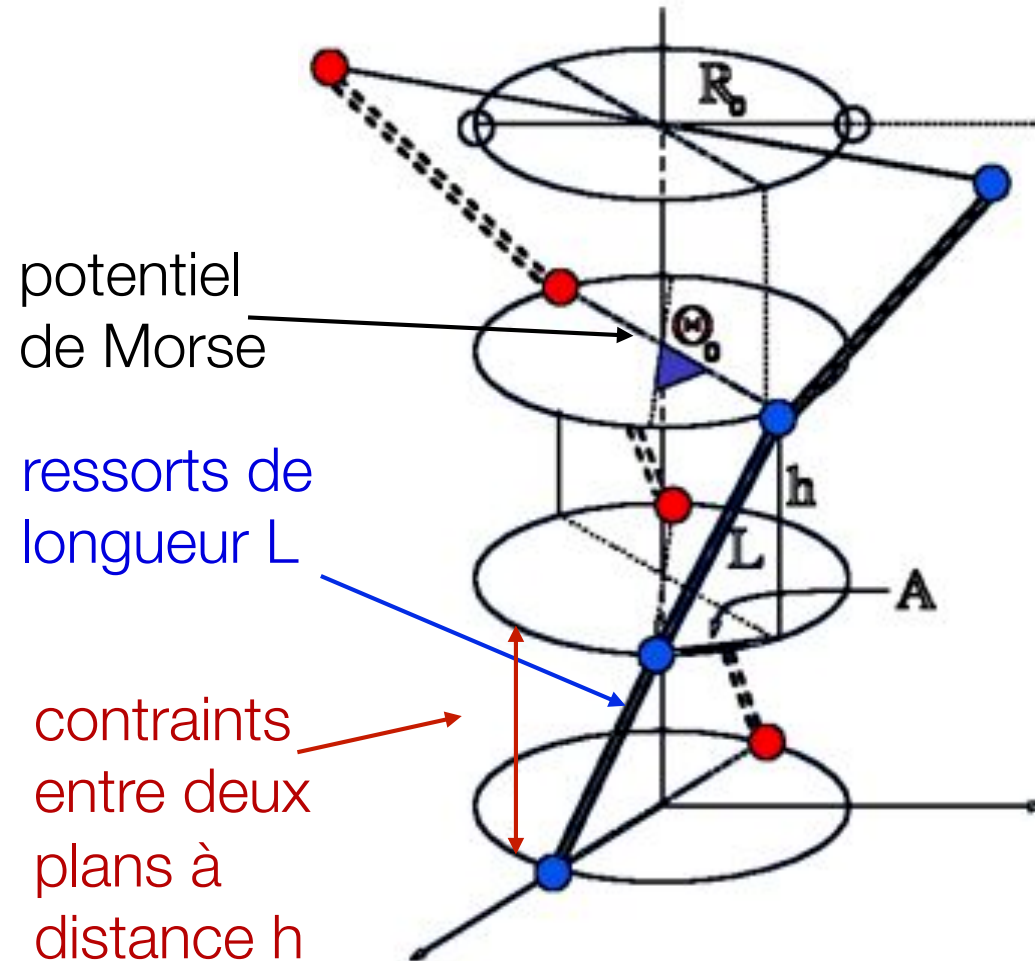


bonus : le modèle PB version hélicoïdale

les résultats précédents ont été obtenus avec

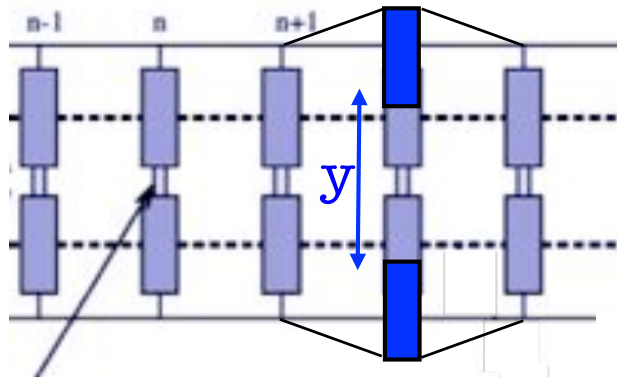


2 degrés de liberté :
séparation R , rotation Θ



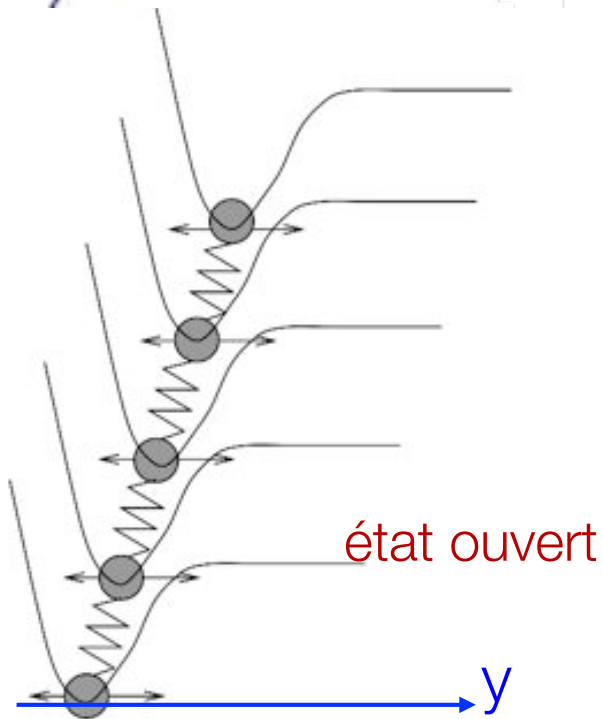
Le TD

modèle de Peyrard-Bishop (TD)



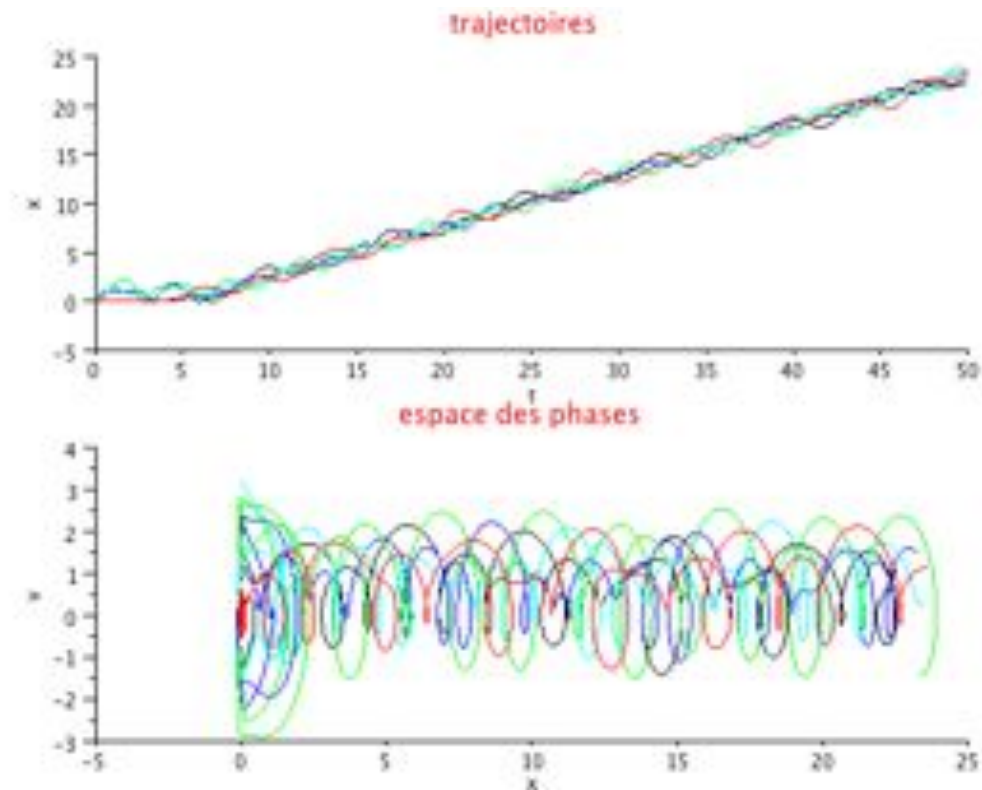
$$V(y_n) = D(e^{-ay_n} - 1)^2$$

$$W(y_n, y_{n-1}) = K(y_n - y_{n-1})^2$$



état lié

état ouvert



Le TD : fonction python « *ode* »

```
def f (t, y, args) :  
    x0 = y[0]  
    x1 = y[1]  
    k = args[0]  
    m = args[1]  
    return [x1, -(k/m)*x0]
```

← définition d'une fonction *force généralisée* $[v(t), f(t)/m]$
→ $[v_1(0), \dots, v_N(0), f_1(0)/m, \dots, f_N(0)/m]$

```
t0 = 0  
y0 = [1., 0.]  
r = ode(f).set_integrator('vode')  
r.set_initial_value(y0, t0).set_f_params([3.0, 1.0])  
t1 = 10.  
dt = 0.02  
sol = [y0]  
while r.successful () and r.t < t1 :  
    r.integrate (r.t+dt)  
    sol.append (r.y)
```

← conditions initiales $[x(0), v(0)]$ → $[x_1(0) \dots x_N(0), v_1(0) \dots v_N(0)]$

← temps final et pas de temps

← solution cherchée : premier élément $[x, v]$

← boucle d'intégration

← solution cherchée : $[x, v]$ suivants en append

appel à ode : « r »
contient la procédure