

Coarse grain models (continuous) ... potentials of mean force

So far ?

- very detailed models
 - atomistic, solvation

What are some reasonable aims ?

- given a set of coordinates
 - are these roughly correct for a protein sequence ?
 - is this more likely to be α -helical or β -sheet ?

Should we approach this with a detailed force field ?

- maybe not-

Aims

- Why atomistic force fields / score functions are not always best
- Different levels of force fields
- Examples of coarse-grain / low-resolution force fields
- Ways to parameterise force fields
- Score functions directly from structural data

- later...
- extending this idea to lattice models

History

History

- Levitt, M and Warshel, A, Nature, 253, 694-698, Computer simulation of protein folding (1975)
- Kuntz, ID, Crippen, GM, Kollman, PA and Kimelman, D, J. Mol. Biol, 106, 983-994, Calculation of protein tertiary structure (1976)
- Levitt, M, J. Mol. Biol, 104, 59-107, A simplified representation of protein conformations for rapid simulation of protein folding (1976)
- through to today

Problems with detailed force fields

Time

- typical atomistic protein simulations 10^{-9} to 10^{-6} s
- too short for folding

Radius of convergence

- I have coordinates where atoms are perturbed by 1 Å
 - easy to fix – atoms move quickly
- I have completely misfolded, but well packed coordinates
 - may be difficult to fix
 - what dominates ?
 - atomic packing
 - charges
 - solvation ?

Do I care about details ?

Coarse grain / low resolution

Forget atomic details

- build something like energy which encapsulates our ideas
- example – define a function which is happiest with
 - hydrophobic residues together
 - charged residues on outside
- would this be enough ?
 - maybe / not for everything

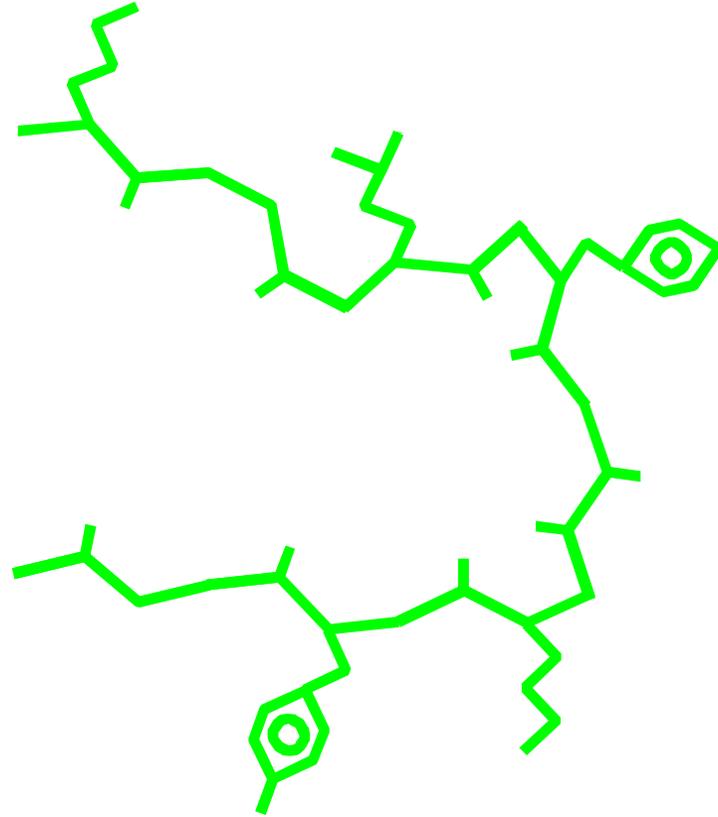
What will I need ?

- some residues like to be near each other (hydrophobic)
- residues are always some constant distance from each other
- only certain backbone angles are allowed

General implementation (easiest)

How do we represent a protein ?

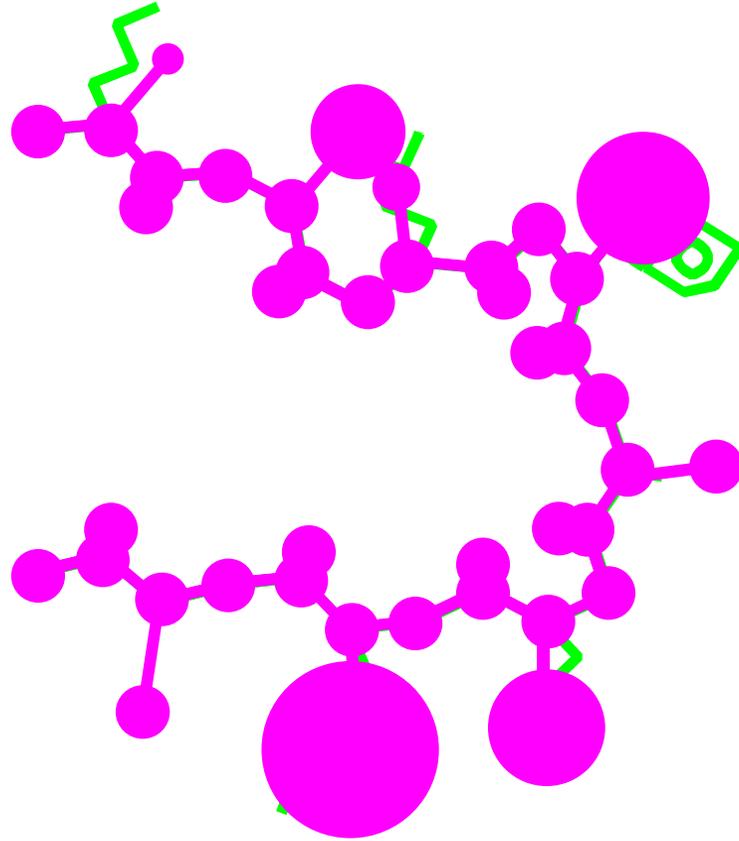
- decide on number of sites per residue



General implementation (easiest)

How do we represent a protein ?

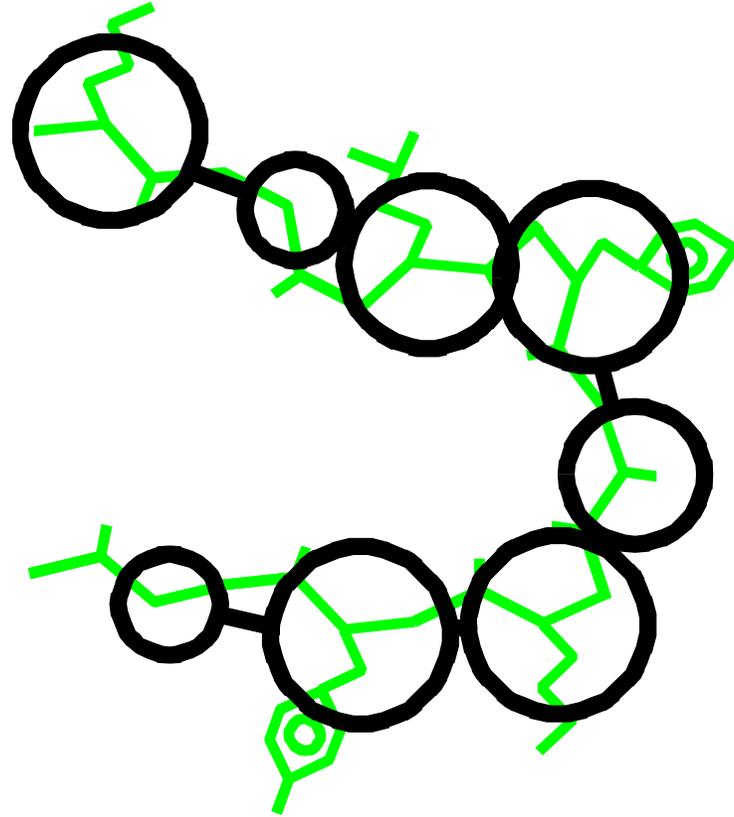
- decide on number of sites per residue



General implementation (easiest)

How do we represent a protein ?

- decide on number of sites per residue



Coarse-graining (steps)

- Decide on representation
- Invent quasi-energy functions

Our plan

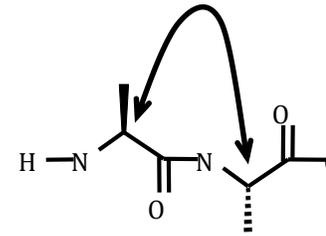
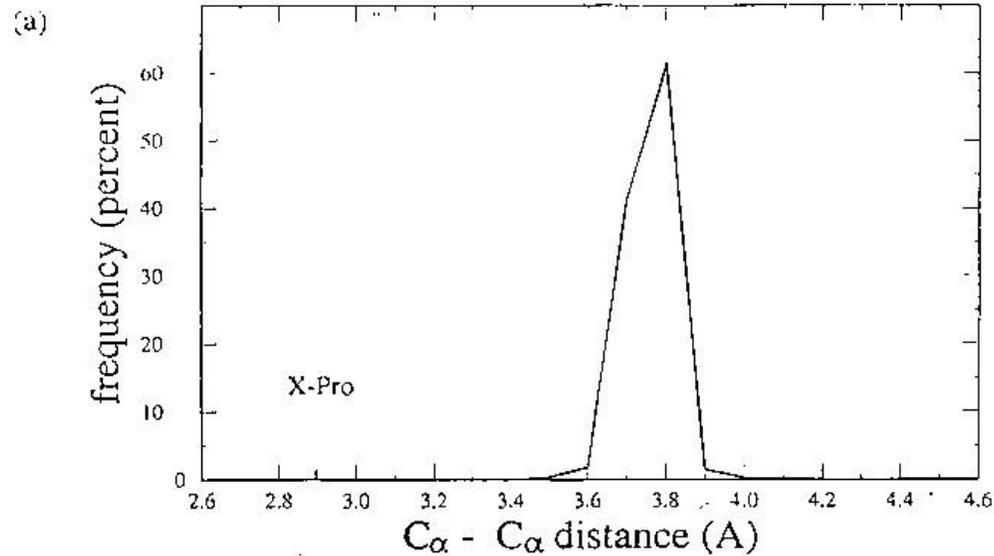
- step through some examples from literature

Common features

- some way to maintain basic geometry
- size
- hydrophobicity ? Which residues interact with each other/solvent

Basic geometry

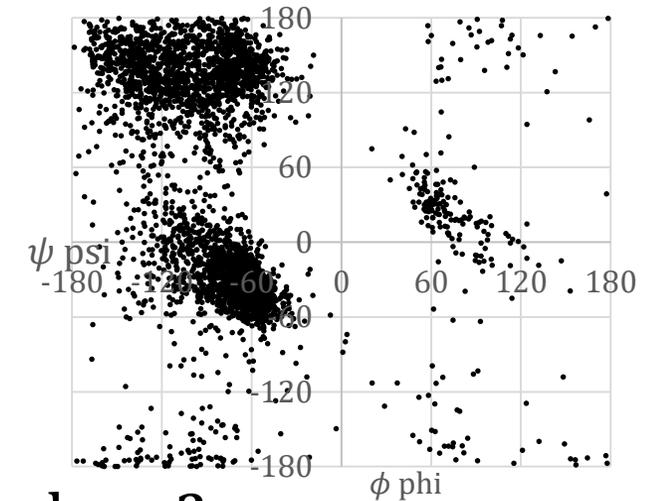
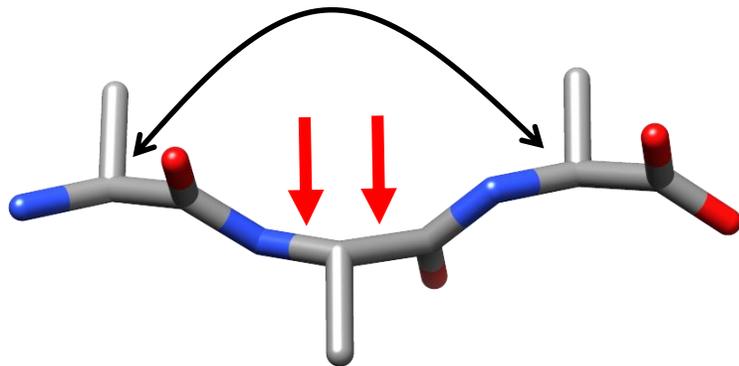
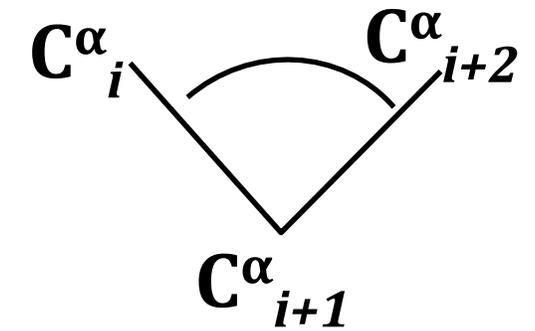
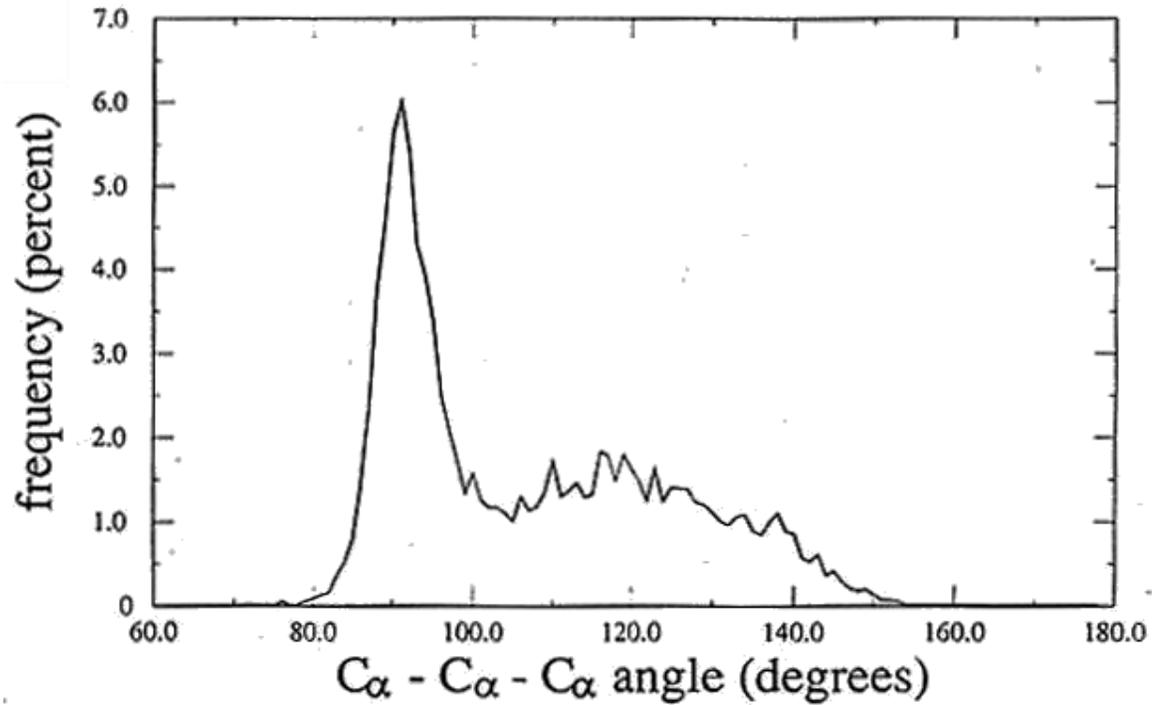
Survey protein data bank files and look at $C\alpha$ to $C\alpha$ distances



Conclusion is easy

- any model should fix $C_{i,i+1}^{\alpha}$ distances at 3.8 Å
- what other properties do we know ?

$C_{i,i+2}^\alpha$ distance / angle



- why is distance less clear ?
- think of ramachandran plot

First simple model

n residues, n interaction sites $i, i + 1$ restrained (C^β formulation)

Overlap penalty / radii

- lys 4.3 Å, gly 2.0 Å, ... trp 5.0 Å
- $U(r_{ij}) = (\text{radius}_i + \text{radius}_j)^2 - r_{ij}^2$

force hydrophilic residues to surface, for these residues

- $U^*(r_{ij}) = (100 - d_i^2)$ where
 d_i is distance to centre, 100 is arbitrary

disulfide bonds

- very strong

residue specific interactions

- $U^{long}(r_{ij}) = c_{ij}(r_{ij}^2 - R^2)$ where c_{ij} is residue specific
- R is 10 Å for attraction, 15 Å for repulsion

residue specific part of interaction

- c_{ij} table
- features
 - hydrophobic
 - + -
 - nothing much

	lys	glu	...	gly	pro	val
lys	25	-10		0	0	10
glu	-10	25		0	0	10
...						
gly	0	0		0	0	0
pro	0	0		0	0	0
val	10	10		0	0	-8

summary

- $i, i+1$ residue-residue
- overlap
- long range
- solvation

where is physics ?

- solvation ?
 - term pushes some residues away from centre
- electrostatics
- hydrophobic attraction
 - by pair specific c_{ij} terms

other properties

- smooth / continuous function
- derivative with respect to coordinates
 - (good for minimisation)

does it work ? what can one do ?

results from first model

- try to "optimise" protein structure
- for 50 residues, maybe about 5 Å rms
 - maybe not important

Model does..

- make a hydrophobic core
- put charged and polar residues at surface
- differentiate between possible and impossible structures

Model does not reproduce

- any geometry to Å accuracy
- details of secondary structure types (not intended)
- physical pathways
- subtleties of sequence features (simplicity of c_{ij} matrix)

Improvements to simple model

Aim

- biggest improvement for least complication

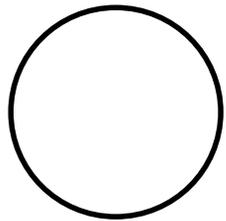
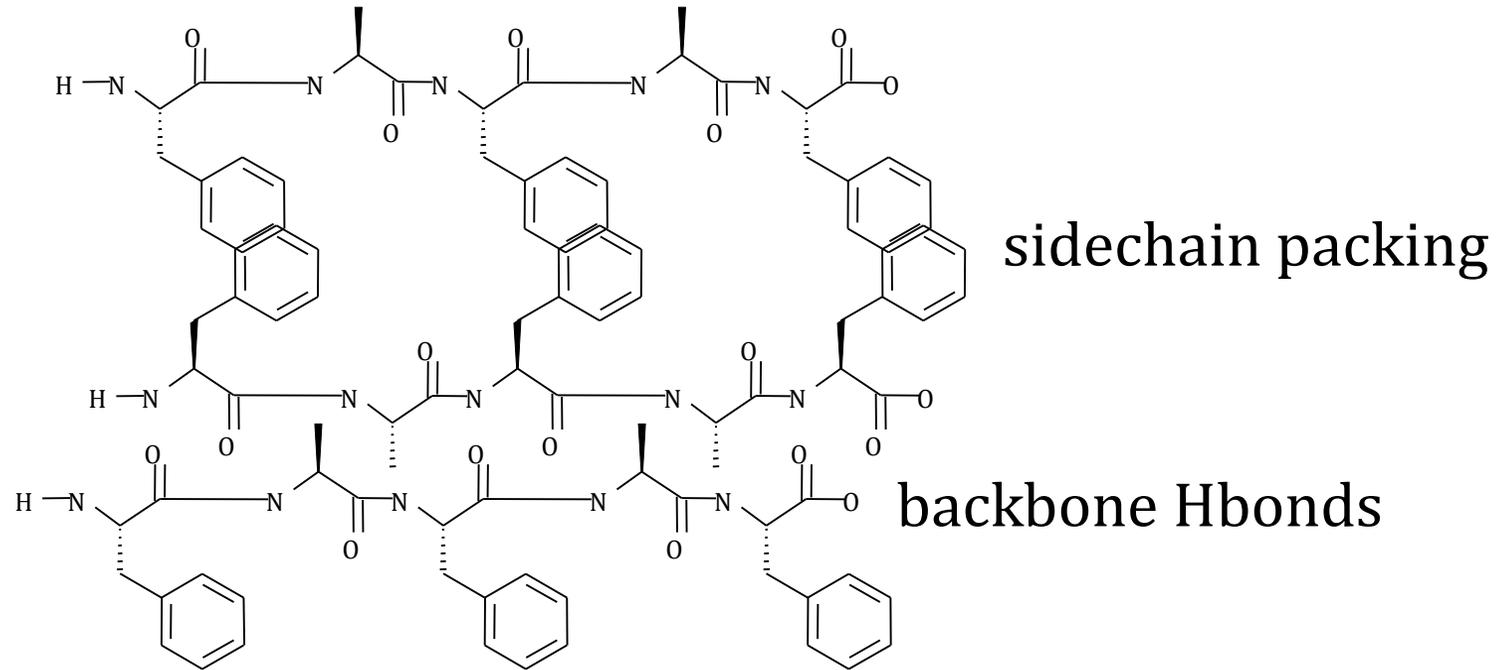
Possibilities

- more points per residue
- more complicated c_{ij} matrix... (more types of interactions)
- an example weakness

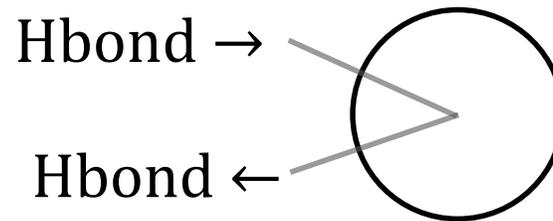
Important structural features of proteins

- all proteins have hydrogen bonds at backbone
- proteins differ in their sidechain interactions..

more complicated interactions



one point residue

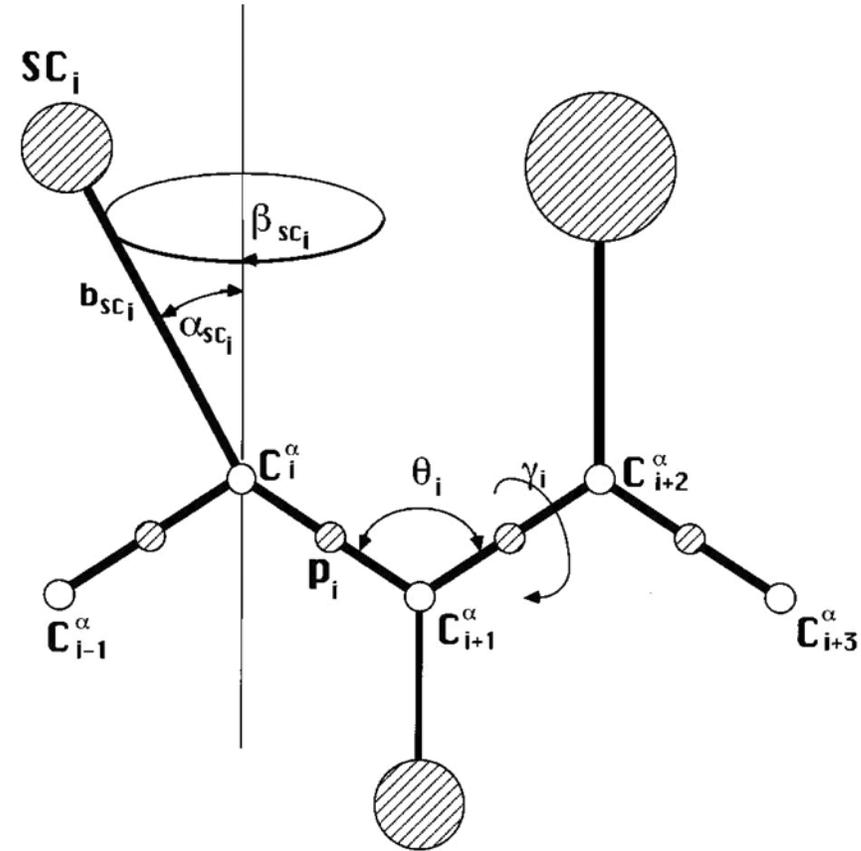


3 points per residue

Scheraga model

3 points per residue

- 2 for interactions
 - p_i is peptide bond centre
 - SC_i is sidechain
- 1 for geometry
 - C^α
- $C^\alpha - C^\alpha$ fixed at 3.8 \AA

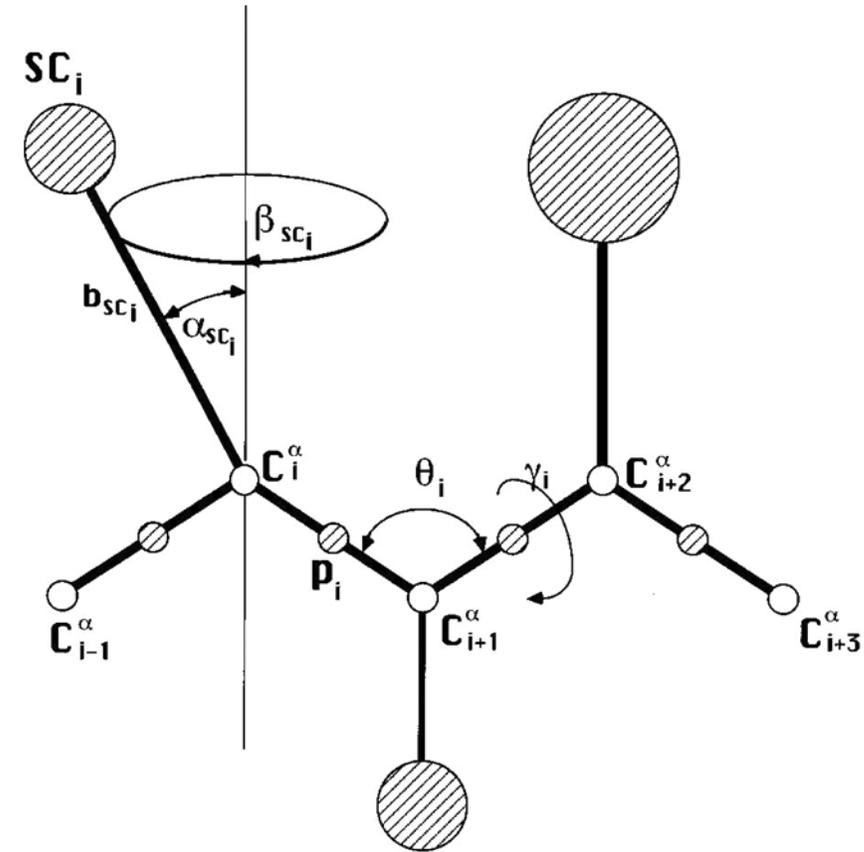


Do interaction sites correspond to atoms ?

Terms in Scheraga model

Total quasi energy =

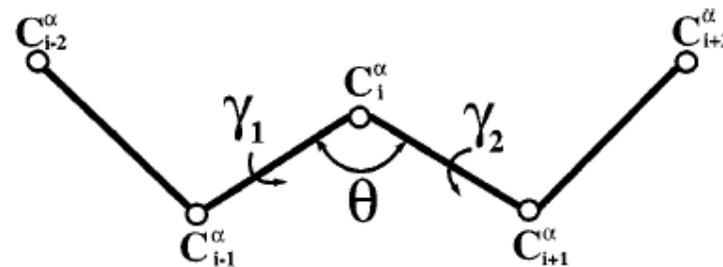
- side-chain to side-chain
- side-chain to peptide
- peptide to peptide
- torsion angle γ
- bending of θ
- ...
 - bending α_{sc}



angle between C^α sites

Cunning approach

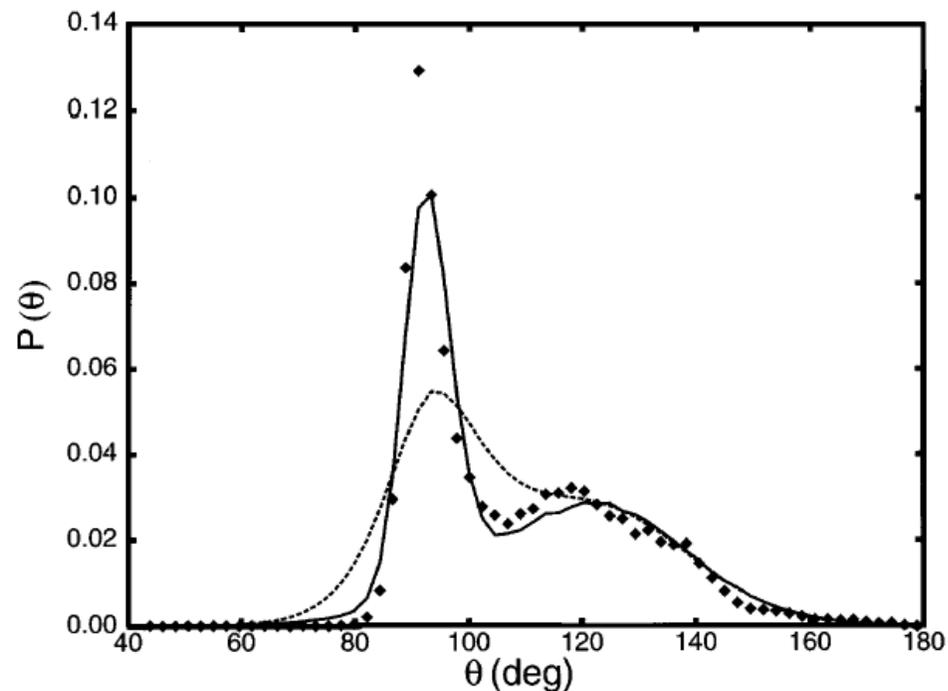
- look at θ distribution
- model with Gaussians



then say

$$U(\theta)^{bend} = -RT \ln P(\theta)$$

where $P(x)$ is the probability of finding a certain x



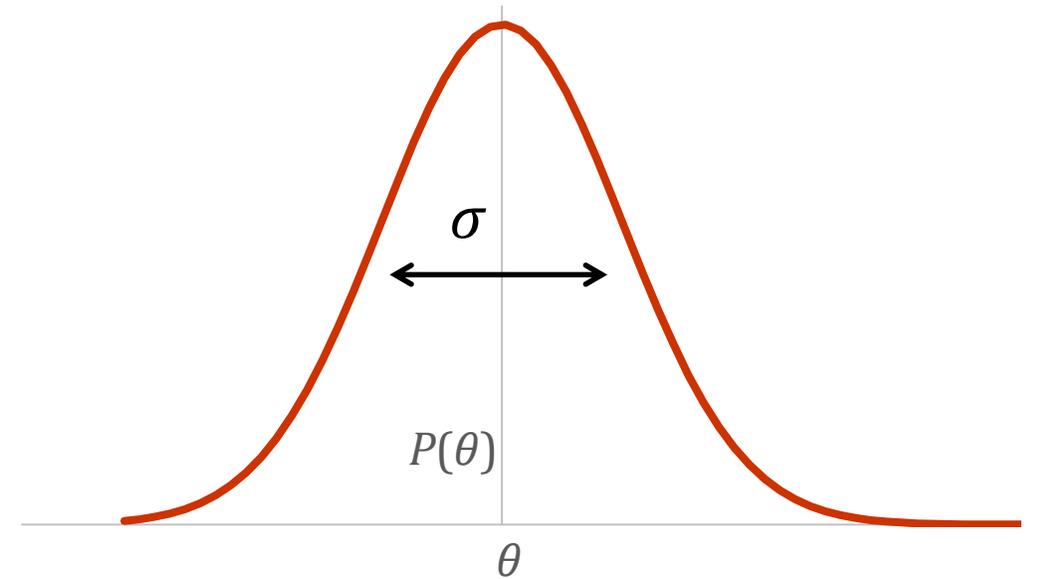
Gaussian reminder

- get μ and σ from fitting
- angle θ depends on fitting

$$P(\theta) = \frac{1}{\sigma\sqrt{2\pi}} \exp\left(\frac{-(\theta - \mu)^2}{2\sigma^2}\right)$$

- how would forces work ?
- express θ in terms of coordinates r
- say $U(\theta)^{bend} = -RT \ln P(\theta)$

- take $\frac{dU}{d\theta} \frac{\partial \theta}{\partial \vec{r}}$



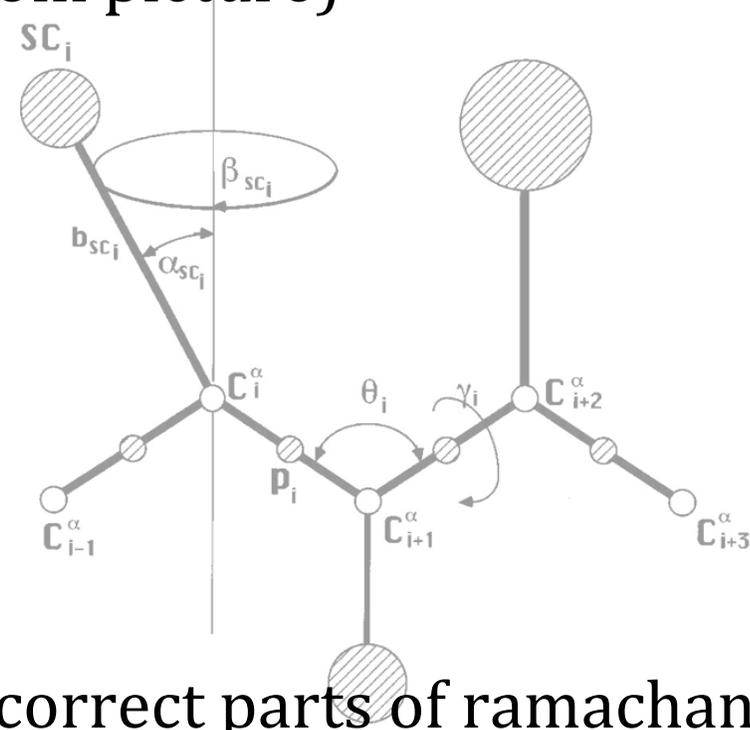
pseudo torsion term

Like atomic torsion $U(\gamma_i) = a_i \cos n\gamma_i + 1 + b_i \sin n\gamma_i + 1$

- n varies from 3 to 6 depending on types $i + 1, i + 2$
(numbering from picture)

Three kinds of pair

- gly
- pro
- others



Net result ?

- residues will be positioned so as to populate correct parts of ramachandran plot
- this model will reproduce α -helix and β -sheets

side-chain peptide

Not so important

- mostly repulsive $U^{sc-p}(r_{sc-p}) = kr_{sc-p}^{-6}$
- k is positive, so energy goes up as particles approach

side chain interactions

Familiar
$$U(r_{ij}) = 4\varepsilon_{ij} \left(\left(\frac{\sigma_{ij}}{r_{ij}} \right)^{-12} - \left(\frac{\sigma_{ij}}{r_{ij}} \right)^{-6} \right)$$

- but, consider all the σ and ε
- main result
 - some side chains like each other (big ε)
 - some pairs can be entirely repulsive (small ε big σ)
 - some not important (small ε small σ)

more complications

Real work used

- different forms for long range interactions
- cross terms in pseudo angles

nicht für Klausur

What can one do ?

Typical application

Background

- protein comparison lectures..
- different sequences have similar structure
 - can we test some structure for a sequence

Remember sequence + structure testing in modelling Übung ?

- here
 - given some possible structures for a sequence
 - can be tested with this simple force field

What can we not do ?

- physical simulations
 - think of energy barriers (not real)
 - time scale

summary of philosophy

- Is any model better than others ?
- Each model represents something of interest
 - hydrophobic / hydrophilic separation
 - reasonably good quality structure with
 - real secondary structure
 - accurate geometry

Main aims

- pick the simplest model which reproduces quantity of interest

Are there bad models ?

- complicated, but not effective
- interaction sites at wrong places
 - not efficient
 - not effective

Parameterisation..

Problem example

- charge of an atom ?
 - can be guessed, measured ? - calculated from QM
- ϵ and σ in atomistic systems
 - can be taken from experiment (maybe)
 - adjust to reproduce something like density

What if a particle is a whole amino acid or sidechain ?

- is there such a thing as
- charge ?
- ϵ and σ ?

Approaches to parameterisation

General methods

- average over more detailed force field (brief)
- optimise / adjust for properties (brief)
- potentials of mean force / knowledge based (detailed)

From detailed to coarse grain

Assume detailed model is best

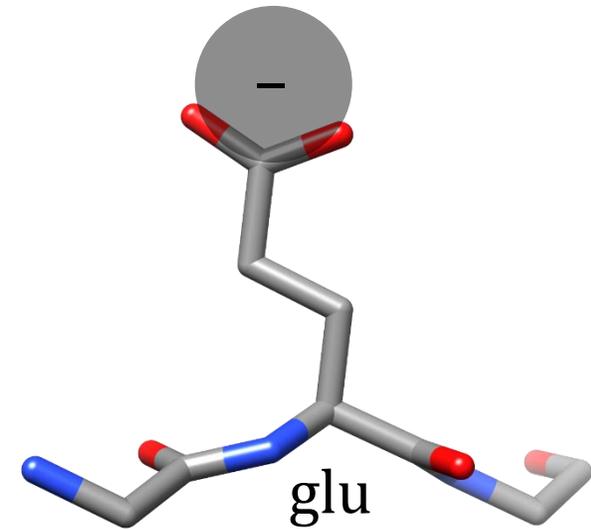
- Can we derive coarse grain properties from detailed ?

Examples – consider one or two sites per residue

- mass ? easy – add up the mass of atoms (also boring)

Charge ? not easy

- size of charge - obvious
- location ?
 - not easy
 - does this let us include polarity ? No.
- is this the right way to think about it ?...



Averaging over details is not easy

General interaction between two residues

- will depend on orientation, distance, other neighbours
- not all orientations occur equally likely
- sensible averaging not obvious
- better approach ...

Parameterising by adjustment / optimisation

```
for (parameter = small; parameter < big ; parameter++)  
    measure happiness
```

Define happiness - what do you want ?

- density at equilibrium
- free energy change of some process
- distance of average protein structure from X-ray
-

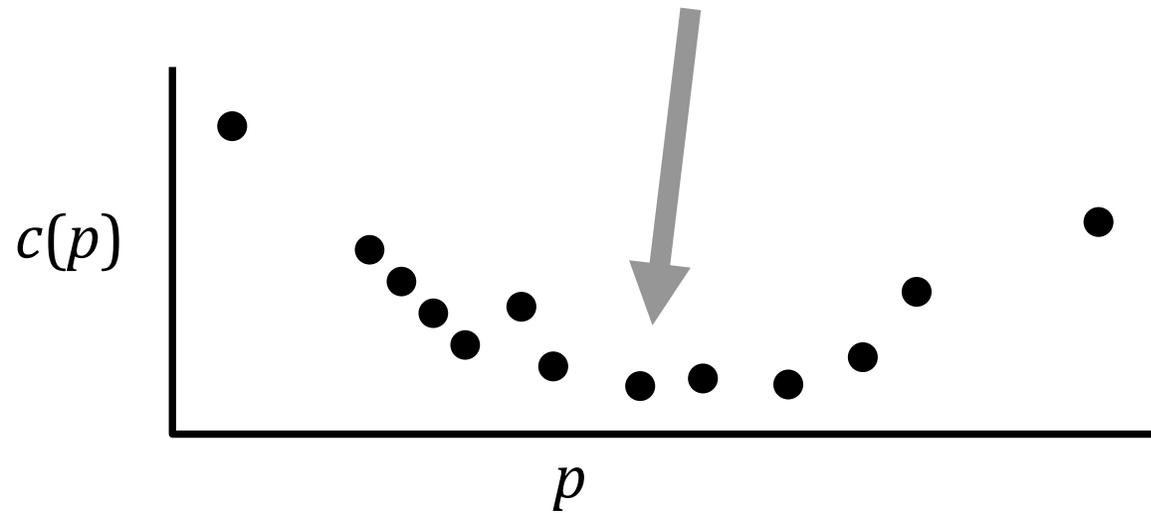
cost function

For your definition of happiness

- some measured observable \mathcal{A}_{obs}
 - density, dielectric constant, diffusion constant, ..

From simulation with parameter p

- simulate and get \mathcal{A}_p
- unhappiness (cost) is a function of p , so we have $c(p)$
 - $c(p) = |\mathcal{A}_{obs} - \mathcal{A}_p|$
 - or maybe $c(p) = (\mathcal{A}_{obs} - \mathcal{A}_p)^2$
- very concrete



- each point is result from a simulation
- noise / inaccuracy, not symmetric / linear

Example p is σ in $U(r_{ij}) = 4\varepsilon_{ij} \left(\left(\frac{\sigma_{ij}}{r_{ij}} \right)^{-12} - \left(\frac{\sigma_{ij}}{r_{ij}} \right)^{-6} \right)$

we would be adjusting the size of particles

parameters optimisation – boring ? easy ?

You would not choose p values randomly or by systematic search

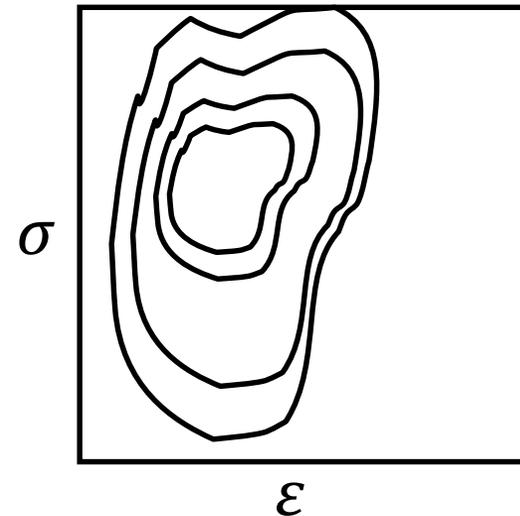
- (use a classic optimisation method)

Is this too easy and dull ?

- what you probably have is several parameters $c(p_1, p_2)$

$$U(r_{ij}) = 4\varepsilon_{ij} \left(\left(\frac{\sigma_{ij}}{r_{ij}} \right)^{-12} - \left(\frac{\sigma_{ij}}{r_{ij}} \right)^{-6} \right)$$

- measure the error/cost in 2D space



mapping parameter space

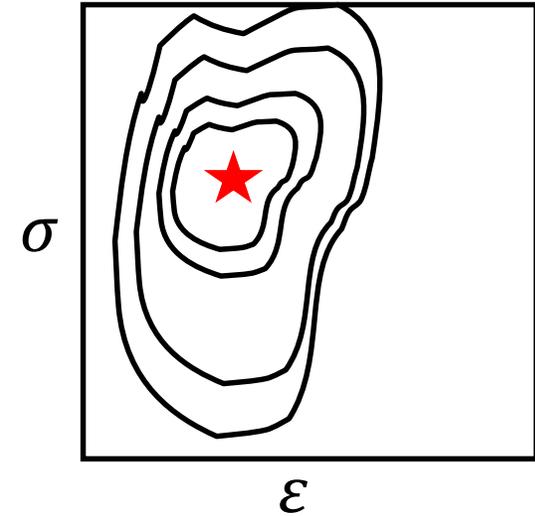
What does this tell us ?

- find best ε and σ
- see that ε is critical, σ less so

Practical implementation

- systematic search ? Inefficient
- automate the optimisation

Problems...



Problems with parameterisation

- scheme requires a believable measure of quality
- easy for two parameters
- possible for 3, 4 parameters
- very difficult for 100 parameters

Optimising for some properties

- you optimize for density
 - diffusion, free energy changes
 - all broken
 - you optimise based on 10 proteins
 - test of 11th - bad results (too small training set)

Different kind of score function

Change of style...

- questions on coarse-graining ?
- why is entropy an issue ? (numbers of particles / states)

- from nice ideas to dumb empiricism

Potentials of mean force

Potential of mean force ... knowledge based score functions

- very general
- history from atomistic simulations

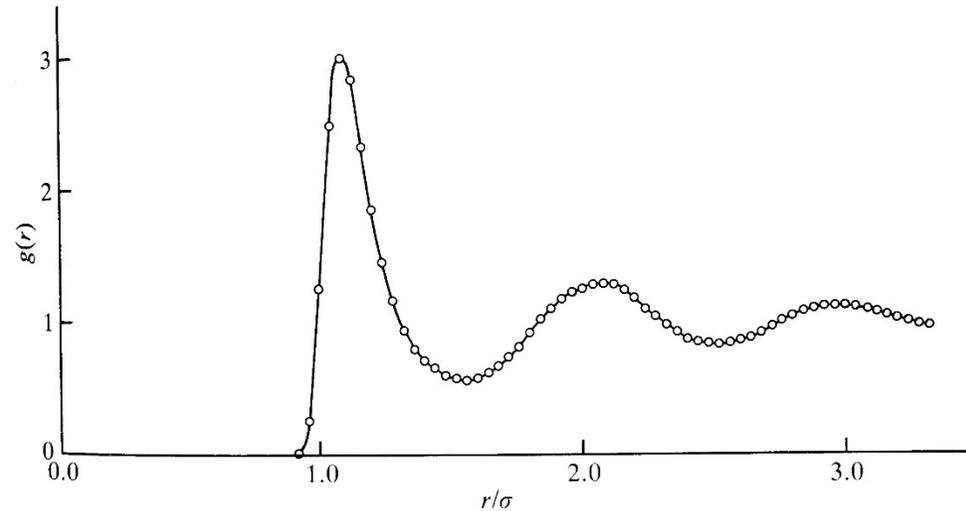
Basic idea .. easy

- from radial distribution function, to something like energy..

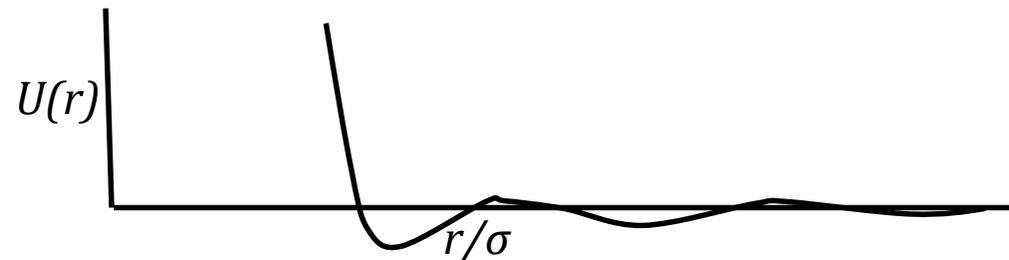
Intuitive version of potential of mean force

Radial distribution function $g(r)$

- probability of finding a neighbour at a certain distance

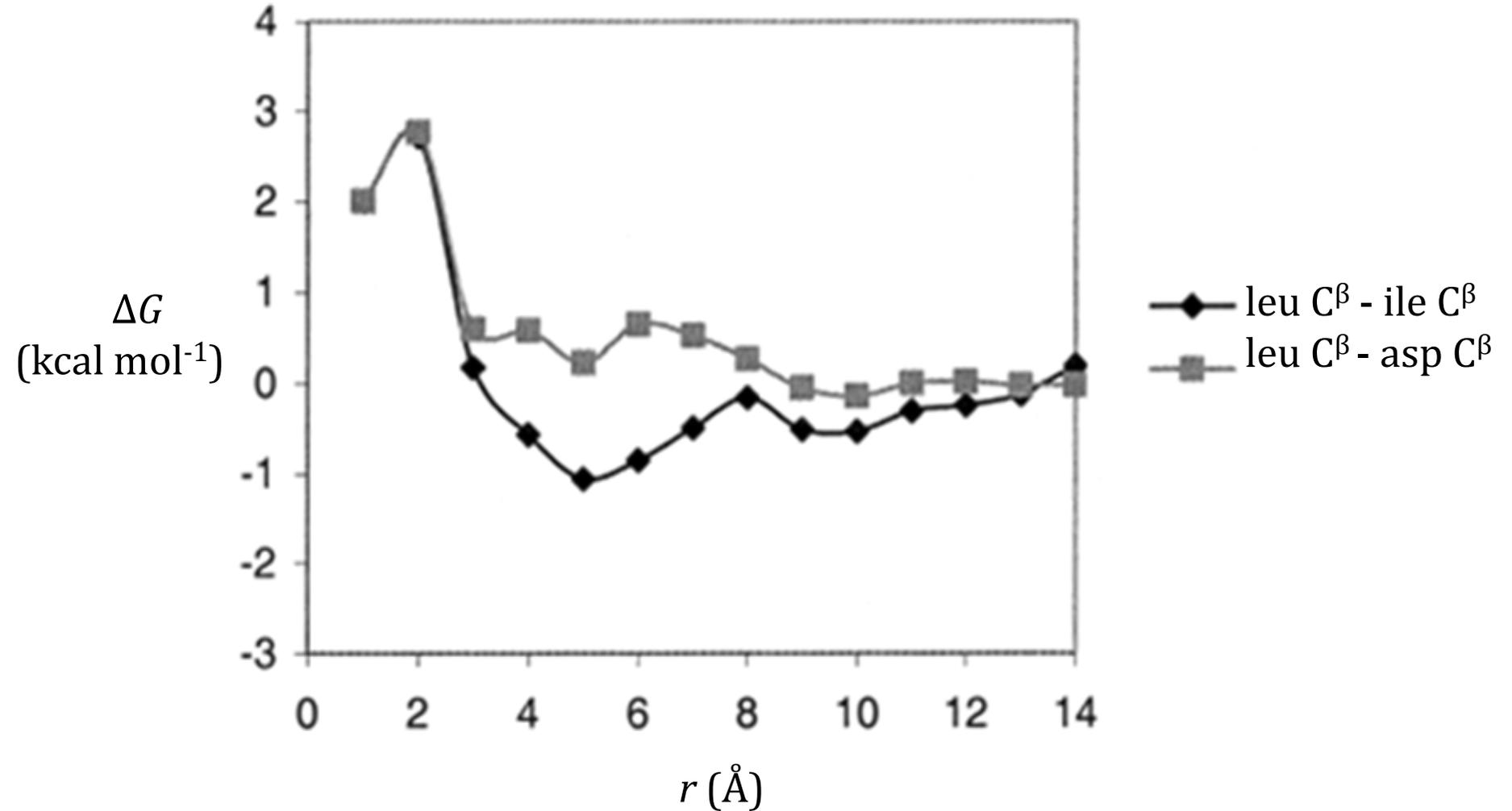


What does this suggest about energy ?



Goal

What I really want is..

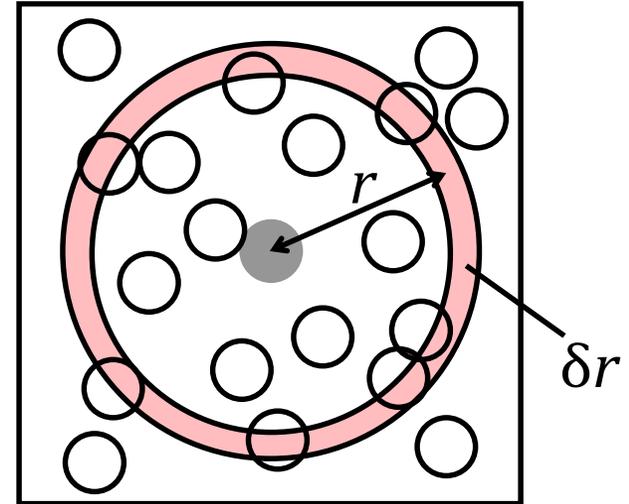


Radial distribution function

Formal idea $g(r) = \frac{N_{neighbours\ seen}(r)}{N_{neighbours\ expected}(r)}$

$$N_{expected} = \frac{V_{shell}}{V} N$$

- N particles
- V volume
- Calculating it?
 - define a shell thickness (δr)
 - around each particle
 - at each distance, count neighbours within shell



$$g(r) = \frac{V}{NV_{shell}} N_{shell}(r)$$

Rationale for potentials of mean force

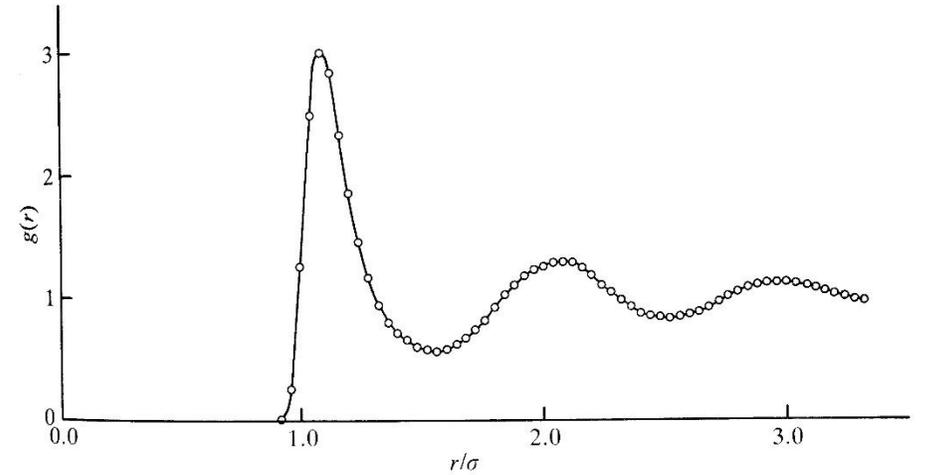
For state i compared to some reference x

$$\frac{p_i}{p_x} = \frac{e^{\frac{-E_i}{kT}}}{e^{\frac{-E_x}{kT}}} = e^{\frac{E_x - E_i}{kT}}$$

$$\ln \frac{p_i}{p_x} = \frac{E_x - E_i}{kT}$$

$$\Delta E = kT \ln \frac{p_i}{p_x}$$

Information in distribution function



Intuitive properties ?

- how likely is it that atoms get near to each other ($< \sigma$) ?
- what would a crystal look like ? (very ordered)
- what if interactions are
 - very strong (compared to temperature)
 - very weak
- Seems to reflect
 - strength of interactions / order

Relate this back to energy

Energy from $g(r)$

From statistical mechanics $g(r) = e^{\frac{-\Delta G}{kT}}$

- not quite textbook formulation – normally describe in terms of work
- what is the Δ in ΔG ?
 - the free energy change going from infinite separation to some distance r

How would we get $g(r)$?

- experiment ? sometimes
- simulation – easy – simulate at high resolution
- soon – protein data bank

Assumptions

- our system is at equilibrium

Generalising ideas of potential of mean force

What else can we do ?

- think of more interesting system (H_2O)

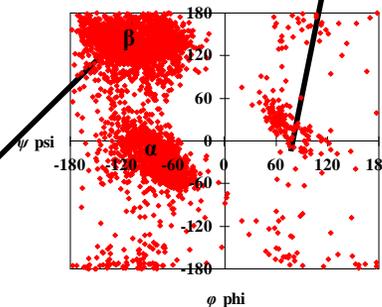
Would we express our function in terms of O ? H ?

- both valid
- could consider work done bringing an O to O, O to H, H to H
 - for fun on next page

More general..

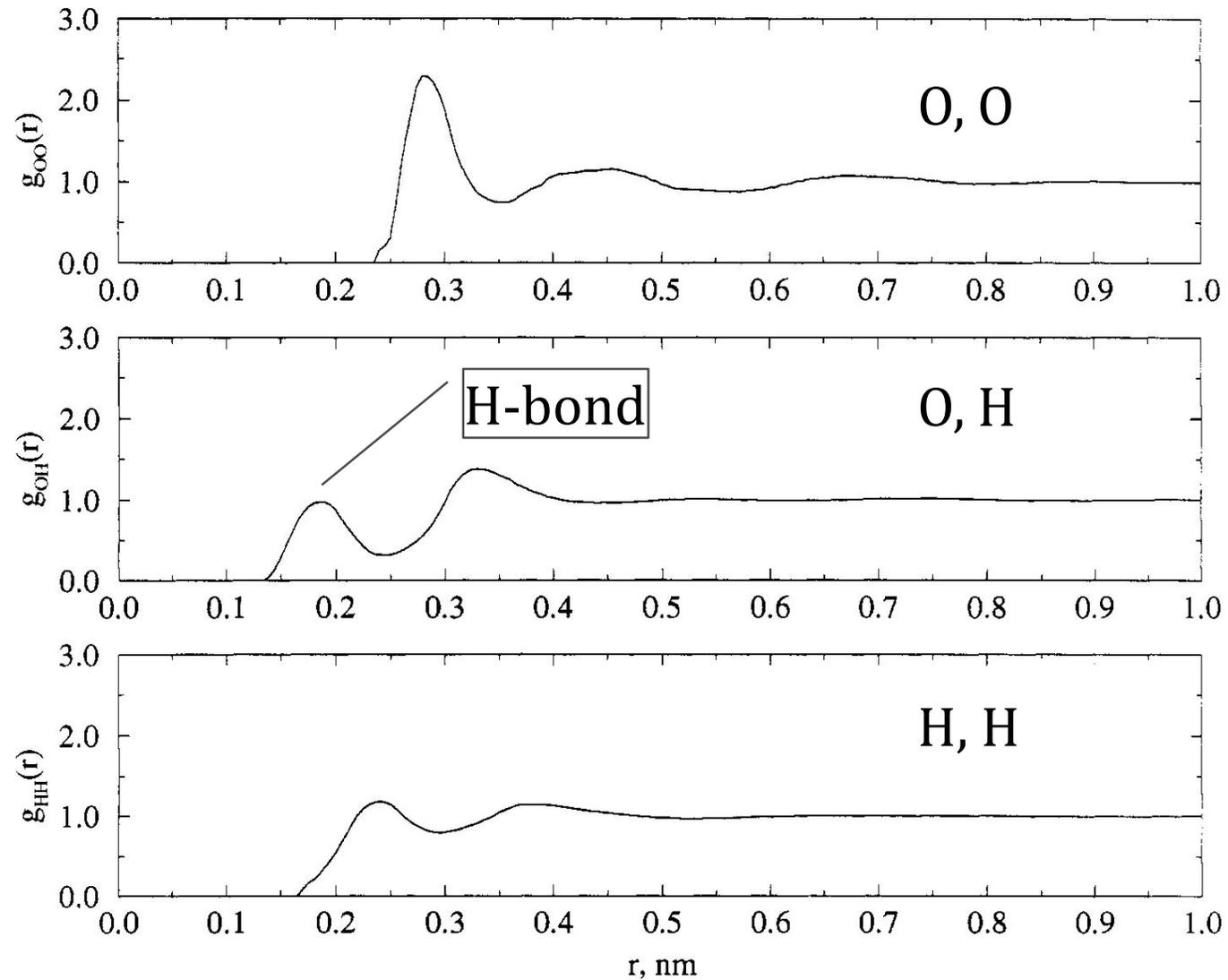
- are we limited to distances ? No
- example – ramachandran plot
- other atoms ? ...

high probability
/ low energy



low probability
/ high energy

radial distribution function (water)



Reformulating for our purposes

Can one use these ideas for proteins ?

Our goal ?

- a force field / score function for deciding if a protein is happy
- work with particles / interaction sites
- slightly different formulation
 - if I see a pair of particles close to each other,
 - is this more or less likely than random chance ?
 - treat pieces of protein like a gas
 - care about types of particles (unlike simple liquid)

Let us define...

Score energy formulation

$$W_{AB}(r) = -RT \ln \left(\frac{N_{AB}^{obs}(r \pm \delta r)}{N_{AB}^{exp}(r \pm \delta r)} \right)$$

N_{AB}^{obs} how many times do we see

- particles of types A and B
- distance r given some range δr

N_{AB}^{exp} how often would you expect to see AB pair at r ?

- remember Boltzmann statistics

This is not yet an energy / score function !

- it is how to build one

Intuitive version

- Cl^- and Na^+ in water like to interact (distance r^0)
- N_{AB}^{obs} is higher than random particles
- $W_{\text{ClNa}}(r)$ is more negative at r^0

Details of formulation

$$W_{AB}(r) = -RT \ln \left(\frac{N_{AB}^{obs}(r \pm \delta r)}{N_{AB}^{exp}(r \pm \delta r)} \right)$$

- looks easy, but what is N^{exp} ?

Maybe fraction of particles is a good approximation

$$N_{AB}^{exp} = N_{all} X_{Na} X_{Cl} \quad (\text{use mole fractions})$$

- use this idea to build a protein force field / score function

Protein score function

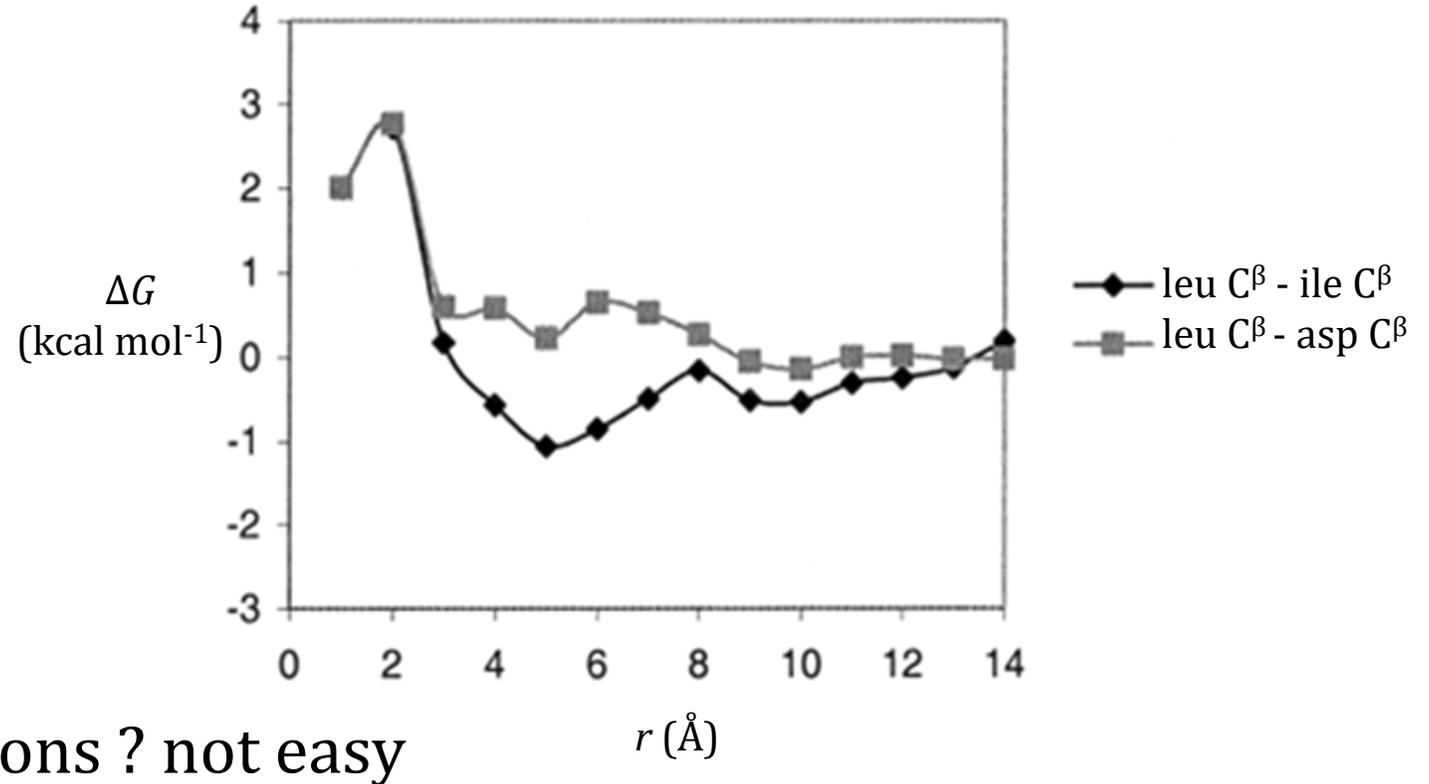
Arbitrarily

- define interaction sites as one per residue
 - maybe at C^α or C^β
- collect set of structures from protein data bank
- define a distance (4 Å) and range (± 0.5 Å)
- count how often do I see
 - gly-gly at this range, gly-ala, gly-X, X-Y ...
 - gives me N^{obs}
 - how many pairs of type gly-gly, gly-ala, gly-X, X-Y... are there ?
 - gives me N^{exp}
 - repeat for 5 Å, 6 Å, ...
- resulting score function...

final score function

For every type of interaction AB ($20 \times 21 / 2$)

- set of $W_{AB}(r)$



All ingredients in place

- can we use this for simulations ? not easy
- can we use to score a protein ? yes

Names

- Boltzmann-based, knowledge based

Applying knowledge-based score function

Take your protein

- for every pair of residues
 - calculate C^β C^β distance (for example)
 - look up type of residues (ala-ala, trp-ala, ...)
 - look up distance range
 - add in value from table

What is intuitive result from a

- a sensible protein / a misfolded protein ?

Is this a real force field ? yes

Is this like the atomistic ones ? no

- there are no derivatives $\left(\frac{dU}{dr}\right)$
- it is not necessarily defined for all coordinates

Practical Problems Boltzmann score functions

Do we have enough data ?

- how common are Asp-Asp pairs at short distance ?

How should we pick distance ranges ?

How far should we look ? (r_{AB}) ?

What are my interaction sites ?

- C^α ? C^β ? both ?

Data bias

- Can I ever find a representative set of proteins ?
 - PDB is a set of proteins which have been crystallised

Reminder

- we want low-resolution score functions
- if we work in a Boltzmann framework, we work with real energies
- everything ends up as $\frac{p_i}{p_j} = e^{-\frac{\Delta E}{RT}}$ or here $\Delta E = -RT \ln \frac{p_i}{p_j}$ or $\Delta E = -RT \ln \frac{N_{obs}}{N_{exp}}$
- we are comparing against what you expect from random events without interactions p_j
- work with kJ mol^{-1} , we can
 - make real energetic predictions (kinetics, equilibria)
 - combine with other energy terms

Problems of Principle

Boltzmann statistics

- is the protein data bank a set of structures at equilibrium ?

Is this a potential of mean force ? Think of Na, Cl example

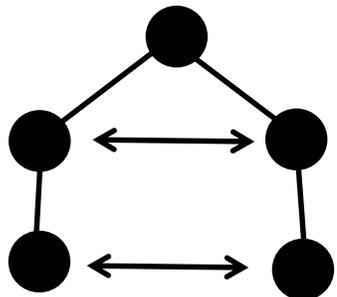
- that is a valid PMF since we can average over the system

Energy / Free energy

- how real ?

N^{exp} ? how should it be calculated ?

- is the fraction of amino acid a good estimate ? No.
- there are well known effects.. Examples



$i, i+2$

$i, i+4$ very different statistics

Boltzmann based scores: improvements / applications

- collect data separately for $(i, i+2)$, $(i, i+3)$, ...
 - problems with sparse (missing) data
- collect data on angles
- collect data from different atoms
- collect protein – small molecule data

Are these functions useful ?

- not perfect, not much good for simulation
- we can take any coordinates and calculate a score
 - directly reflects how likely the coordinates are
- threading / fold recognition / model quality

Parameterising summary

- Inventing a score function / force field needs parameters
- totally invented (Crippen, Kuntz, ...)
- optimisation / systematic search
- statistics + Boltzmann distribution

Summary of low-resolution force fields

Properties

- do we always need a physical basis ?
- do we need physical score (energy) ?

Questions

- pick interaction sites
- pick interaction functions / tables

What is your application ?

- simulation
 - reproducing a physical phenomenon (folding, binding)
- scoring coordinates

Parameterisation

- Averaging, optimisation, potentials of mean force

Next – less physical