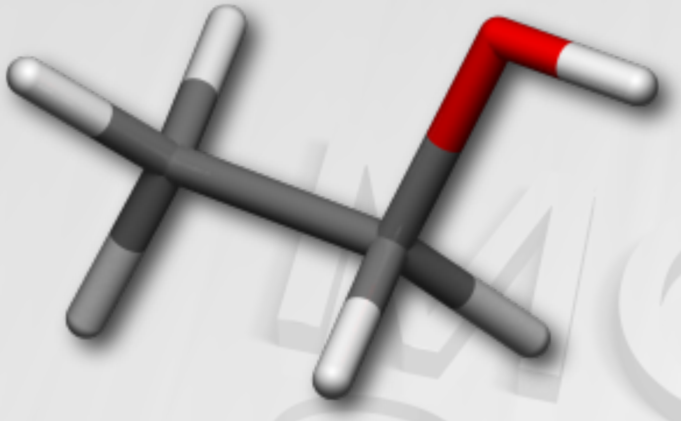
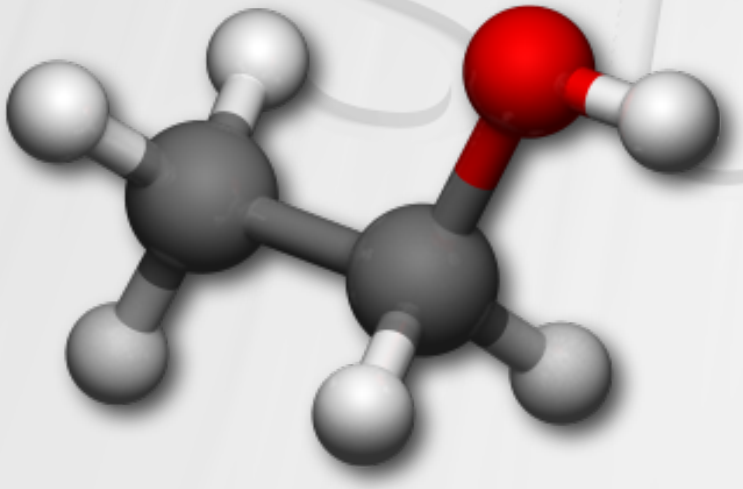
MOLECULAR SURFACES

Theoretical Seminar

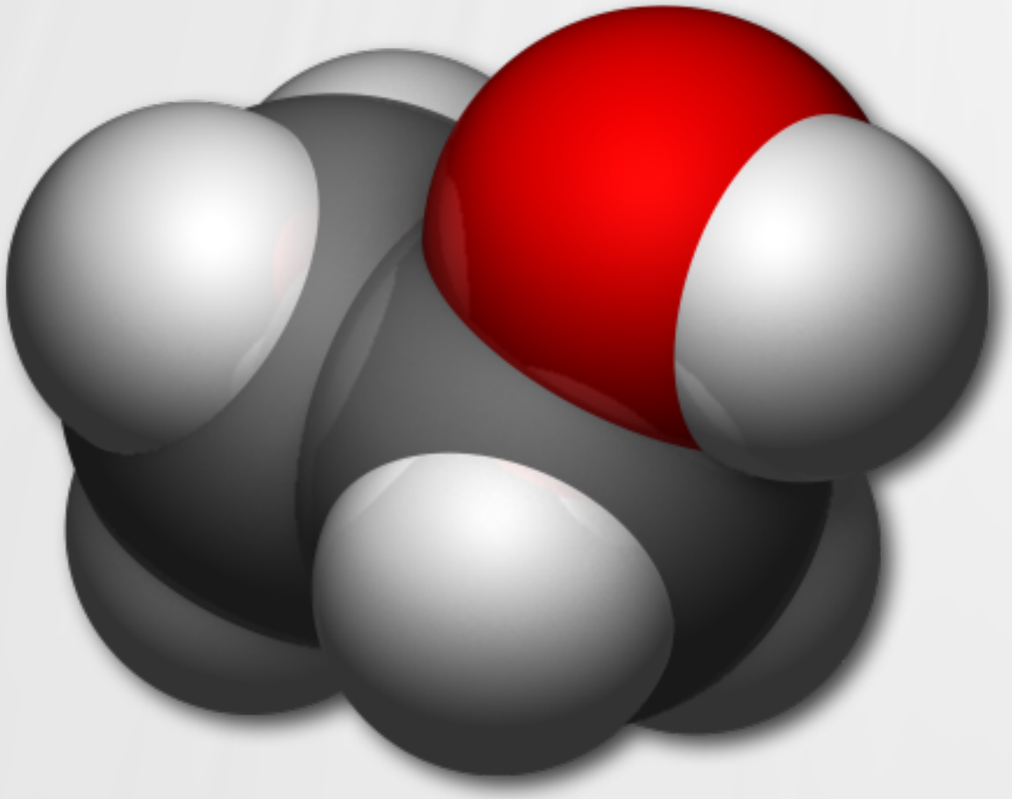
David Poger



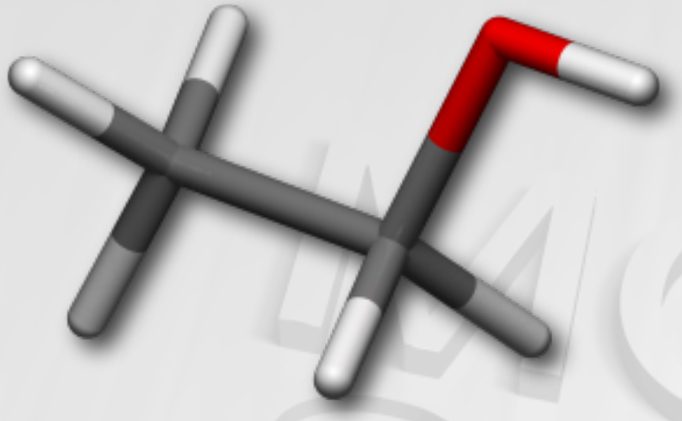
Line model
Licorice model



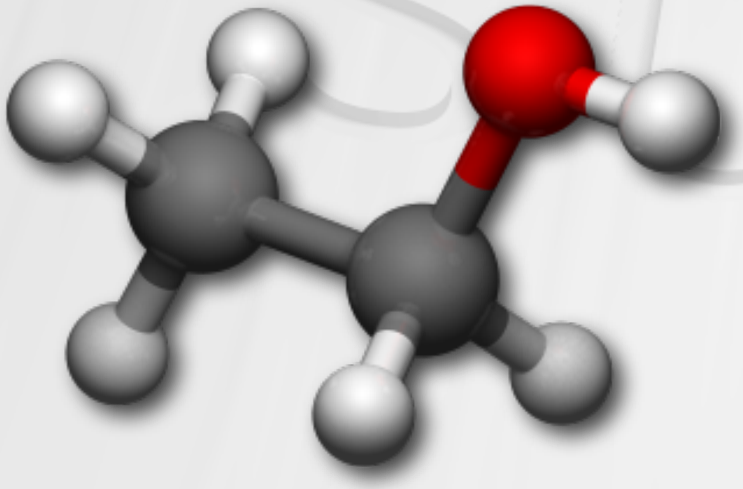
Ball-and-stick model
CPK model *(R. Corey, L. Pauling & W. Koltun)*



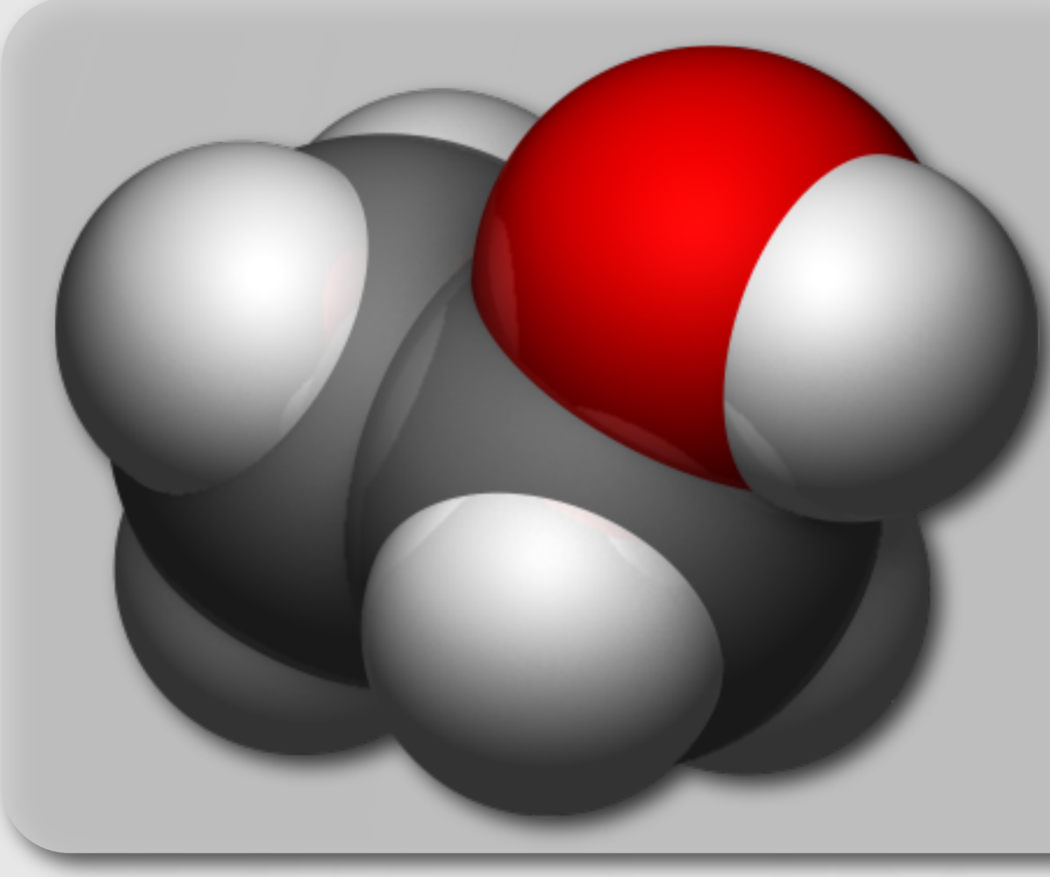
van der Waals model
Ball model
CPK model *(R. Corey, L. Pauling & W. Koltun)*
Space-filling model
Calotte model
Surface model



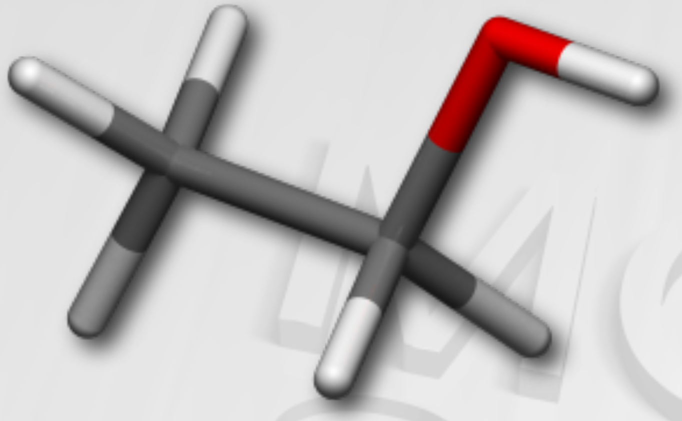
Line model
Licorice model



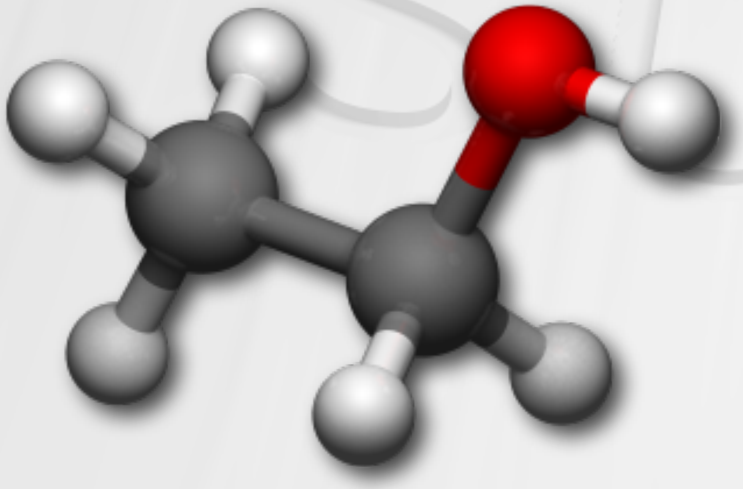
Ball-and-stick model
CPK model *(R. Corey, L. Pauling & W. Koltun)*



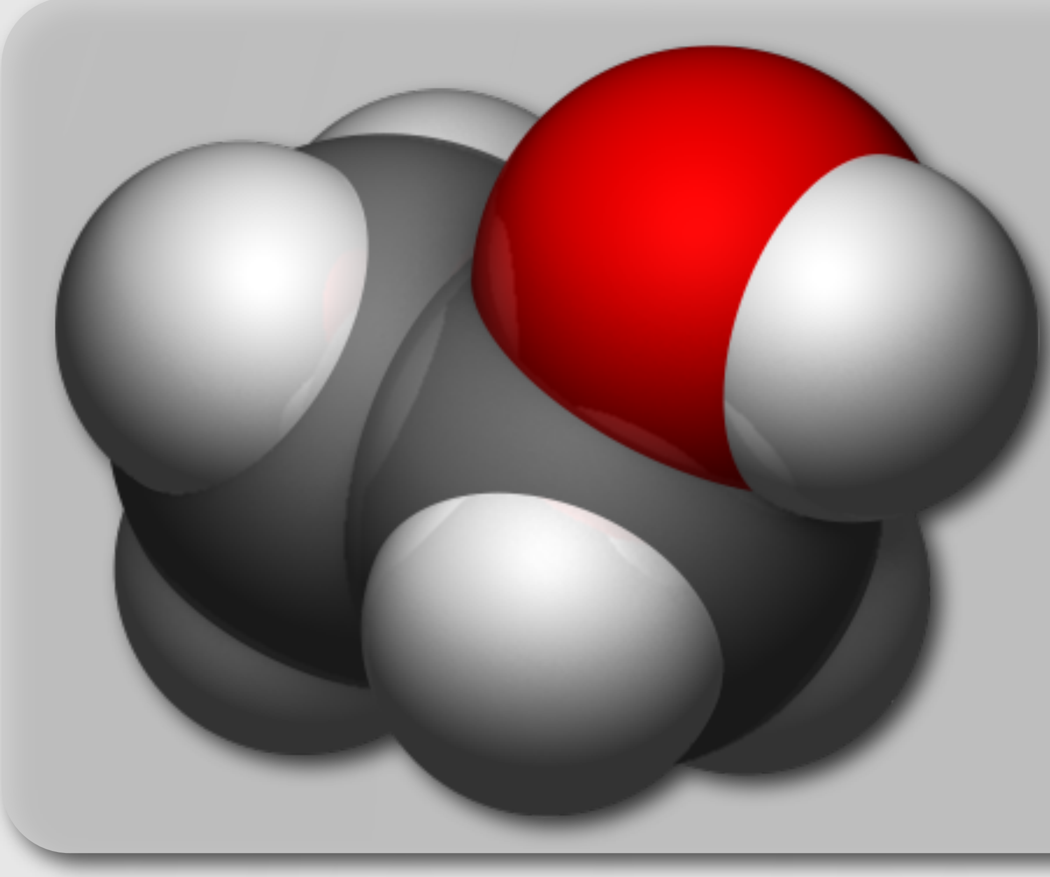
van der Waals model
Ball model
CPK model *(R. Corey, L. Pauling & W. Koltun)*
Space-filling model
Calotte model
Surface model



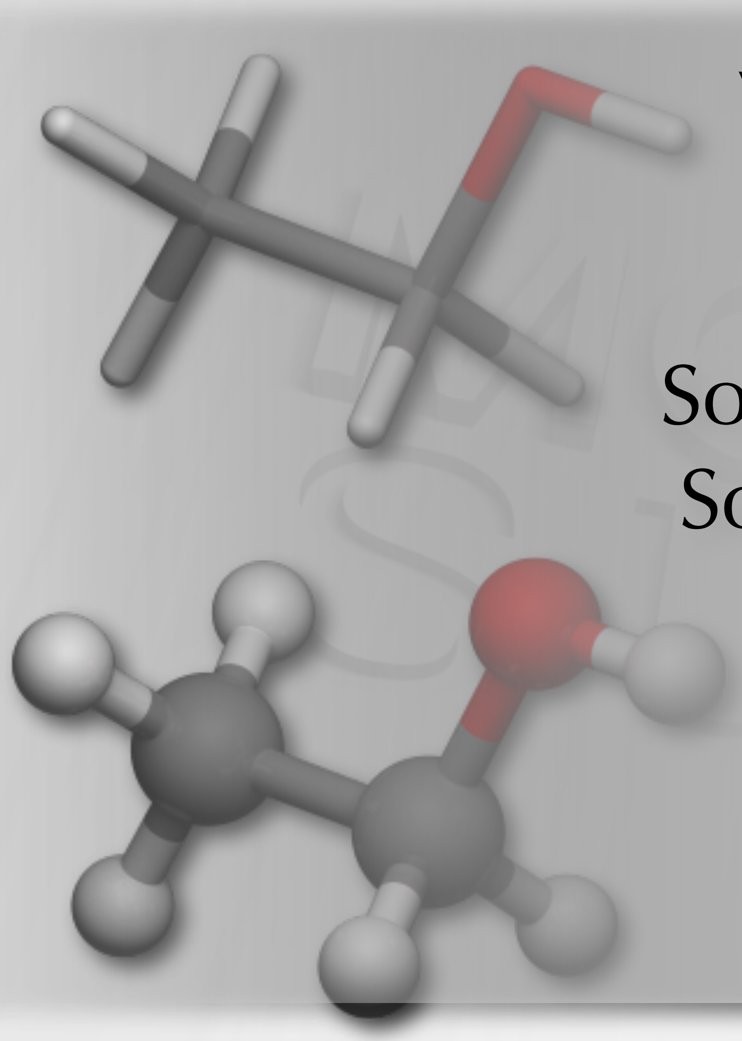
Line model
Licorice model



Ball-and-stick model
CPK model *(R. Corey, L. Pauling & W. Koltun)*



van der Waals model
Ball model
CPK model *(R. Corey, L. Pauling & W. Koltun)*
Space-filling model
Calotte model
Surface model



van der Waals surface

Connolly surface

Lee-Richards surface

Solvent-accessible surface

Solvent-excluded surface

Interstitial surface

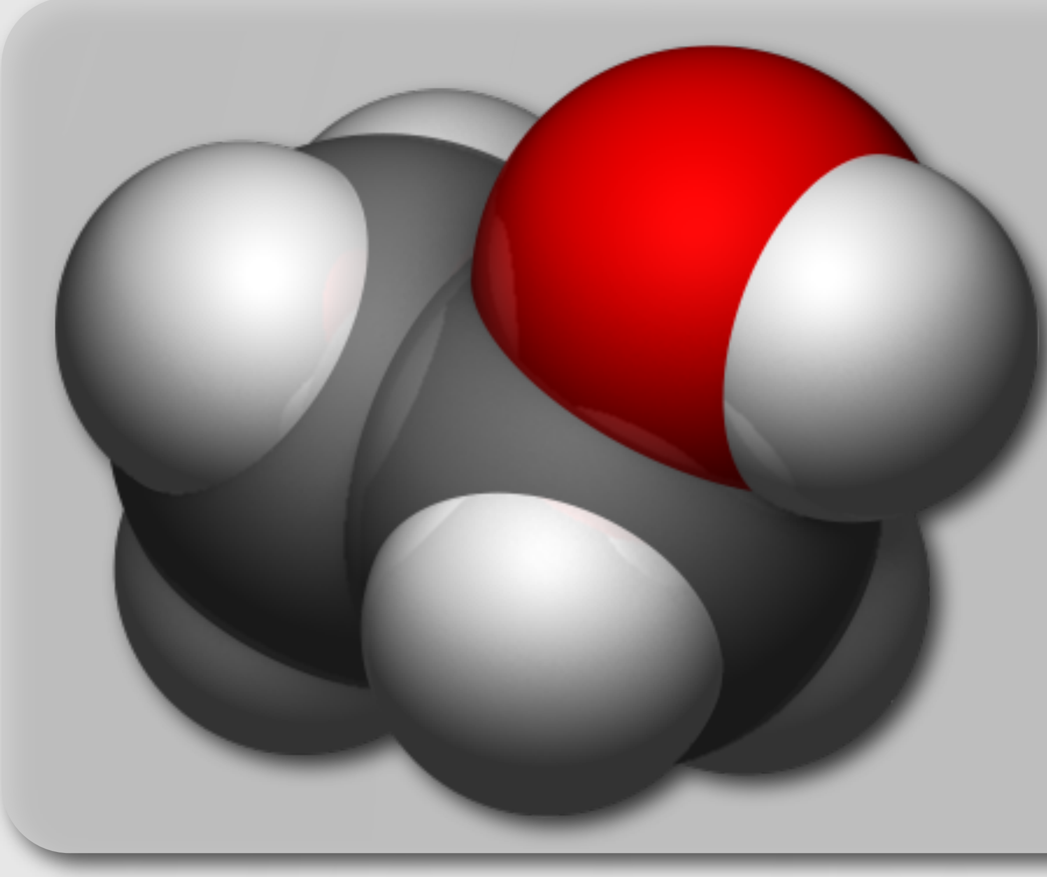
Molecular surface

Contact surface

Re-entrant surface

model

(R. Corey, L. Pauling & W. Koltun)



van der Waals model

Ball model

CPK model *(R. Corey, L. Pauling & W. Koltun)*

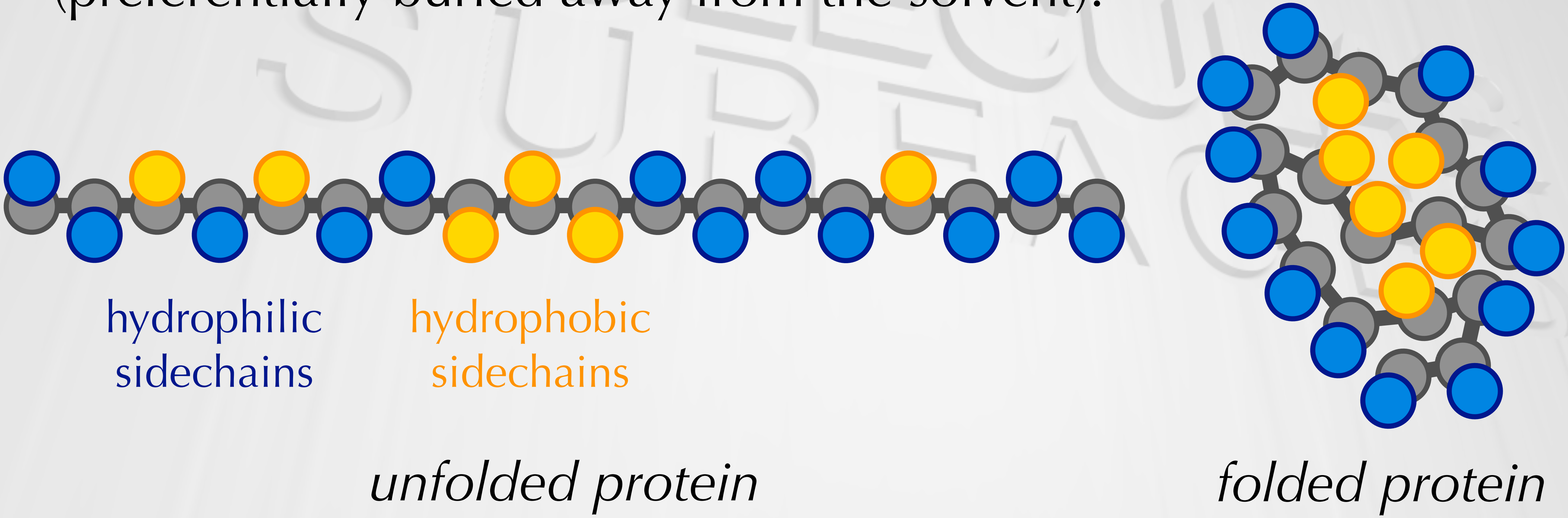
Space-filling model

Calotte model

Surface model

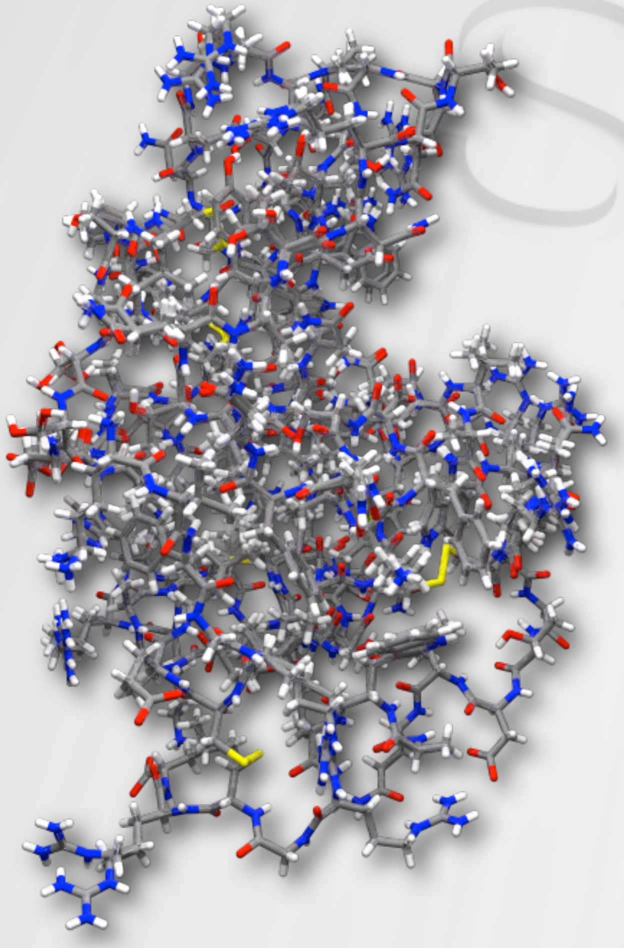
WHY STUDYING THE SURFACE OF PROTEINS?

- Historically, the interest on protein surfaces came from studies on protein folding and packing of hydrophobic aminoacids (preferentially buried away from the solvent).

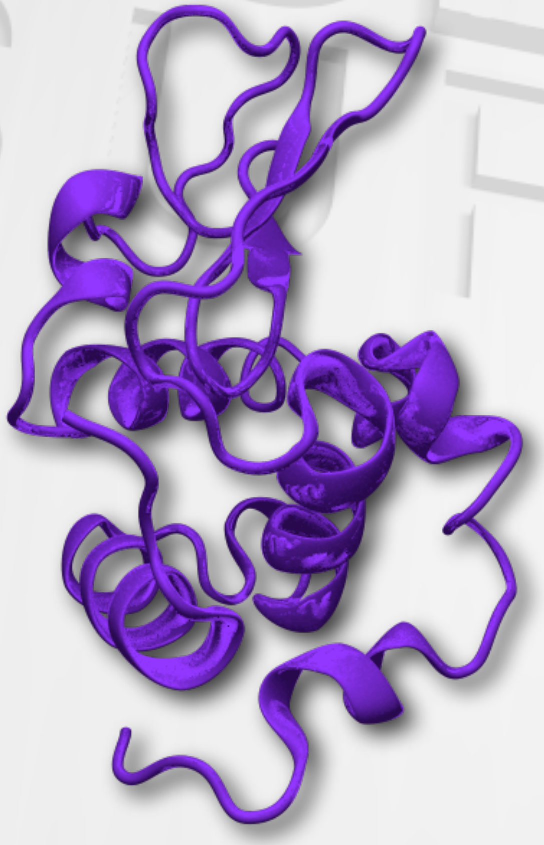


- Solvent accessibility as a way to quantify hydrophobic burial (Lee, B. and Richards, F. M. (1971) *J. Mol. Biol.* 55, 379–400): "*The topology of the surface of a protein is intimately related to its function; [...] the solvent–protein interface is almost certainly related to the structure of the native molecule.*"

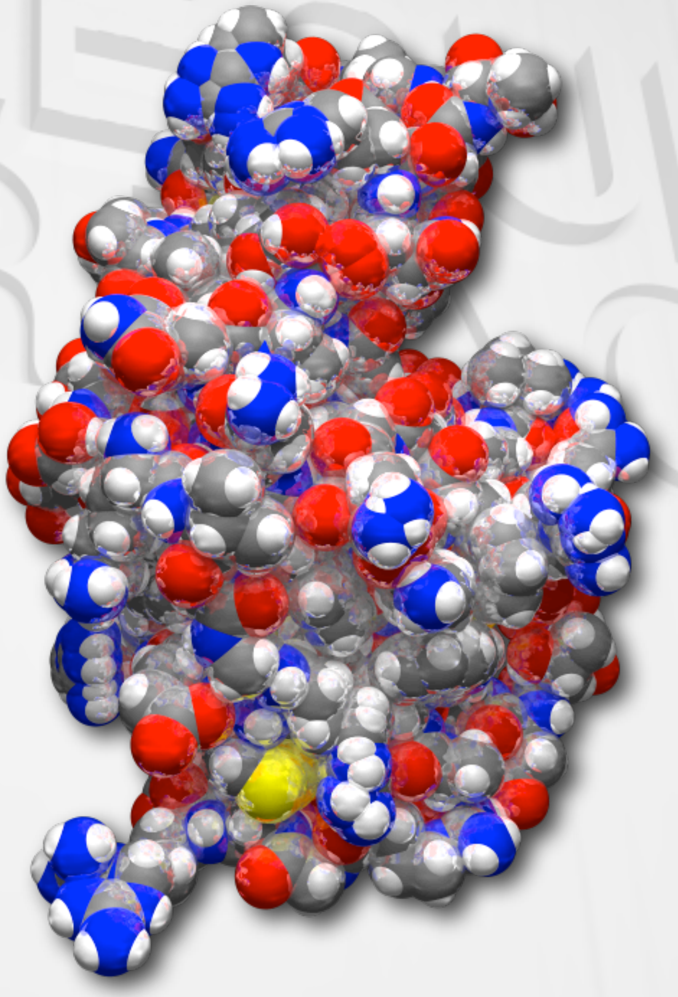
What You (May) Have Already Visualised



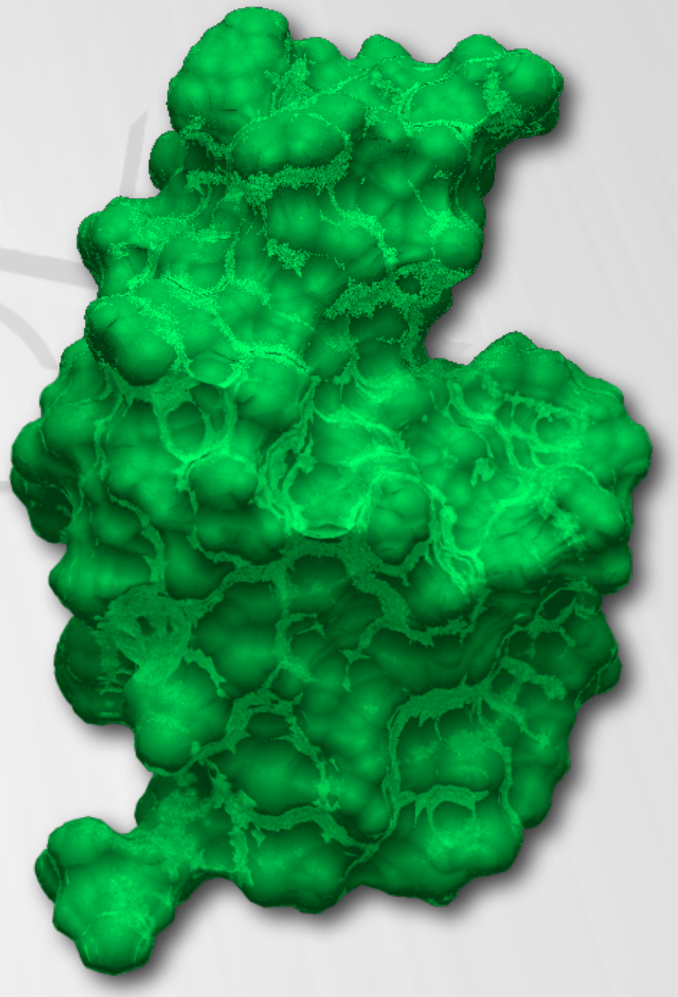
All the atoms



Fold



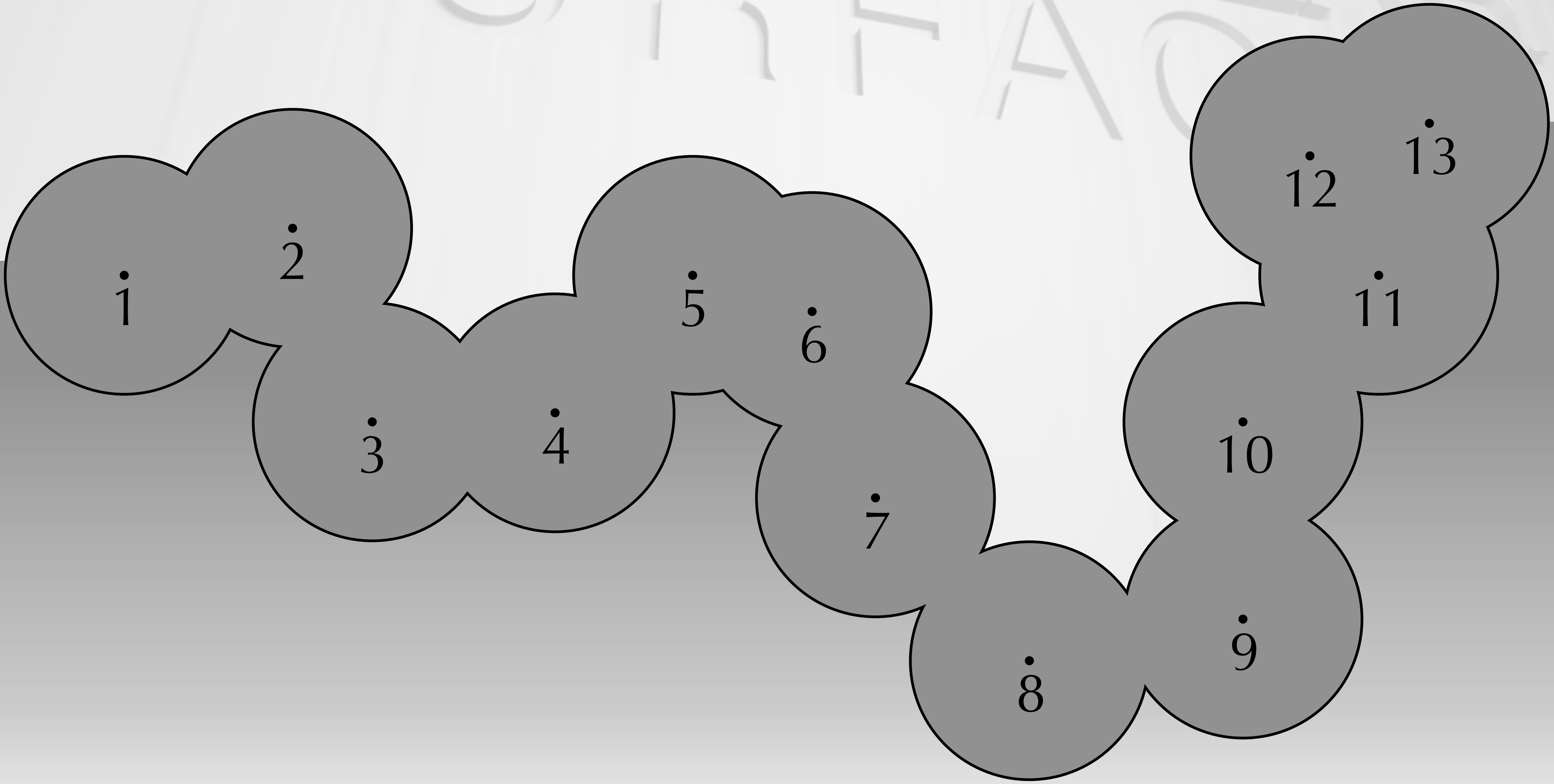
van der Waals



Surface

Hen egg white lysozyme – 0.65 Å (PDB 2vb1)

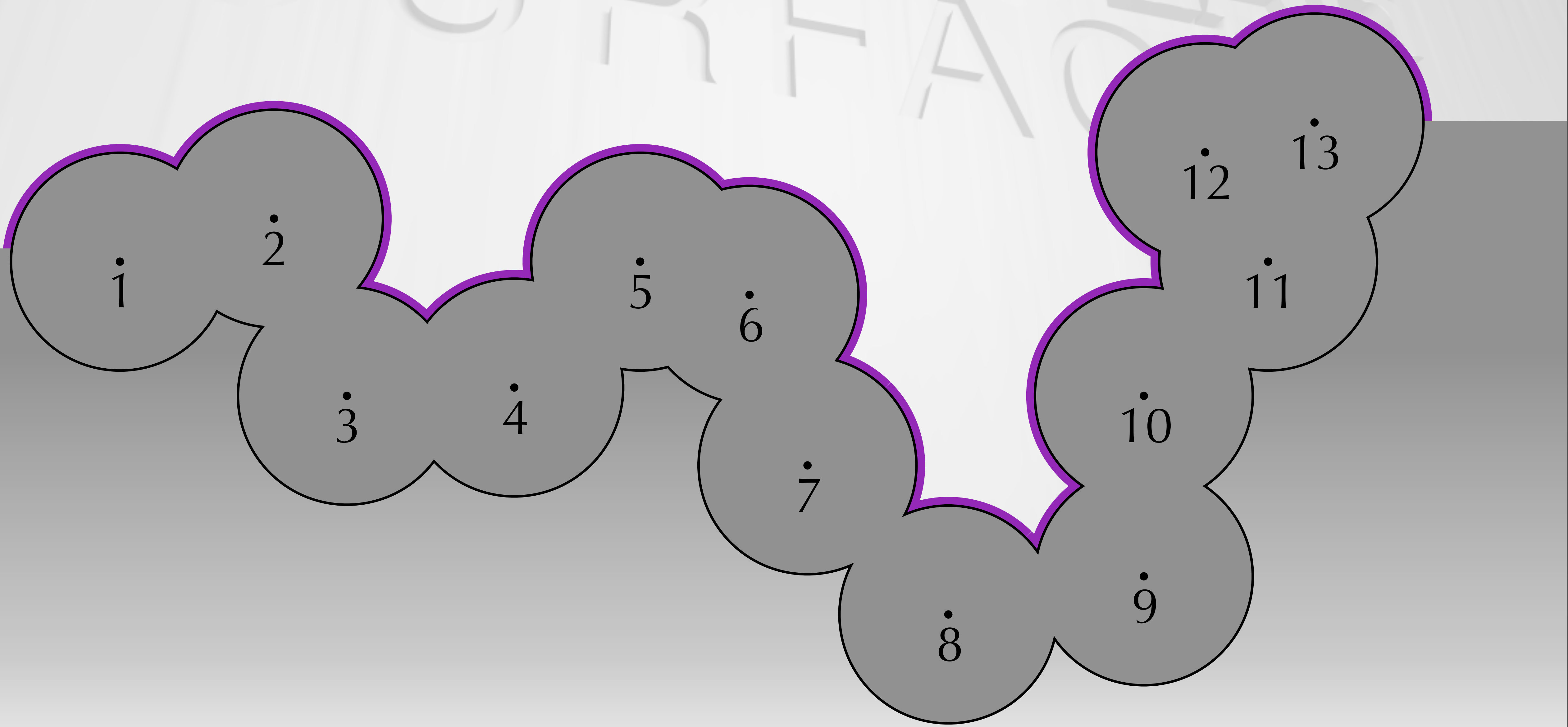
DEFINITION OF MOLECULAR SURFACES



inside of the protein

Definition of Molecular Surfaces

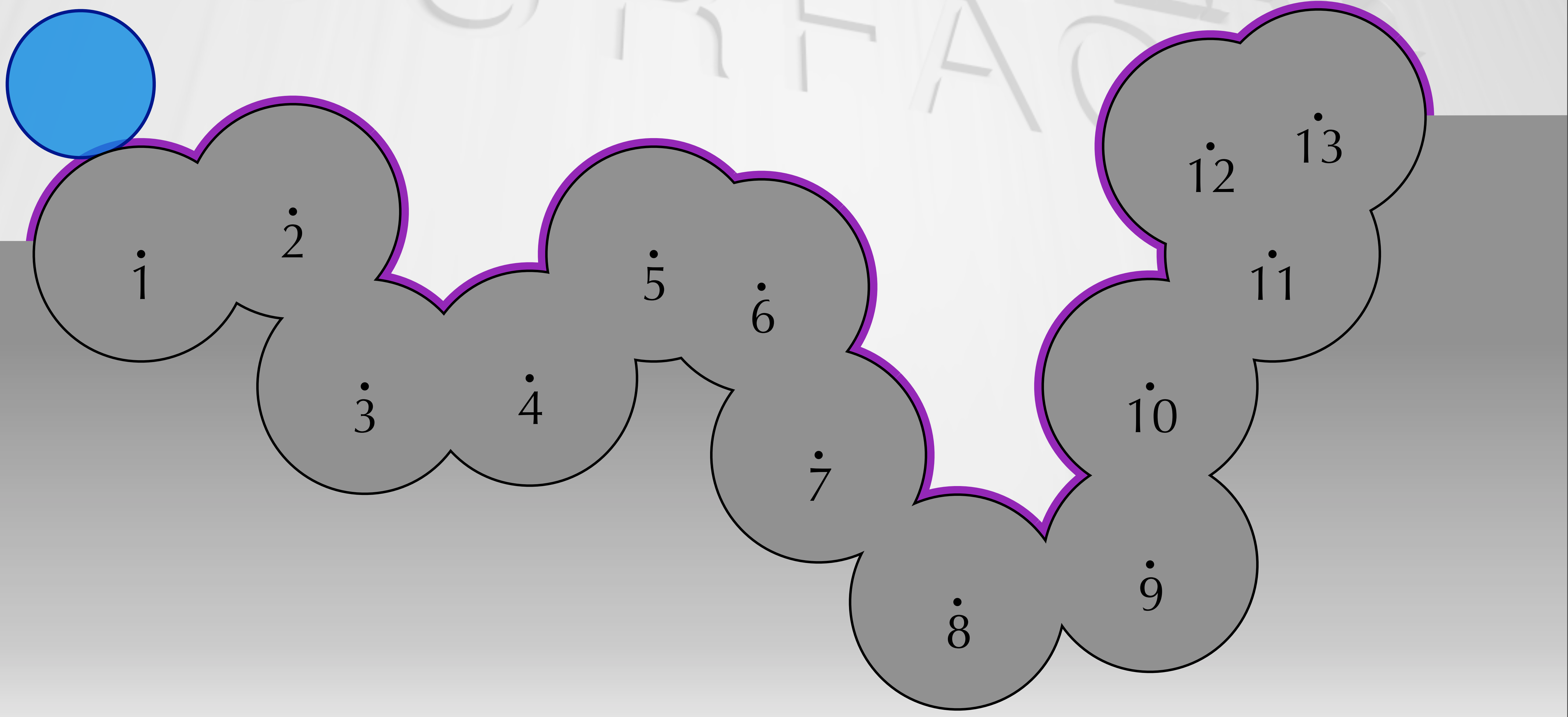
van der Waals surface



inside of the protein

Definition of Molecular Surfaces

van der Waals surface

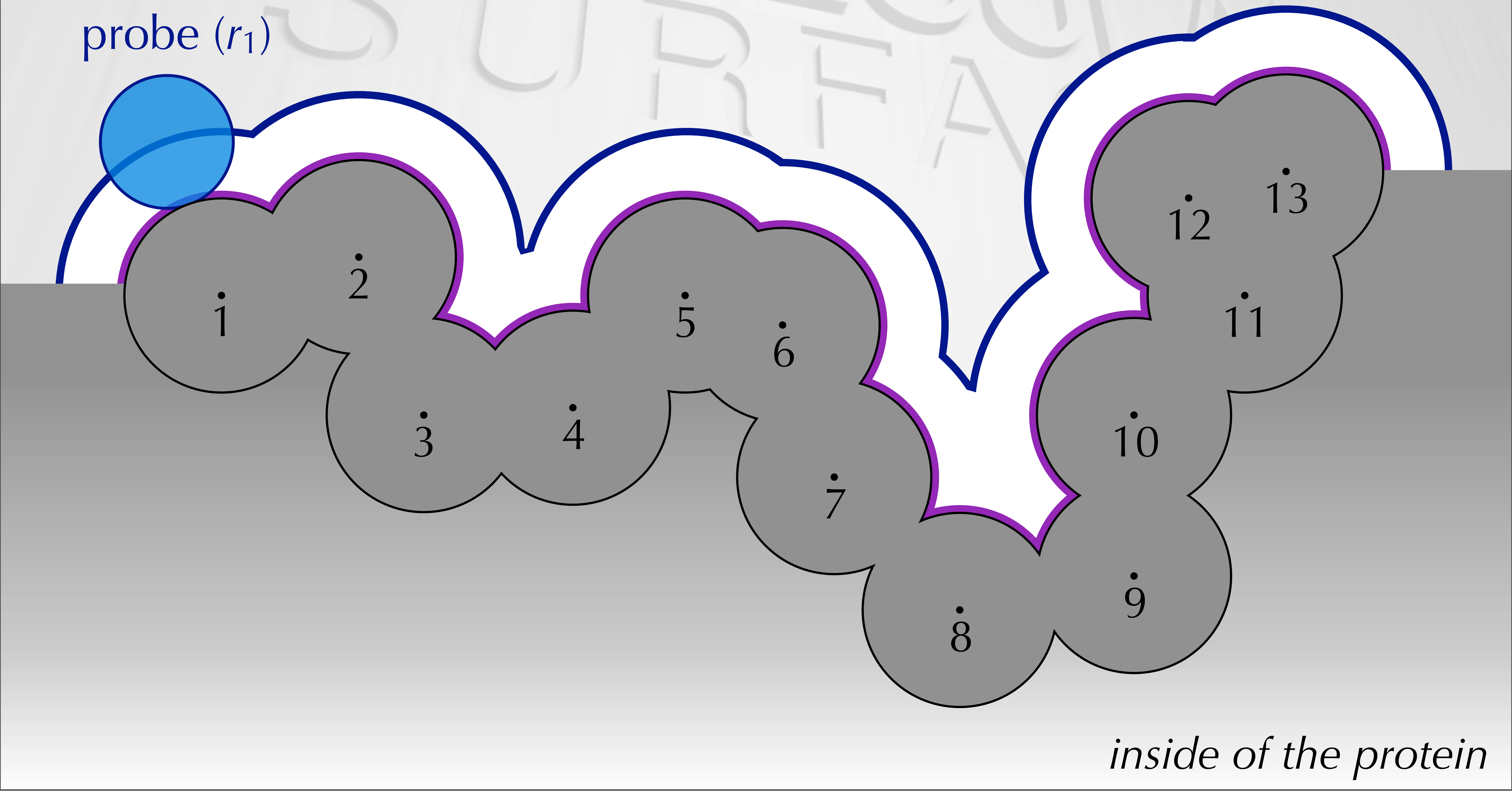


Definition of Molecular Surfaces

van der Waals surface

(solvent-)accessible surface

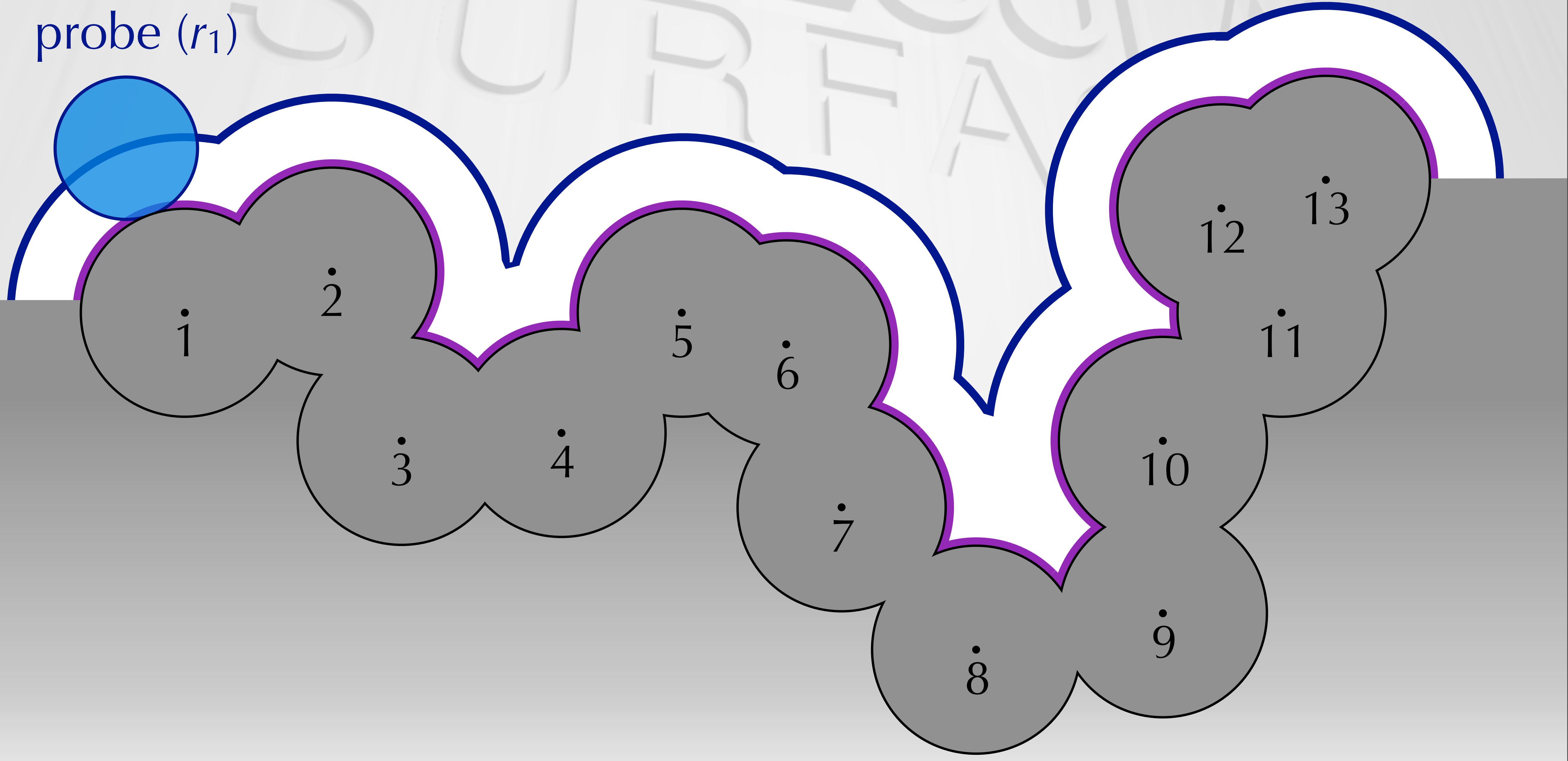
probe (r_1)



Definition of Molecular Surfaces

van der Waals surface

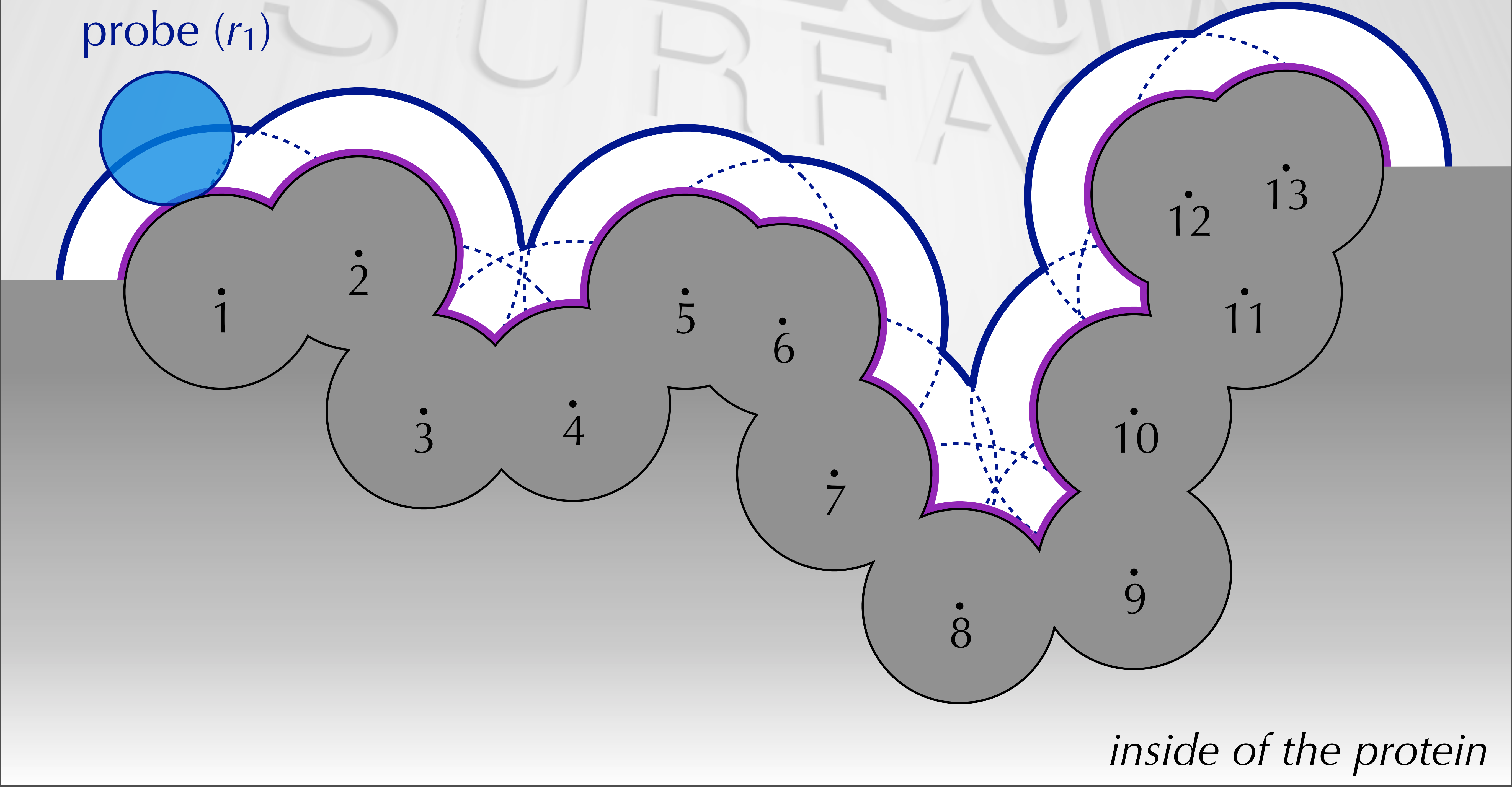
(solvent-)accessible surface



Definition of Molecular Surfaces

van der Waals surface

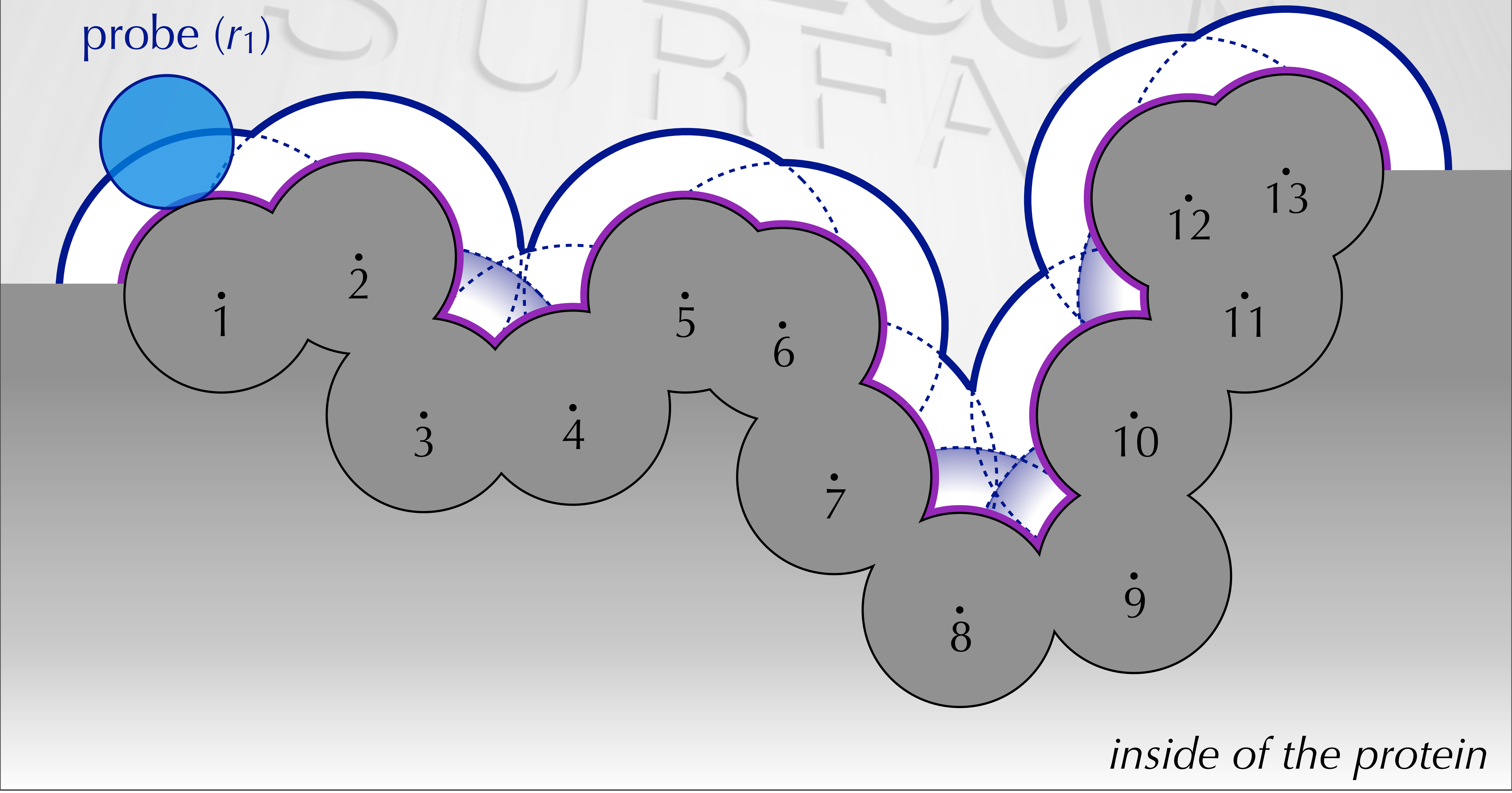
(solvent-)accessible surface



Definition of Molecular Surfaces

van der Waals surface

(solvent-)accessible surface

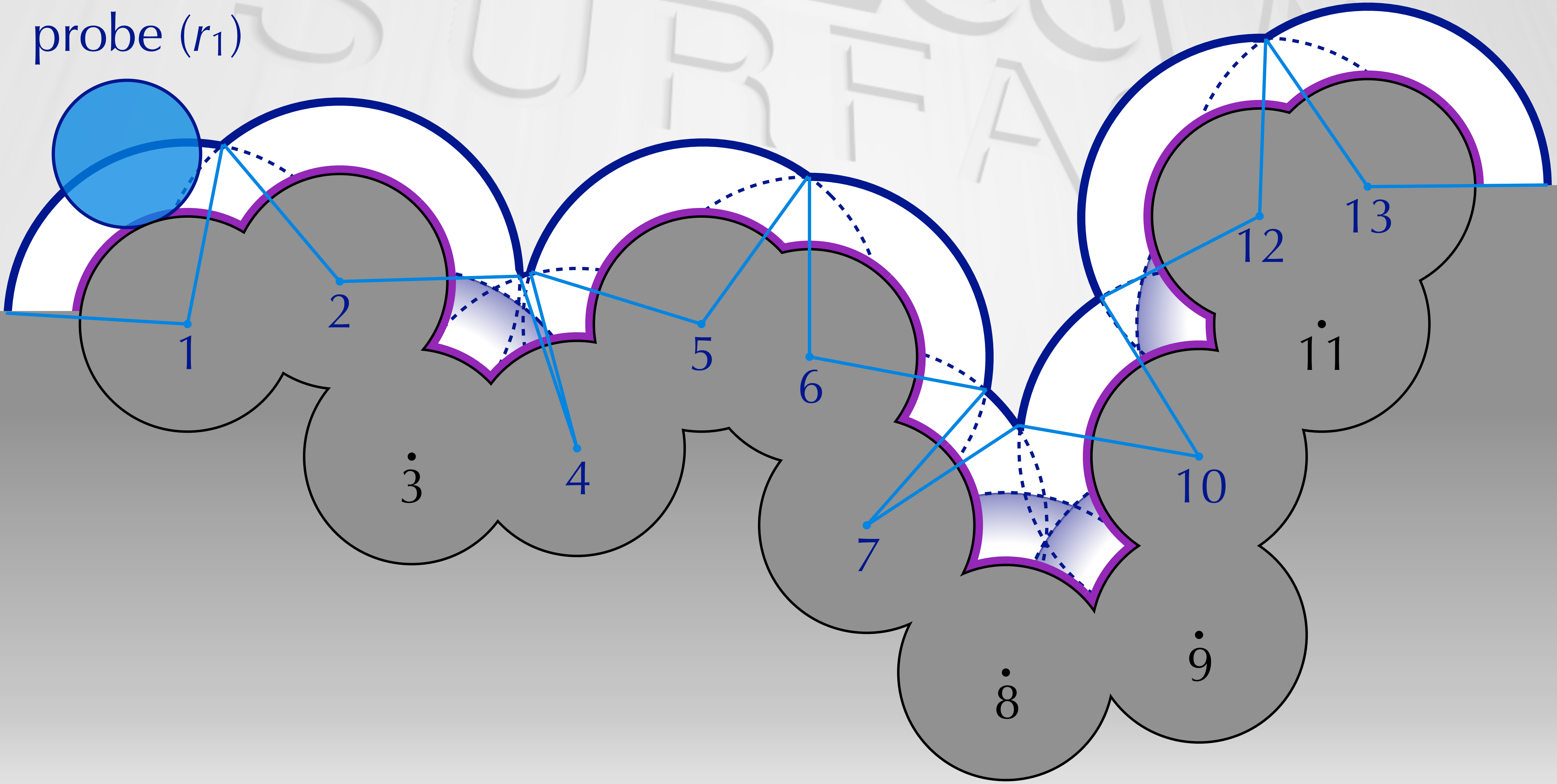


Definition of Molecular Surfaces

van der Waals surface

(solvent-)accessible surface

probe (r_1)



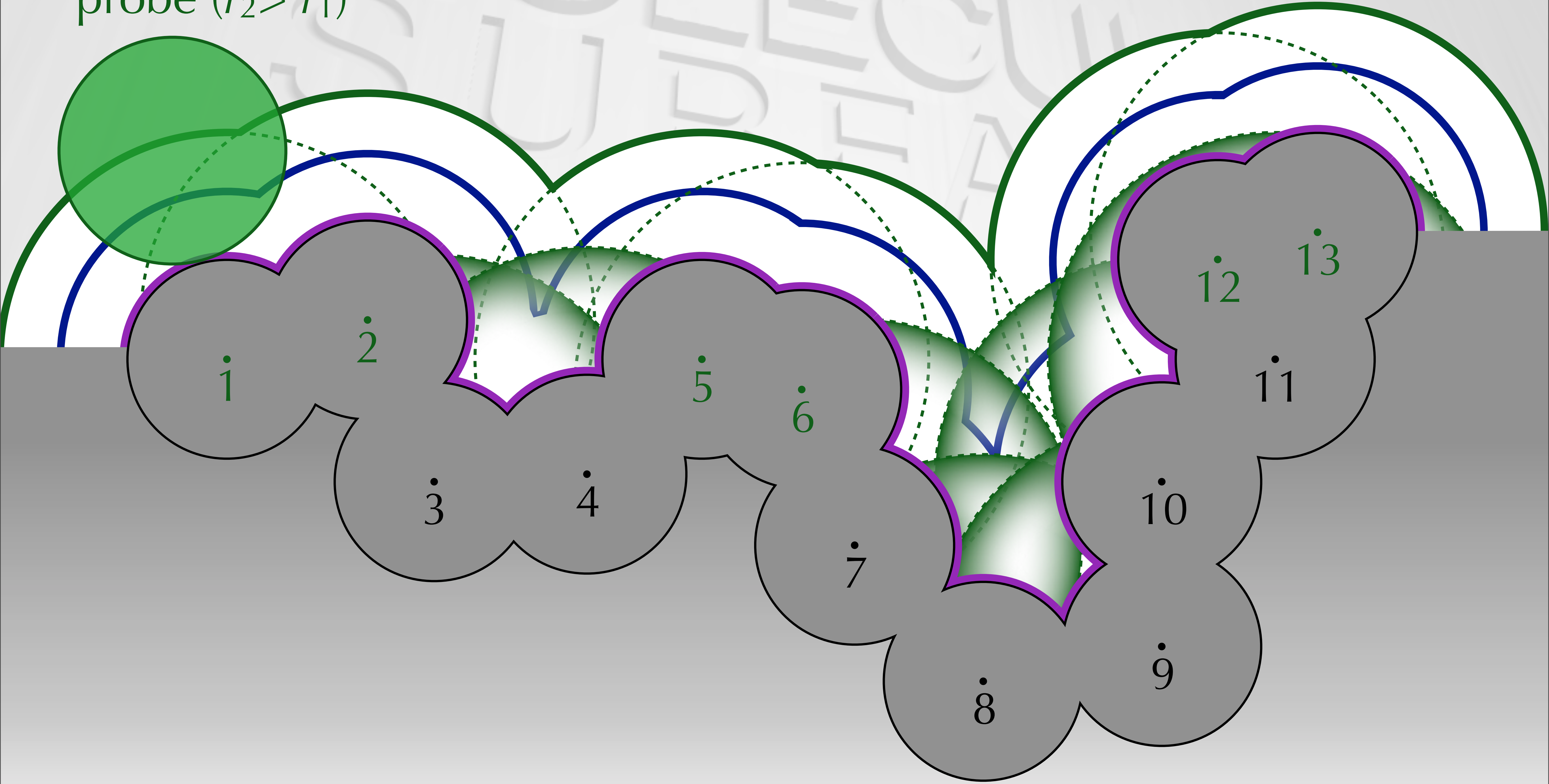
inside of the protein

Definition of Molecular Surfaces

van der Waals surface

(solvent-)accessible surface

probe ($r_2 > r_1$)



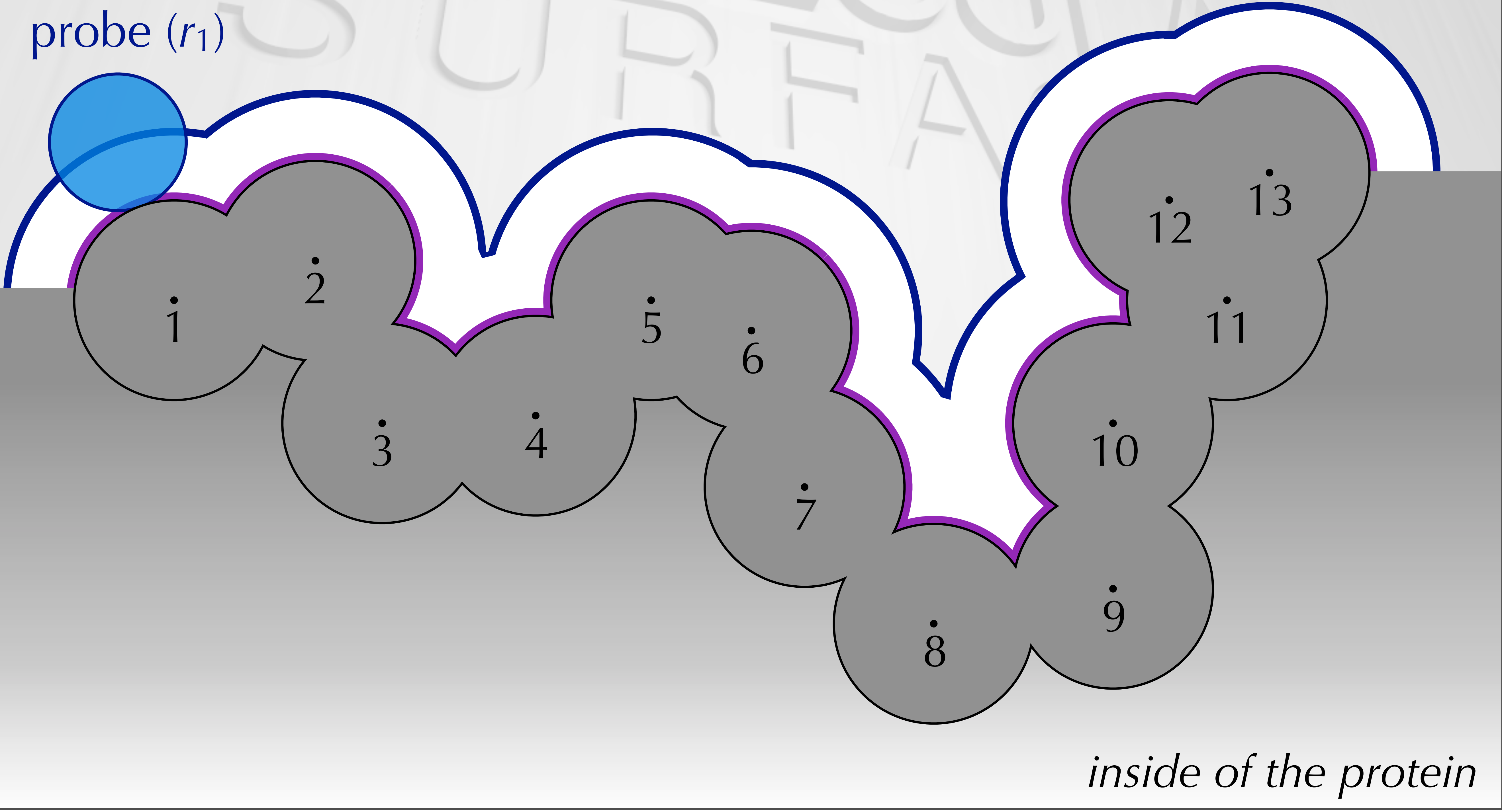
inside of the protein

Thursday, 3 June 2010

Definition of Molecular Surfaces

van der Waals surface

(solvent-)accessible surface



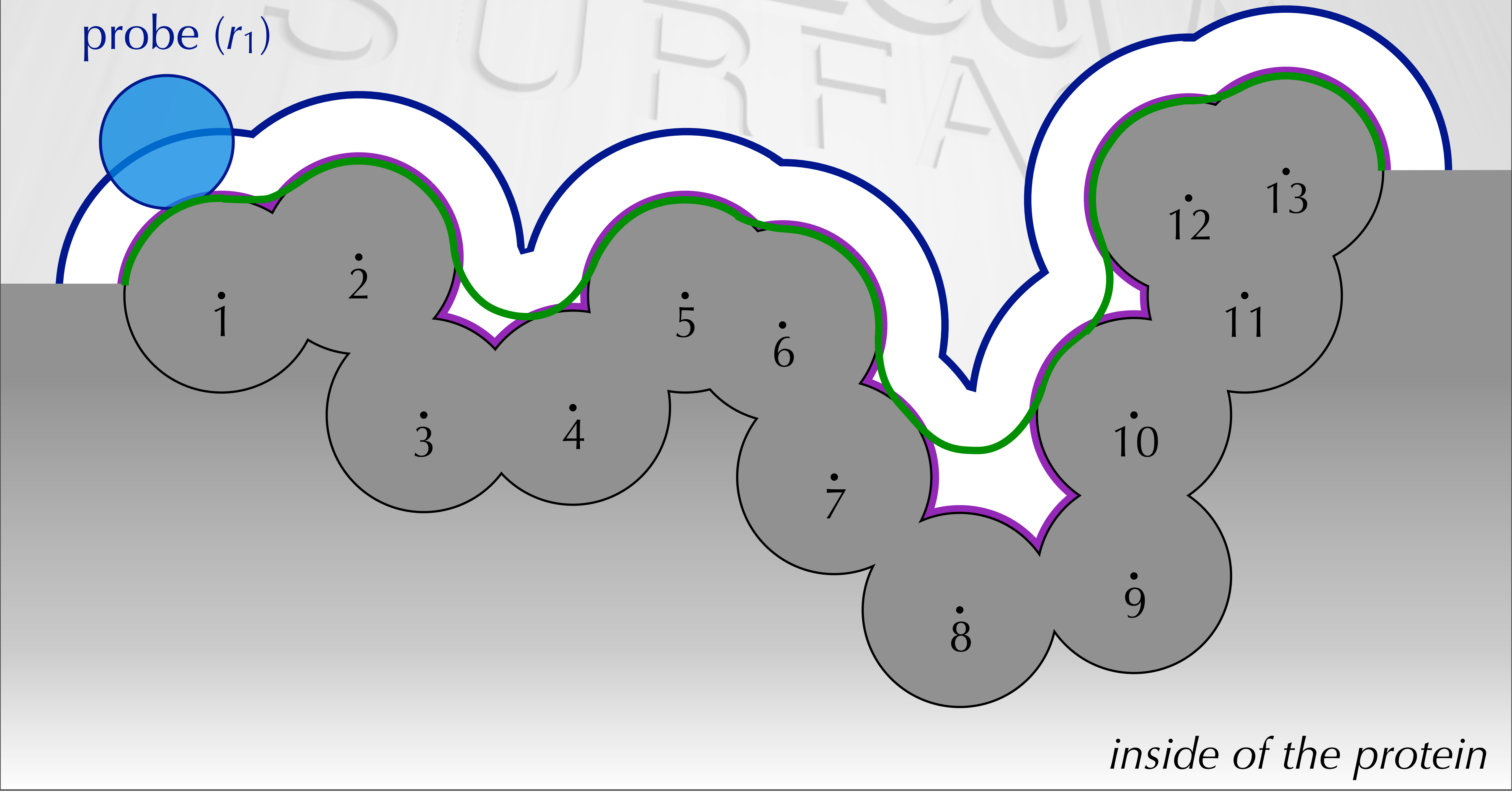
Definition of Molecular Surfaces

van der Waals surface

(solvent-)accessible surface

molecular surface

probe (r_1)



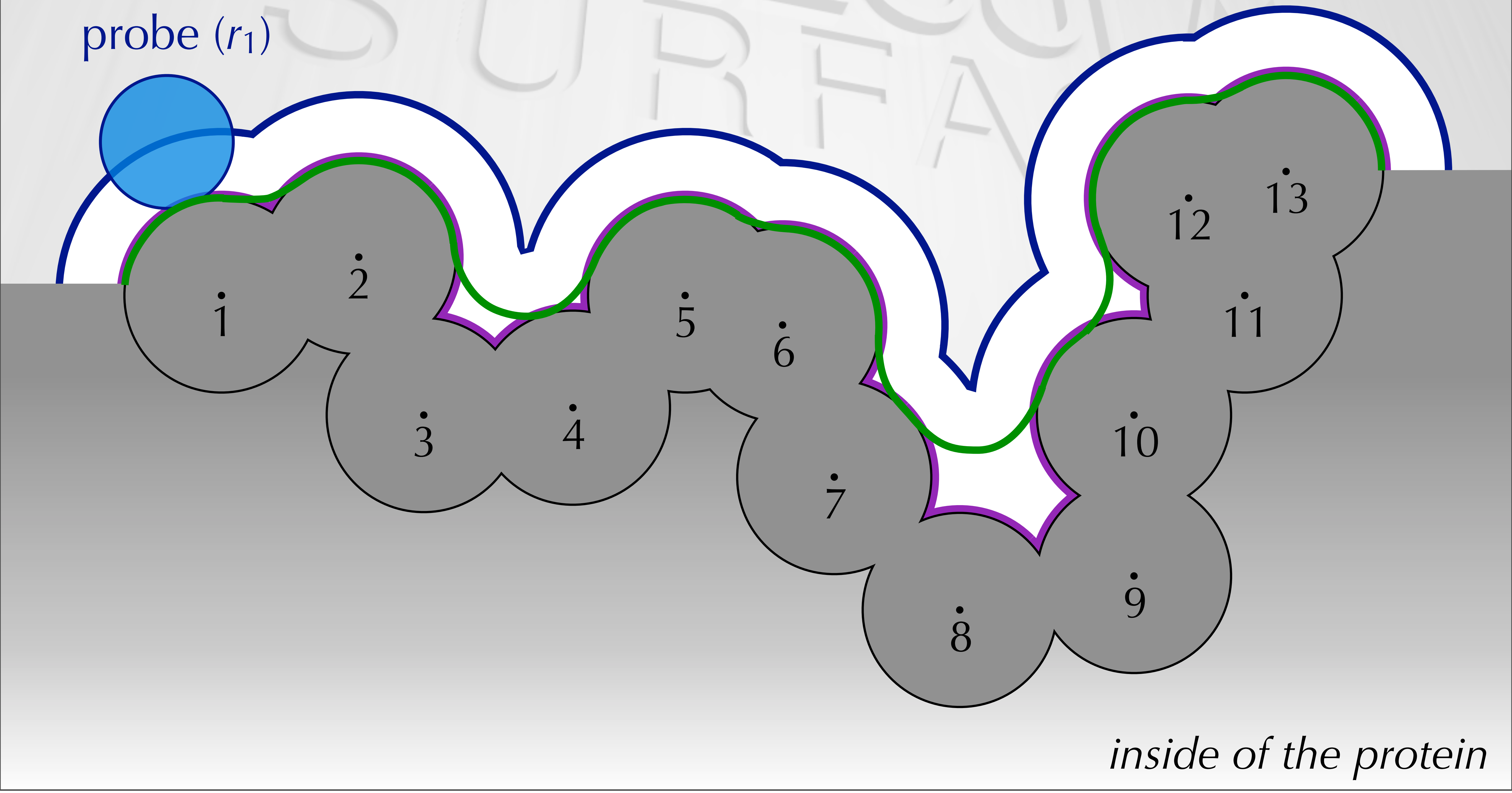
inside of the protein

Definition of Molecular Surfaces

van der Waals surface

(solvent-)accessible surface

molecular surface



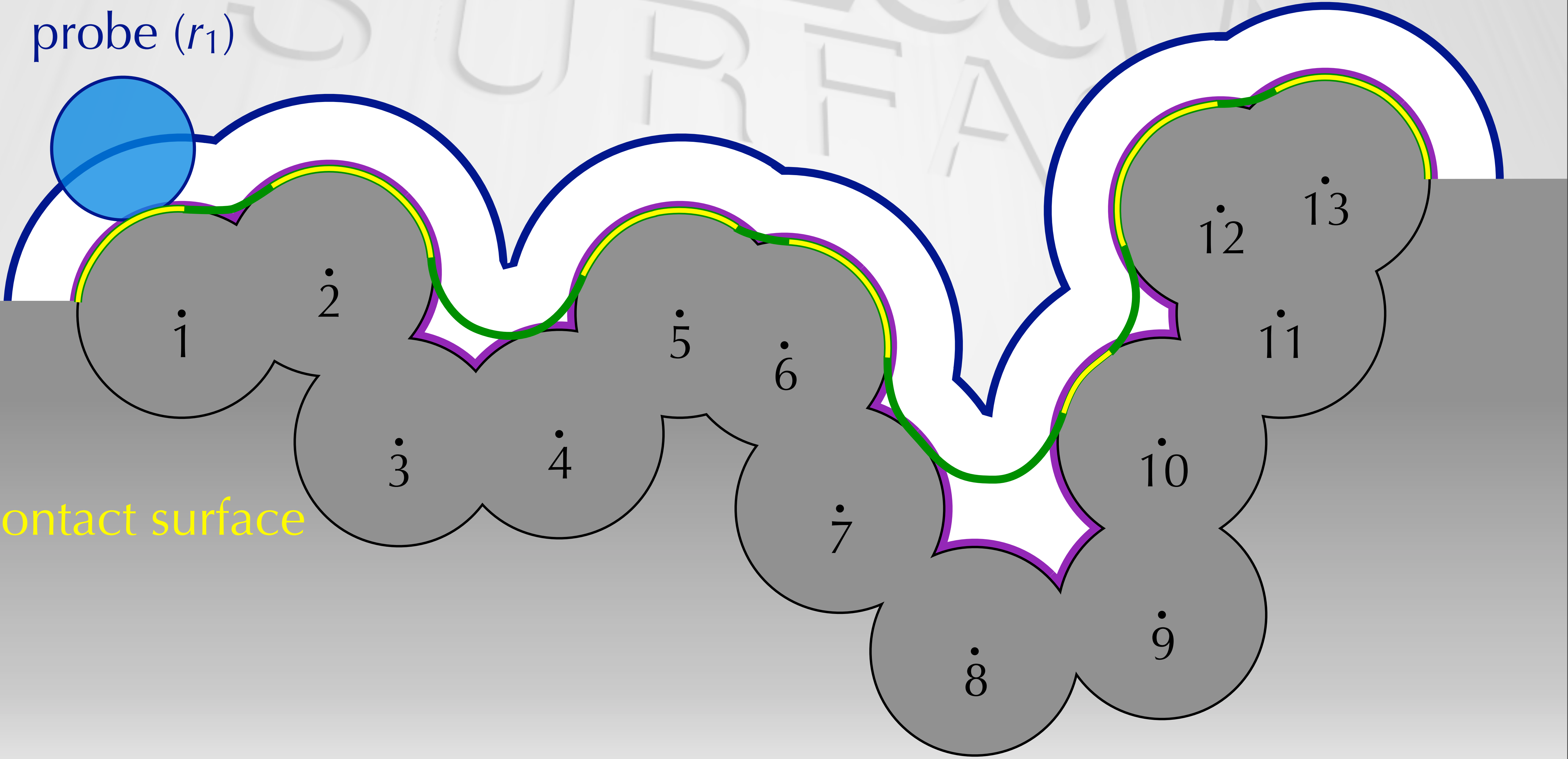
Definition of Molecular Surfaces

van der Waals surface

(solvent-)accessible surface

molecular surface

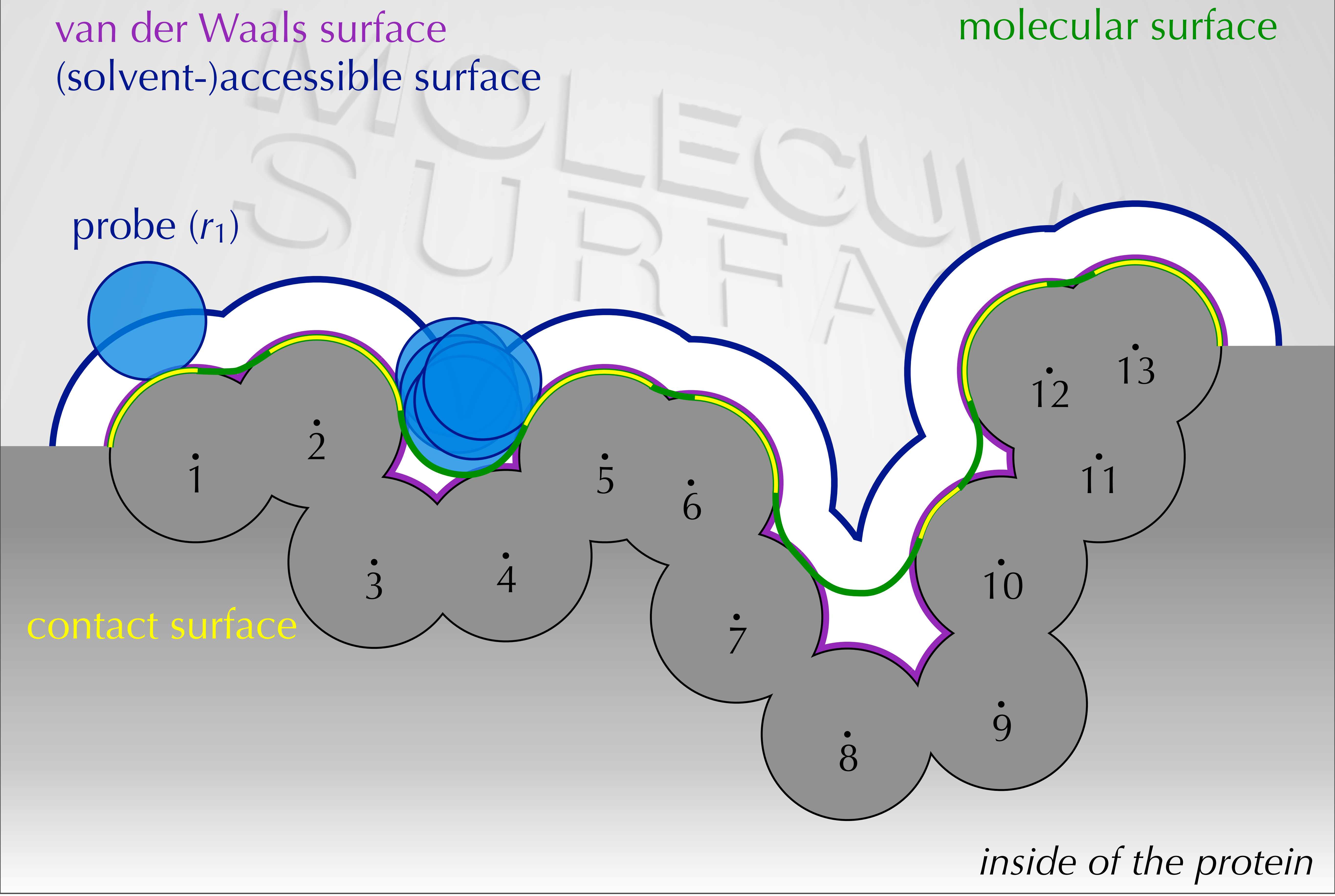
probe (r_1)



contact surface

inside of the protein

Definition of Molecular Surfaces



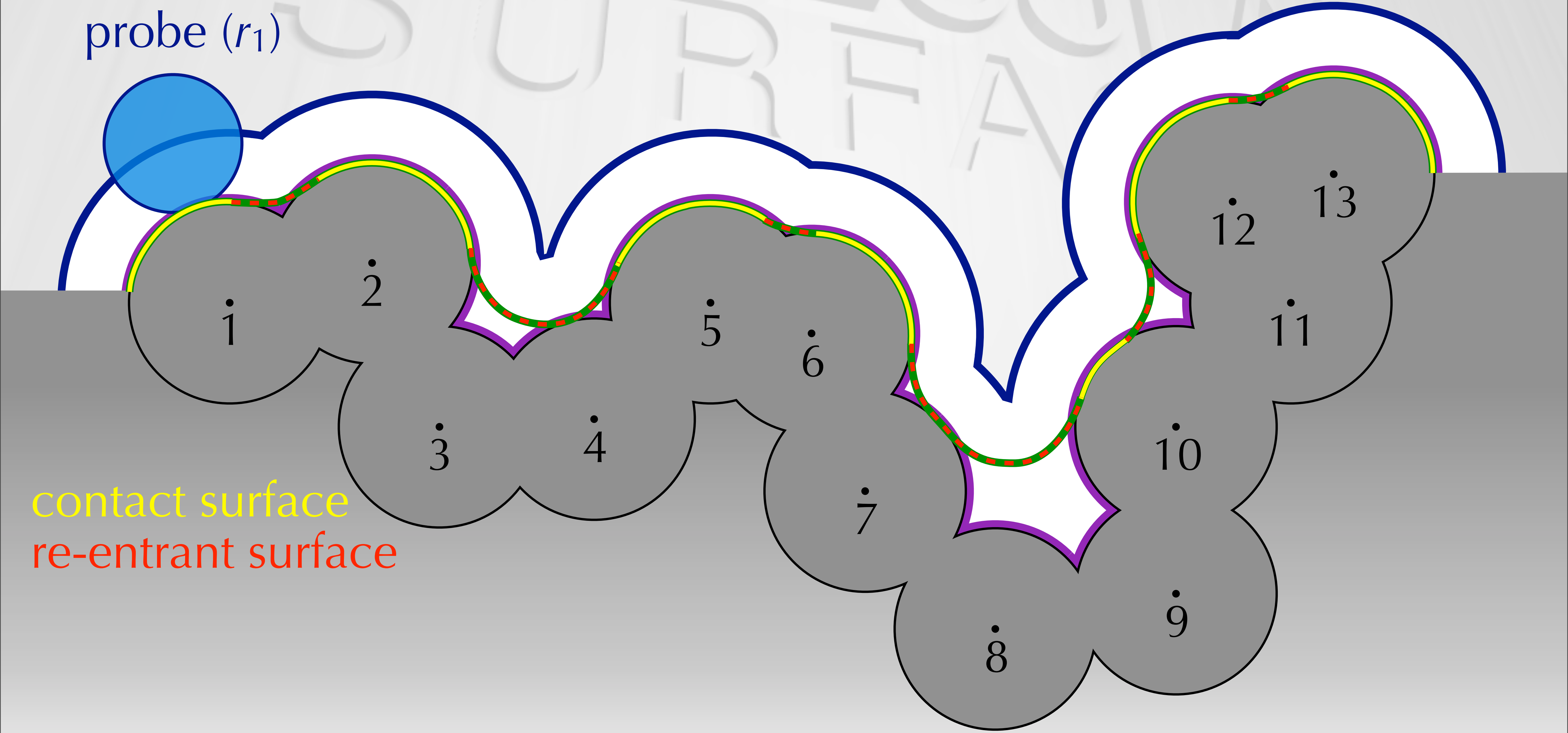
Definition of Molecular Surfaces

van der Waals surface

(solvent-)accessible surface

molecular surface

probe (r_1)



contact surface

re-entrant surface

inside of the protein

Definition of Molecular Surfaces

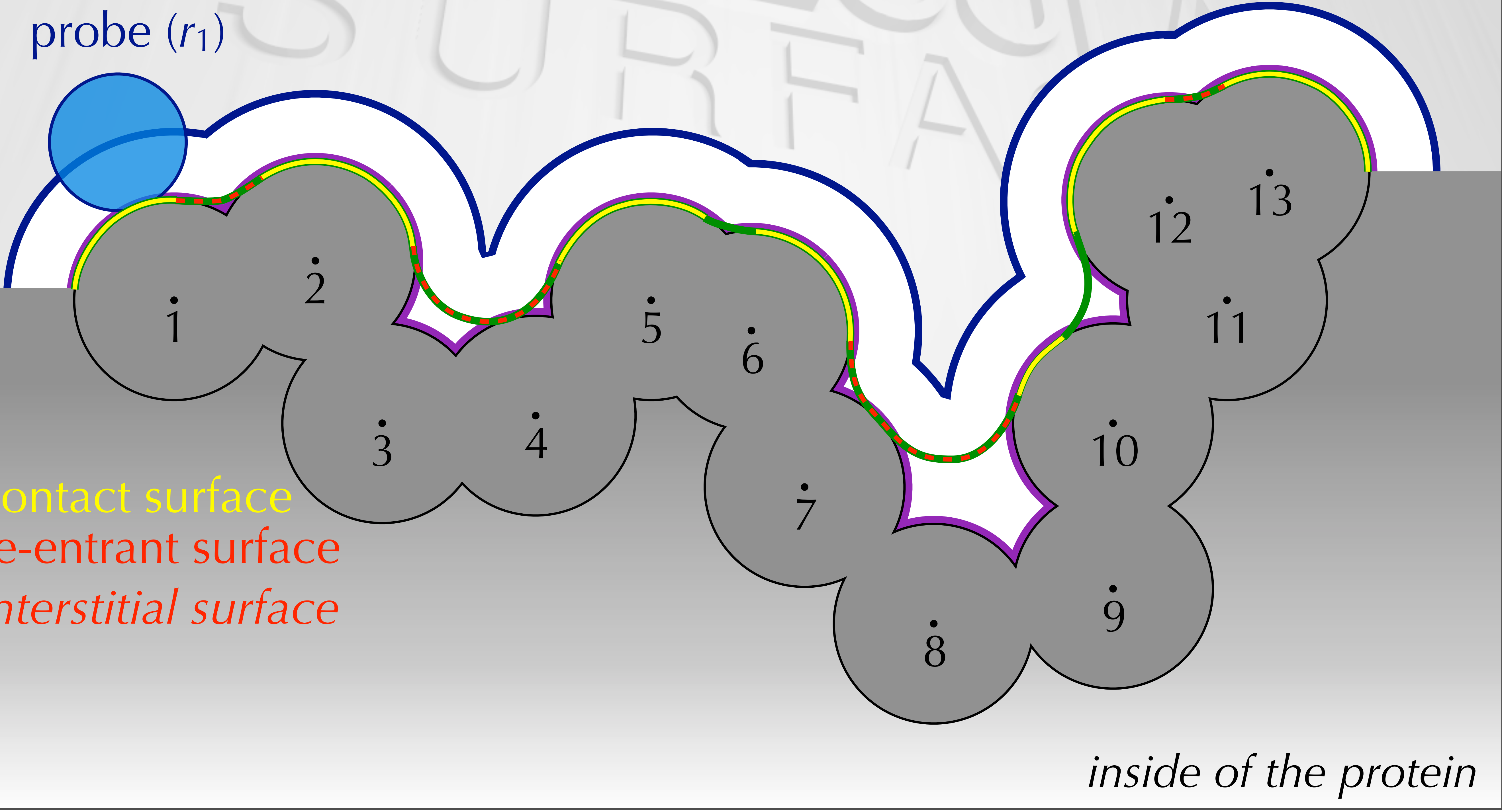
van der Waals surface

(solvent-)accessible surface

Lee-Richards surface

molecular surface

solvent-excluded/Connolly surface



contact surface

re-entrant surface

interstitial surface

inside of the protein

A Few Comments

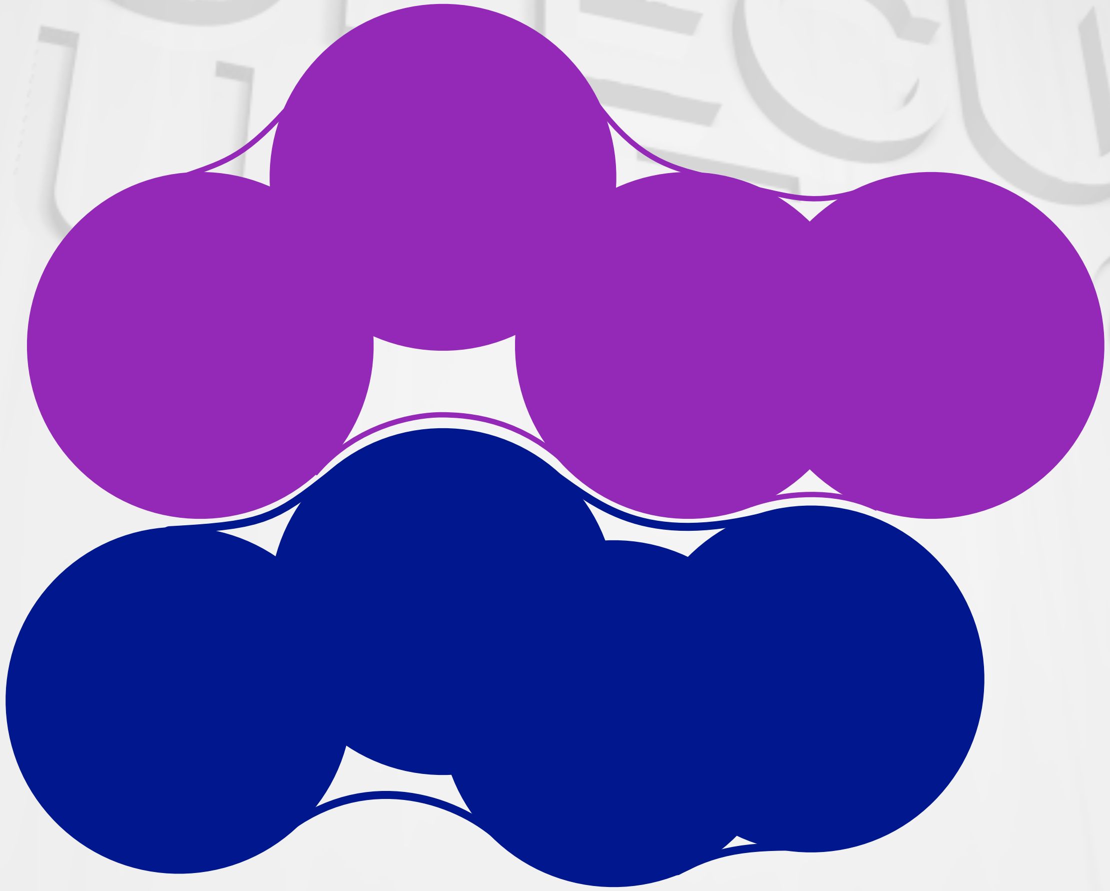
- All these surfaces rely on the van der Waals surface, which cannot be accurately determined! (diffuse distribution of the electron density surrounding the centre of each atom). For each atom, a van der Waals radius needs to be taken.

- Hydrogens are not always taken into account (eg X-ray structures): united-atom models instead for C, N, O and S to include the presence of 1–3 protons.

- The ratio contact surface to re-entrant surface can be a measure of the molecular surface roughness (rugosity).

A Few Comments

- Connolly surfaces are complementary at the interface between two molecules (eg ligand/binding pocket).



- Same definitions apply for the related volumes: van der Waals volume, solvent-accessible volume, solvent-excluded volume, interstitial volume.

Calculation of the Surfaces

- Two approaches to calculate surfaces: numerically or analytically.

- Two sets of variables always need to be defined first (besides the atomic coordinates):
  - the van der Waals radii
  - the probe radius (for water, r_p=1.4–1.5 Å)

- The solvent molecule is approximated by a sphere: for small molecules (eg water, acetone, urea, MeCN, DMSO, PhH and cHxH) it may be OK but for larger (linear) molecules it may be questionable (eg HxH, OcH, OcOH).

- Various programs and algorithms: MS, MSMS, Access, NAccess, Molecular Surface, Surf *etc*.

THE PROGRAMS YOU MAY USE *(NON EXHAUSTIVE)*

Simulation packages

- GROMACS: `g_sas` ➙ calculates SASA
  method: numerical (Double Lattice Cubic Method, DCLM)

- GROMOS (GROMOS++): `sasa` ➙ calculates SASA
  method: numerical (Lee–Richards)

Visualisation softwares

- VMD: `measure sasa` *`probe_radius`* `$group_A [-restrict $group_B]` (B⊂A) ➙ calculates SASA
  method: numerical(?) (SURF) (can also use MSMS)

- PyMOL: by default, PyMOL *shows* the Connolly surface but *calculates* the SASA.
  method: numerical(?)

The Programs You May Use *(Non Exhaustive)*

Specific programs

- MSMS: by default, uses extended atoms (otherwise use `pdb_to_xyzr -h`) → calculates SASA/Connolly surface
  method: analytical
  http://mgltools.scripps.edu/packages/MSMS

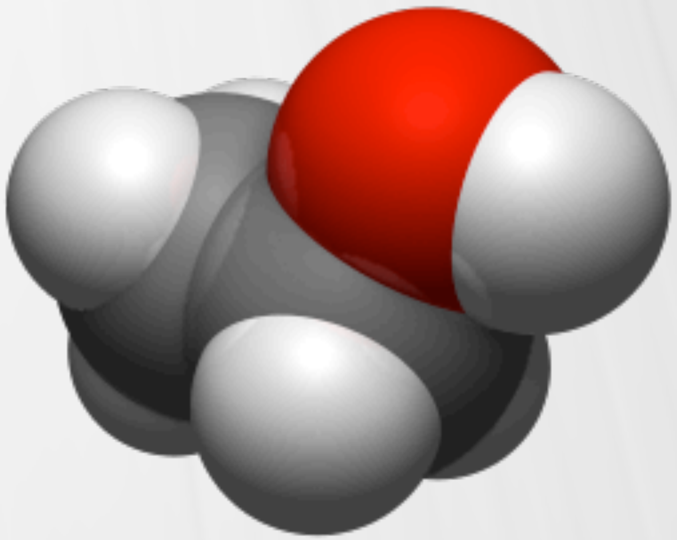
- NAccess: by default ignores protons (otherwise `naccess -h`)
  → calculates SASA
  method: numerical (Lee–Richards)
  http://www.bioinf.manchester.ac.uk/naccess

Many (dedicated or general) programs and packages can calculate molecular surfaces, generally only SASAs. The source code may be available.

THE ROLLING-BALL ALGORITHM (GROMOS, NACCESS)

Historically, the first method (Lee, B. & Richards, F. M. (1971) *J. Mol. Biol.* 55, 379–400).

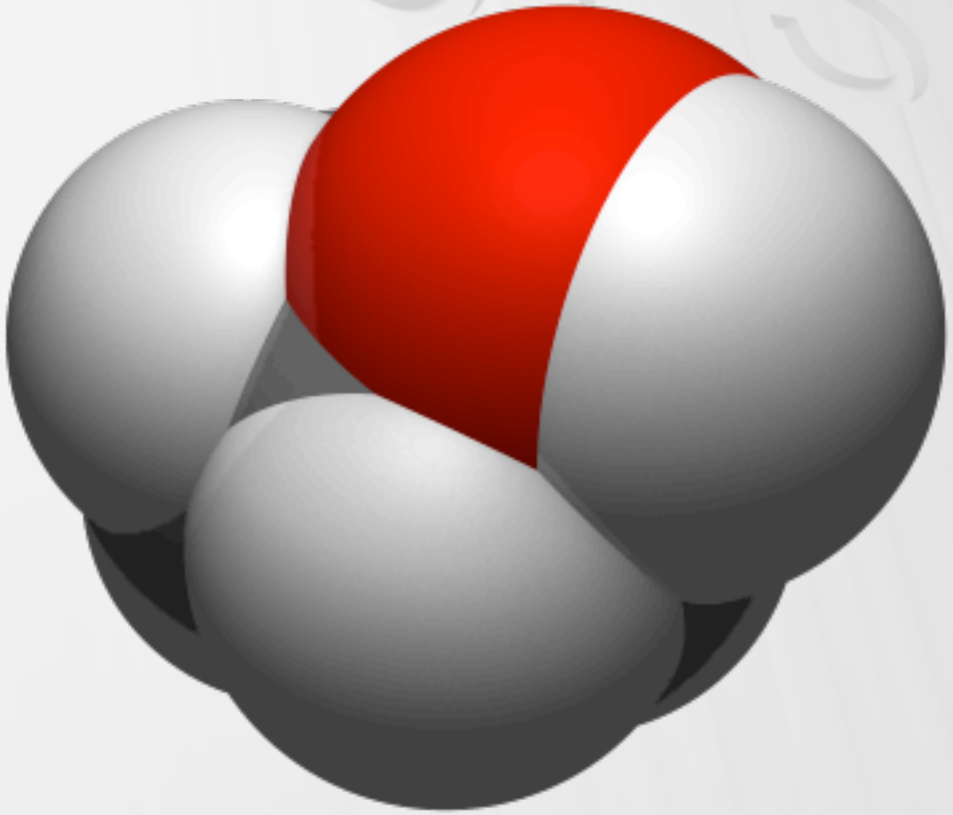
1. To mimic the effect of a water molecule rolling on the surface of the solute, the van der Waals surface is expanded by the radius of a water sphere (1.4–1.5 Å).



THE ROLLING-BALL ALGORITHM (GROMOS, NACCESS)

Historically, the first method (Lee, B. & Richards, F. M. (1971) *J. Mol. Biol.* 55, 379–400).

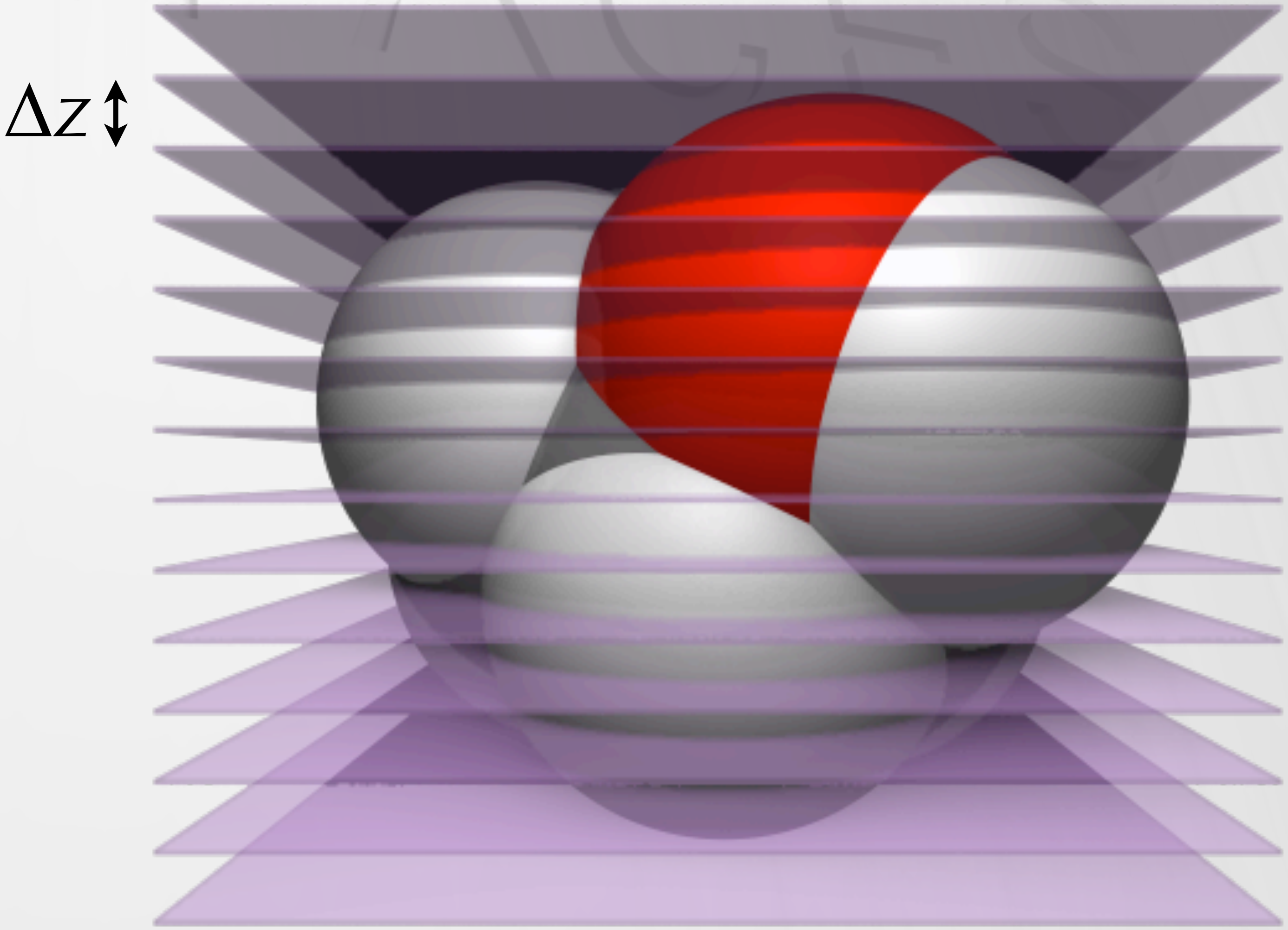
1. To mimic the effect of a water molecule rolling on the surface of the solute, the van der Waals surface is expanded by the radius of a water sphere (1.4–1.5 Å).



The Rolling-Ball Algorithm (Gromos, NAccess)

Historically, the first method (Lee, B. & Richards, F. M. (1971) *J. Mol. Biol.* 55, 379–400).

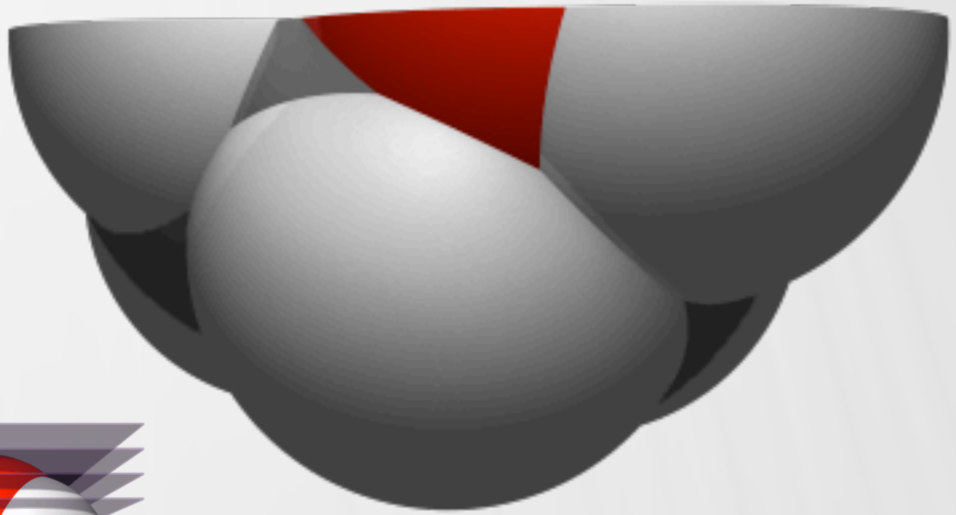
1. To mimic the effect of a water molecule rolling on the surface of the solute, the van der Waals surface is expanded by the radius of a water sphere (1.4–1.5 Å).
2. The expanded molecule is divided into several sections along *z* (spacing Δ*z*).

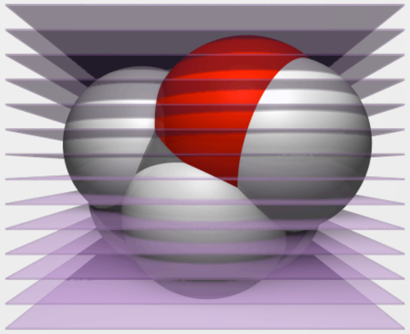


The Rolling-Ball Algorithm (Gromos, NAccess)

Historically, the first method (Lee, B. & Richards, F. M. (1971) *J. Mol. Biol.* 55, 379–400).

1. To mimic the effect of a water molecule rolling on the surface of the solute, the van der Waals surface is expanded by the radius of a water sphere (1.4–1.5 Å).
2. The expanded molecule is divided into several sections along *z* (spacing Δ*z*).

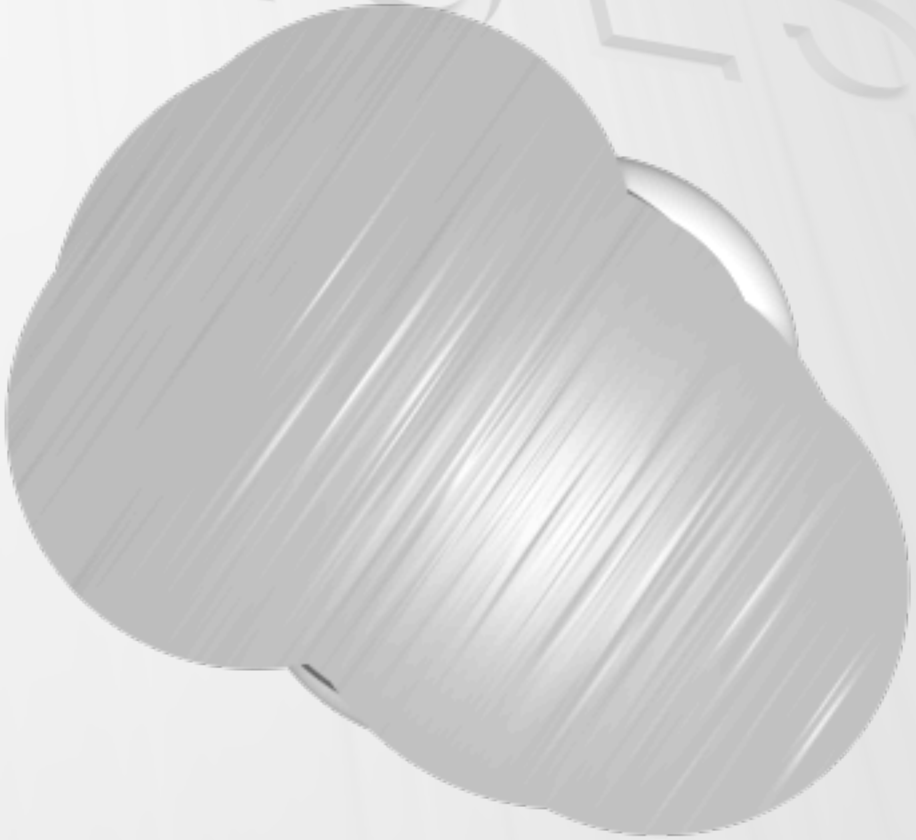


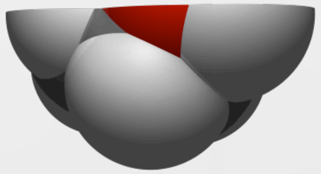


THE ROLLING-BALL ALGORITHM (GROMOS, NACCESS)

Historically, the first method (Lee, B. & Richards, F. M. (1971) *J. Mol. Biol.* 55, 379–400).

1. To mimic the effect of a water molecule rolling on the surface of the solute, the van der Waals surface is expanded by the radius of a water sphere (1.4–1.5 Å).
2. The expanded molecule is divided into several sections along *z* (spacing Δ*z*).
3. Each arc of length L_i of a circle that is not interior to any other atom circles in the same *z*-section is defined as solvent-accessible.

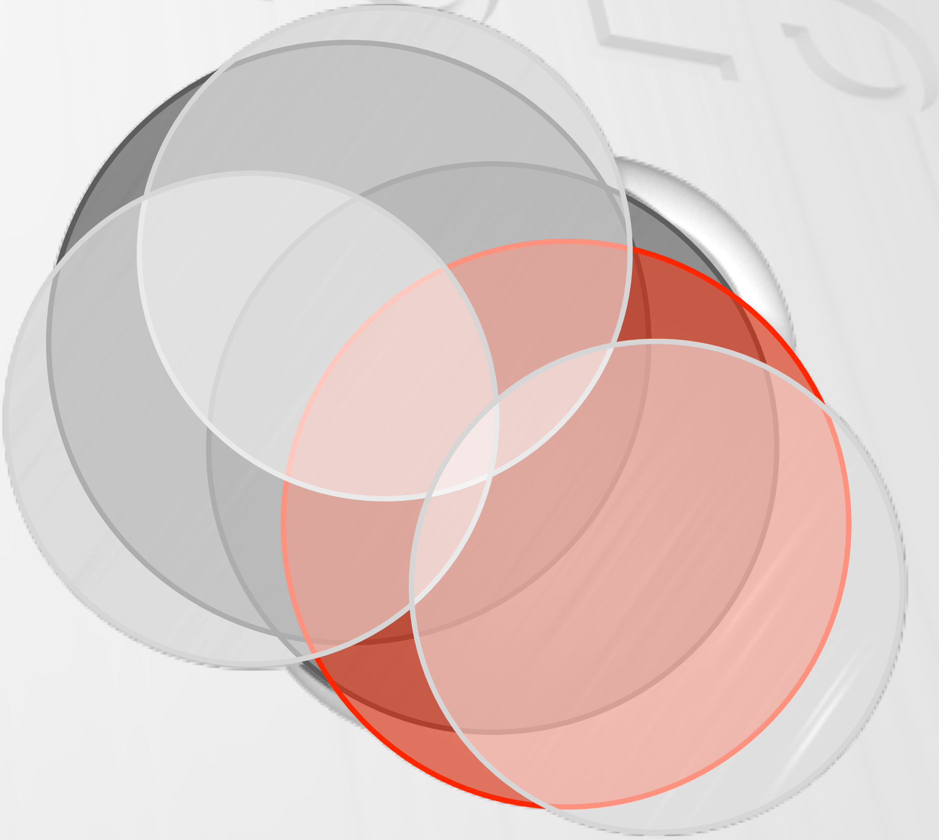


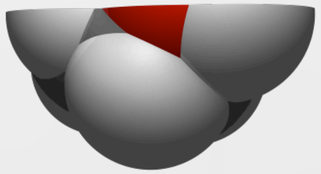


THE ROLLING-BALL ALGORITHM (GROMOS, NACCESS)

Historically, the first method (Lee, B. & Richards, F. M. (1971) *J. Mol. Biol.* 55, 379–400).

1. To mimic the effect of a water molecule rolling on the surface of the solute, the van der Waals surface is expanded by the radius of a water sphere (1.4–1.5 Å).
2. The expanded molecule is divided into several sections along *z* (spacing Δ*z*).
3. Each arc of length L_i of a circle that is not interior to any other atom circles in the same *z*-section is defined as solvent-accessible.



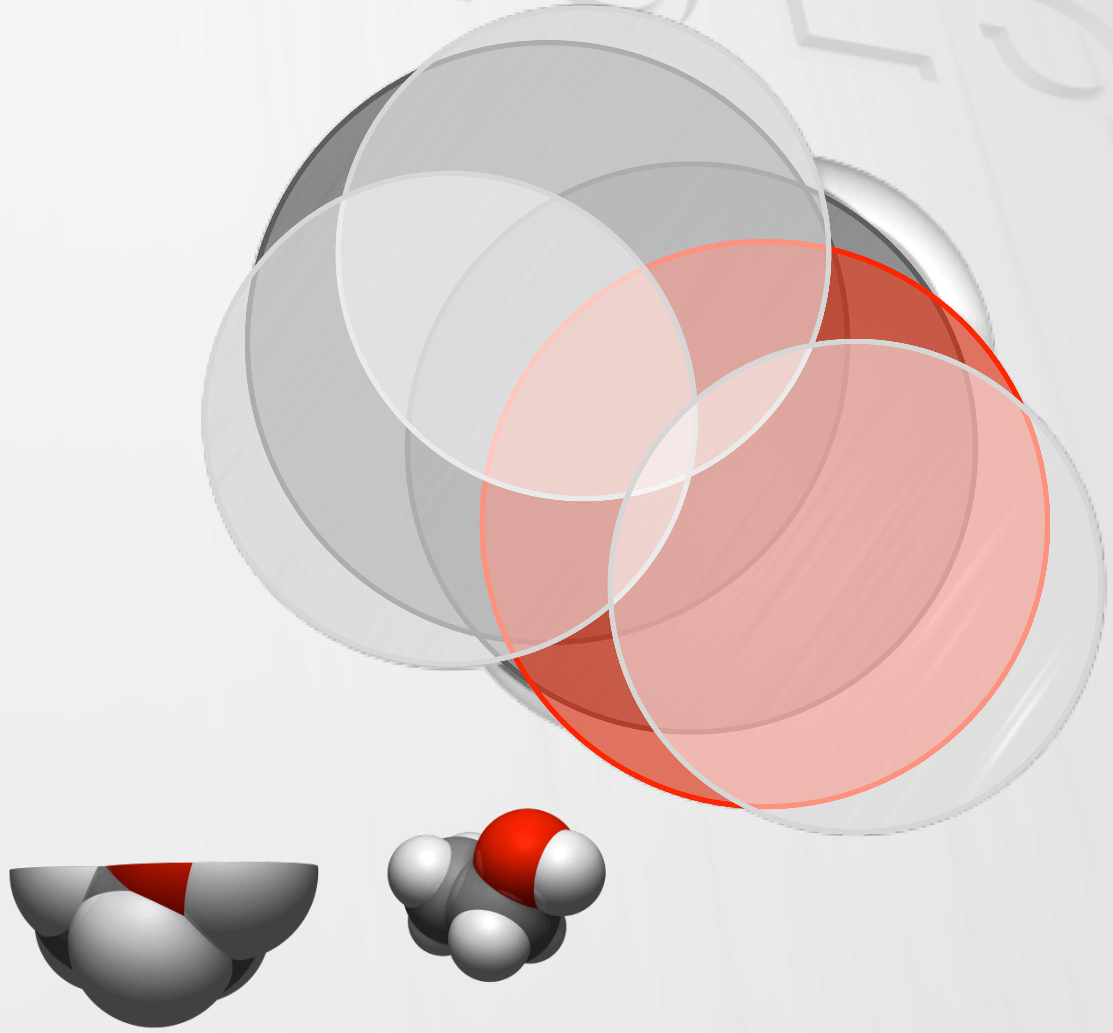


The Rolling-Ball Algorithm (Gromos, NAccess)

Historically, the first method (Lee, B. & Richards, F. M. (1971) *J. Mol. Biol.* 55, 379–400).

1. To mimic the effect of a water molecule rolling on the surface of the solute, the van der Waals surface is expanded by the radius of a water sphere (1.4–1.5 Å).
2. The expanded molecule is divided into several sections along *z* (spacing Δ*z*)
3. Each arc of length L_i of a circle that is not interior to any other atom circles in the same *z*-section is defined as solvent-accessible.
4. The total SASA of a given atom *j* is calculated:

$$A_j = \sum_i L_{i,j} \Delta z$$

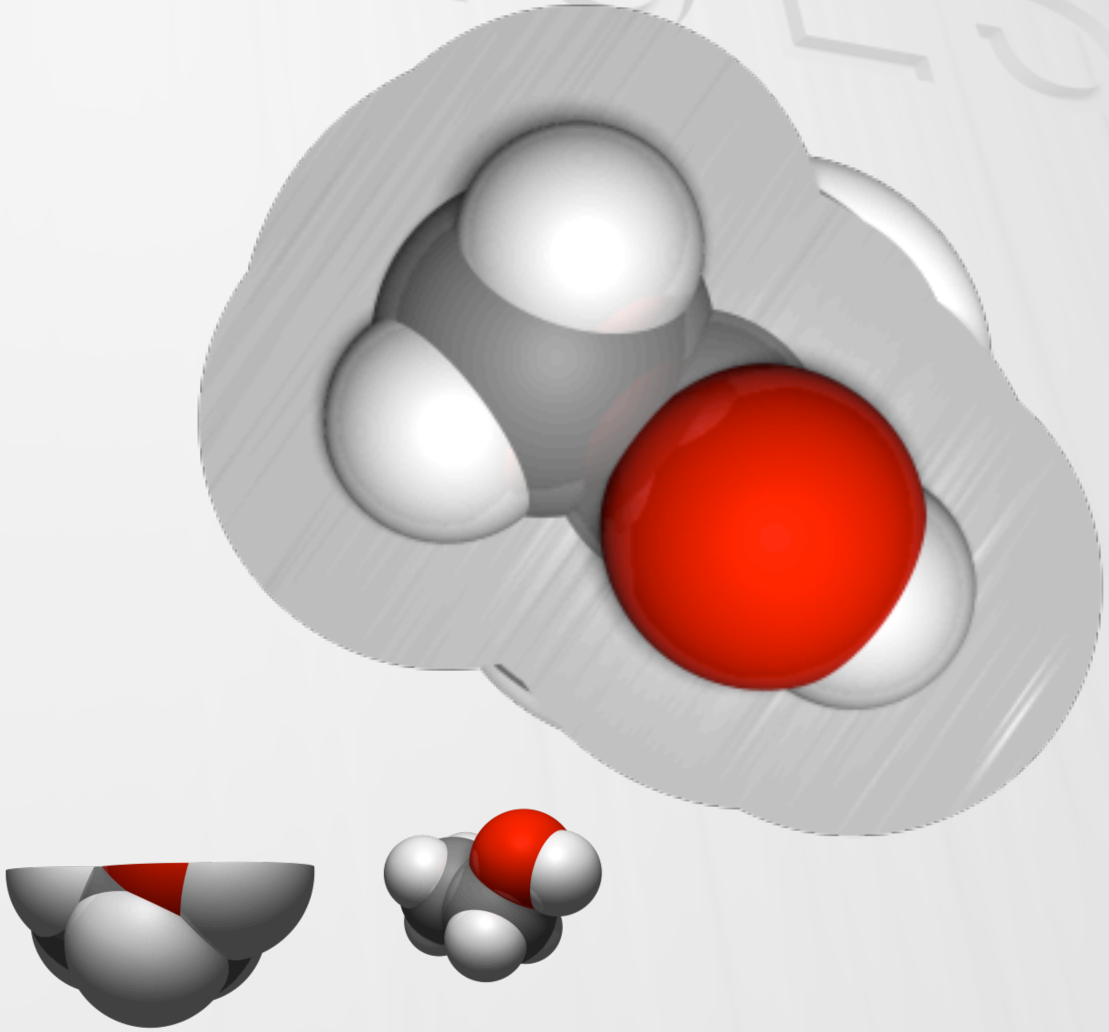


The Rolling-Ball Algorithm (Gromos, NAccess)

Historically, the first method (Lee, B. & Richards, F. M. (1971) *J. Mol. Biol.* 55, 379–400).

1. To mimic the effect of a water molecule rolling on the surface of the solute, the van der Waals surface is expanded by the radius of a water sphere (1.4–1.5 Å).
2. The expanded molecule is divided into several sections along z (spacing Δz)
3. Each arc of length L_i of a circle that is not interior to any other atom circles in the same z-section is defined as solvent-accessible.
4. The total SASA of a given atom j is calculated:

$$A_j = \sum_i L_{i,j}\,\Delta z$$

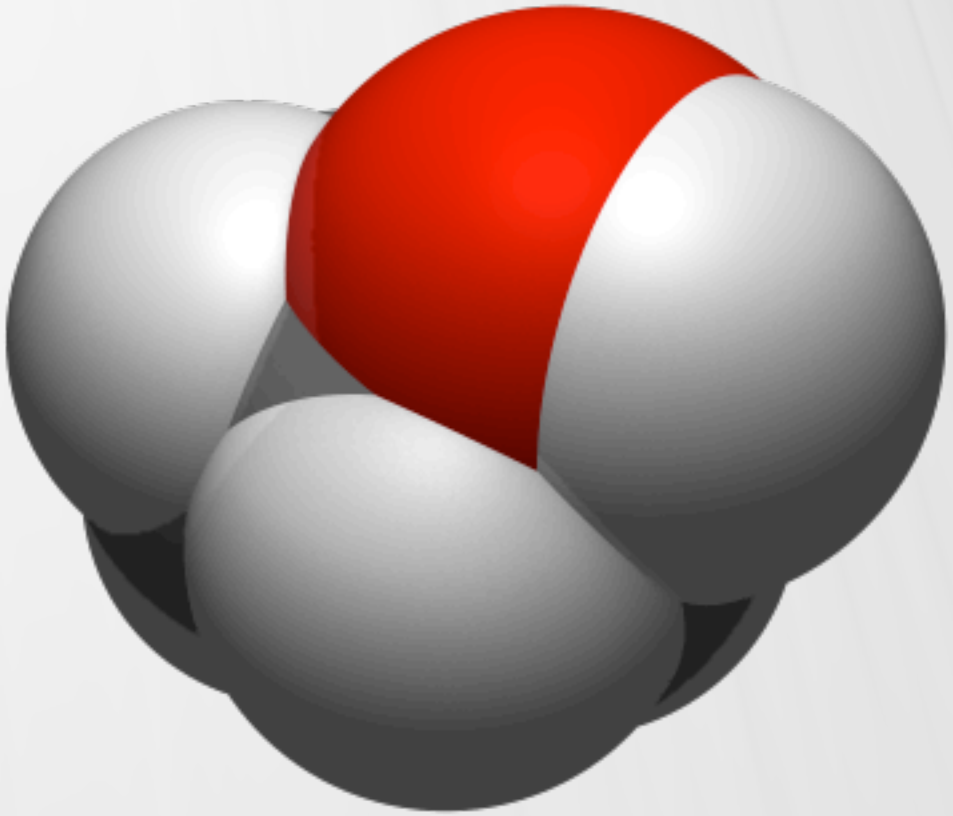


The Rolling-Ball Algorithm (Variations)

Instead of calculating circle arc lengths, a grid-based approach is used.

- Shrake-Rupley algorithm (Shrake, A. & Rupley, J. A. (1973) *J. Mol. Biol.* 79, 351–371):

1. A uniformly distributed mesh of (92) test points $\{t_i\}$ equidistant from each hydrated (extended) atom (the *central atom*) is drawn.

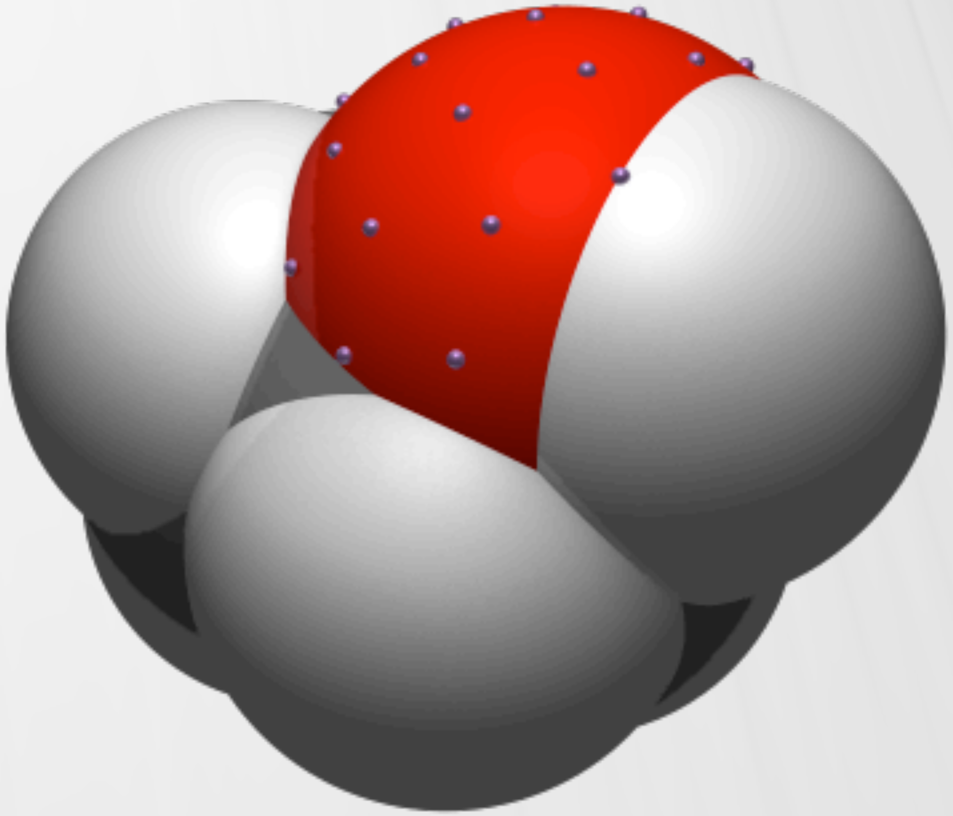


The Rolling-Ball Algorithm (Variations)

Instead of calculating circle arc lengths, a grid-based approach is used.

- Shrake-Rupley algorithm (Shrake, A. & Rupley, J. A. (1973) *J. Mol. Biol.* 79, 351–371):

1. A uniformly distributed mesh of (92) test points $\{t_i\}$ equidistant from each hydrated (extended) atom (the *central atom*) is drawn.
2. Each test point is then defined as occluded (buried) by a neighbouring atom (a *test atom* of centre A and hydrated radius r) if $r > d(A,t_i)$.

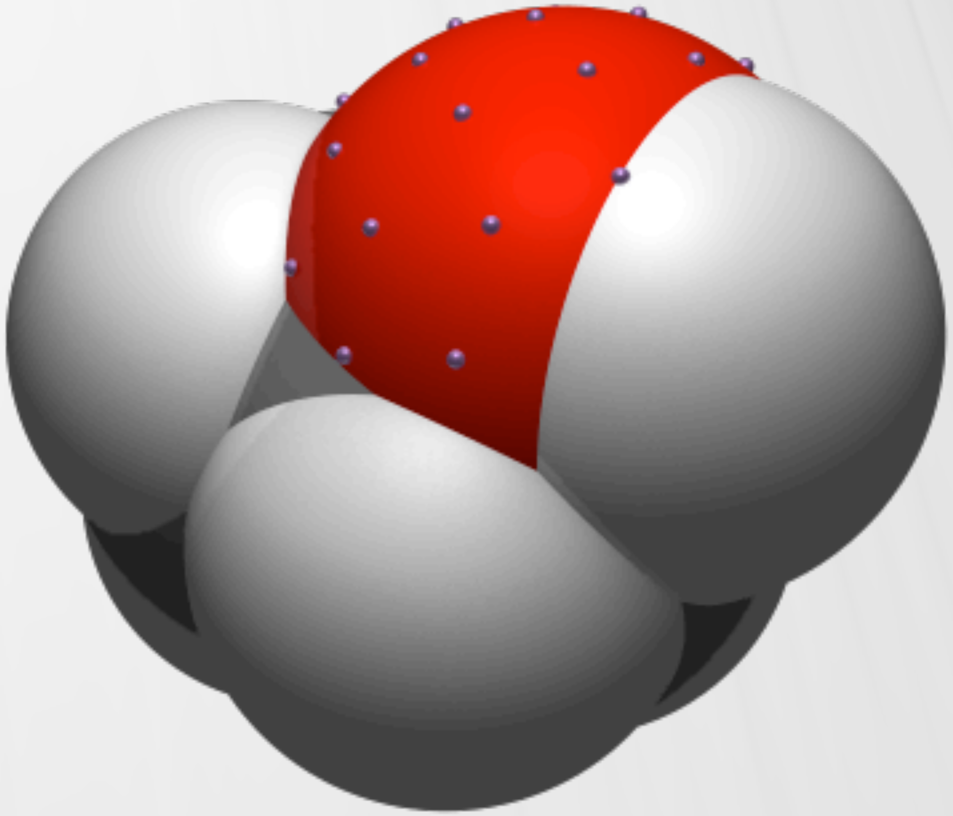


The Rolling-Ball Algorithm (Variations)

Instead of calculating circle arc lengths, a grid-based approach is used.

- Shrake-Rupley algorithm (Shrake, A. & Rupley, J. A. (1973) *J. Mol. Biol.* 79, 351–371):

1. A uniformly distributed mesh of (92) test points $\{t_i\}$ equidistant from each hydrated (extended) atom (the *central atom*) is drawn.
2. Each test point is then defined as occluded (buried) by a neighbouring atom (a *test atom* of centre A and hydrated radius r) if $r > d(A,t_i)$.
3. For each central atom, the SASA is calculated by multiplying the number of solvent-accessible test points by the surface area value corresponding to each test point.

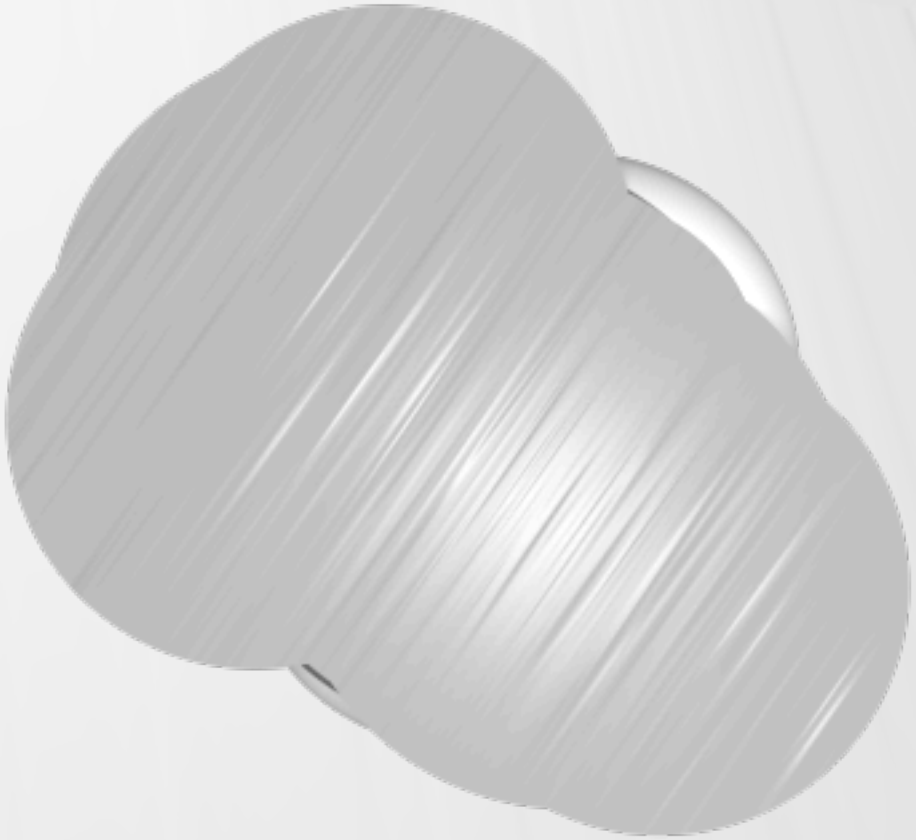


The Rolling-Ball Algorithm (Variations)

Instead of calculating circle arc lengths, a grid-based approach is used.

- Shrake-Rupley algorithm (Shrake, A. & Rupley, J. A. (1973) *J. Mol. Biol.* 79, 351–371).

- The *z*-sections are discretised with a grid. Each cell is defined as *inside* the atoms, *outside* or *in-between*.

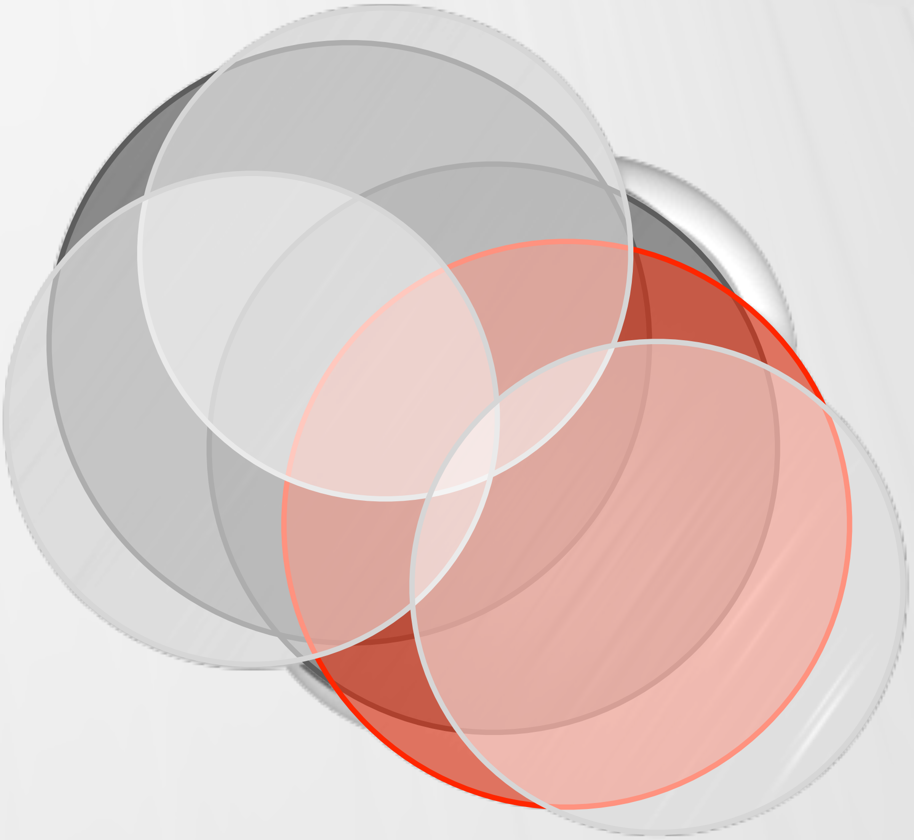


The Rolling-Ball Algorithm (Variations)

Instead of calculating circle arc lengths, a grid-based approach is used.

- Shrake-Rupley algorithm (Shrake, A. & Rupley, J. A. (1973) *J. Mol. Biol.* 79, 351–371).

- The *z*-sections are discretised with a grid. Each cell is defined as *inside* the atoms, *outside* or *in-between*.

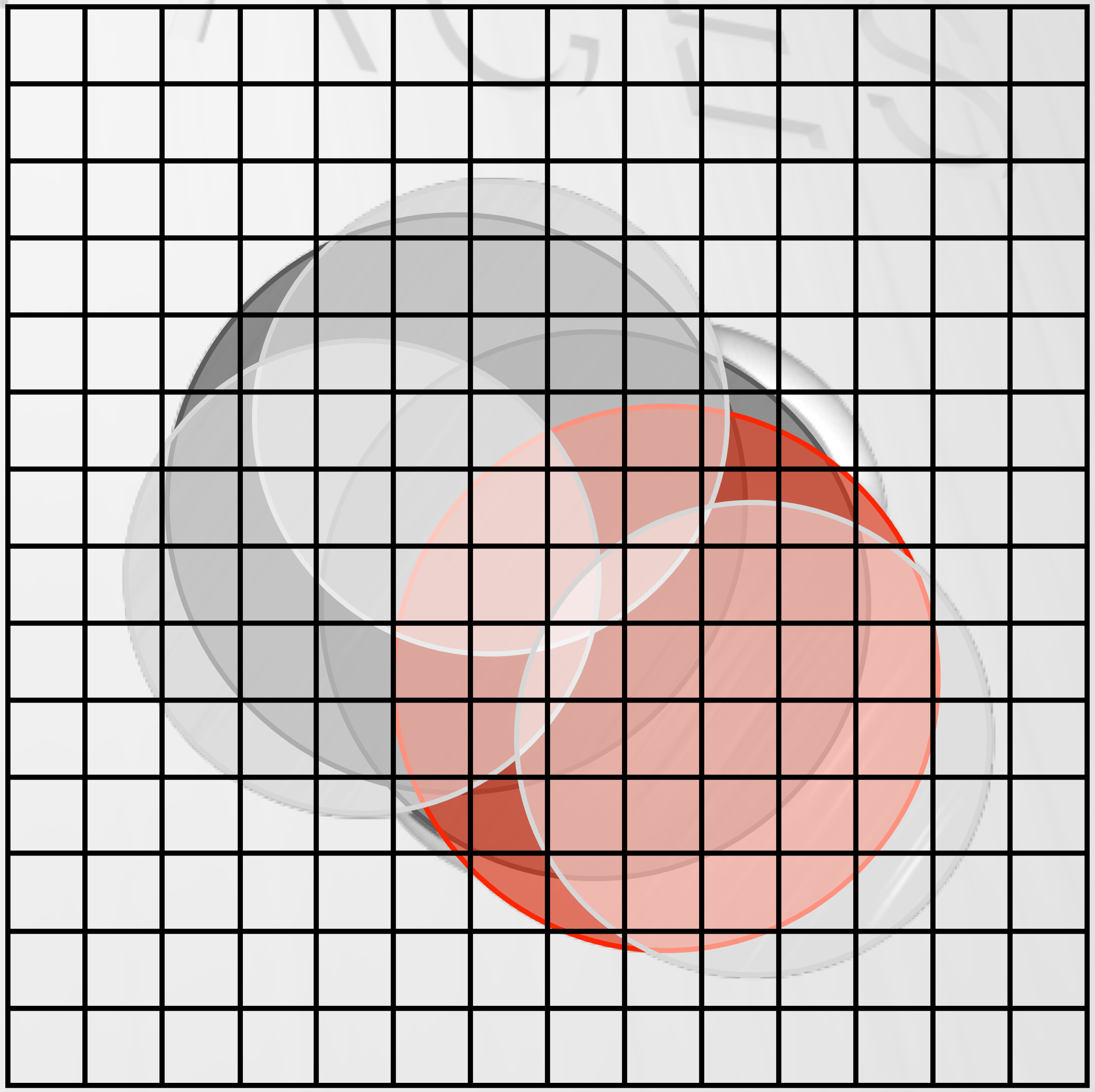


The Rolling-Ball Algorithm (Variations)

Instead of calculating circle arc lengths, a grid-based approach is used.

- Shrake-Rupley algorithm (Shrake, A. & Rupley, J. A. (1973) *J. Mol. Biol.* 79, 351–371).

- The *z*-sections are discretised with a grid. Each cell is defined as *inside* the atoms, *outside* or *in-between*.

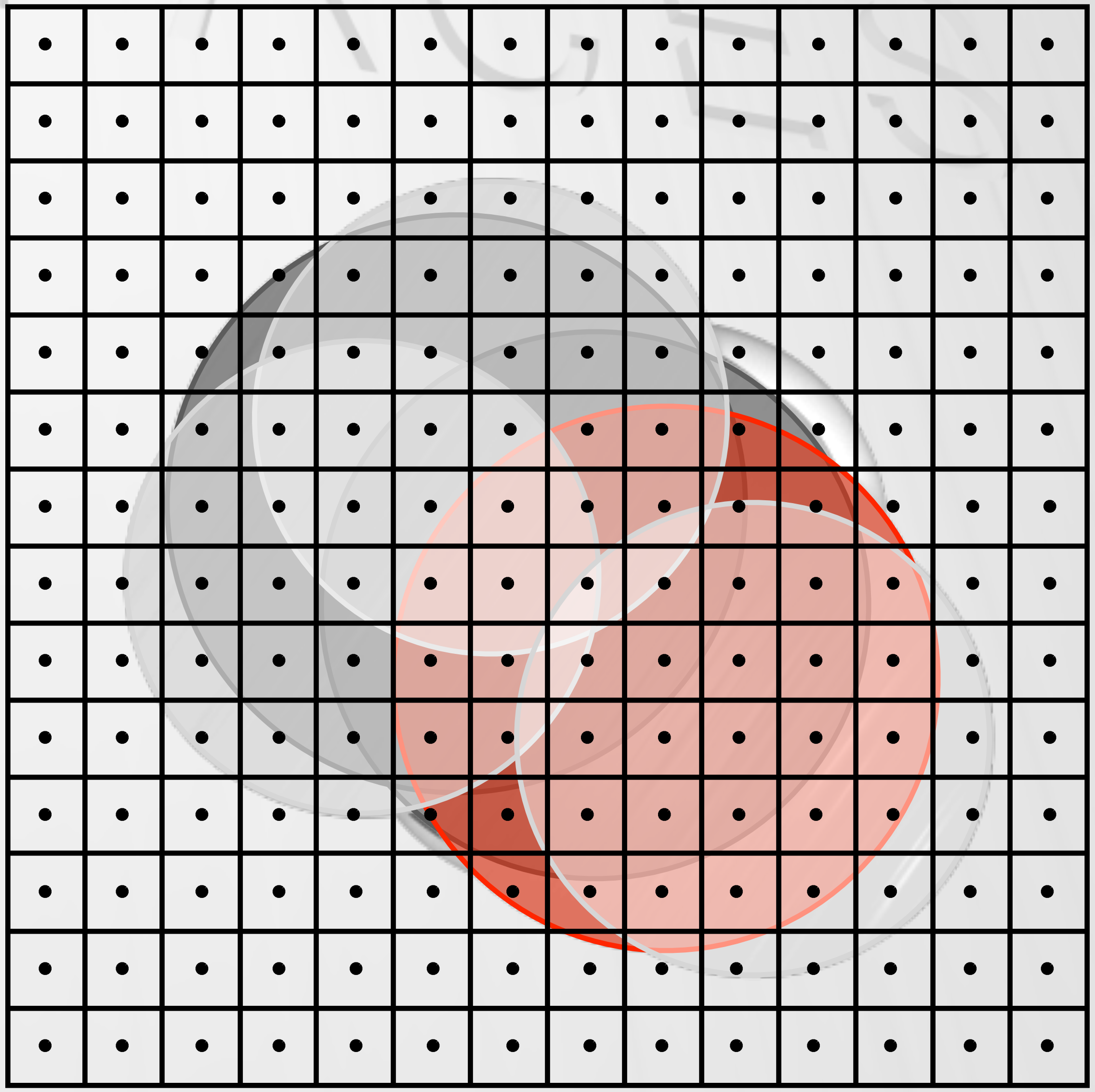


THE ROLLING-BALL ALGORITHM (VARIATIONS)

Instead of calculating circle arc lengths, a grid-based approach is used.

- Shrake-Rupley algorithm (Shrake, A. & Rupley, J. A. (1973) *J. Mol. Biol.* 79, 351–371).

- The *z*-sections are discretised with a grid. Each cell is defined as *inside* the atoms, *outside* or *in-between*.

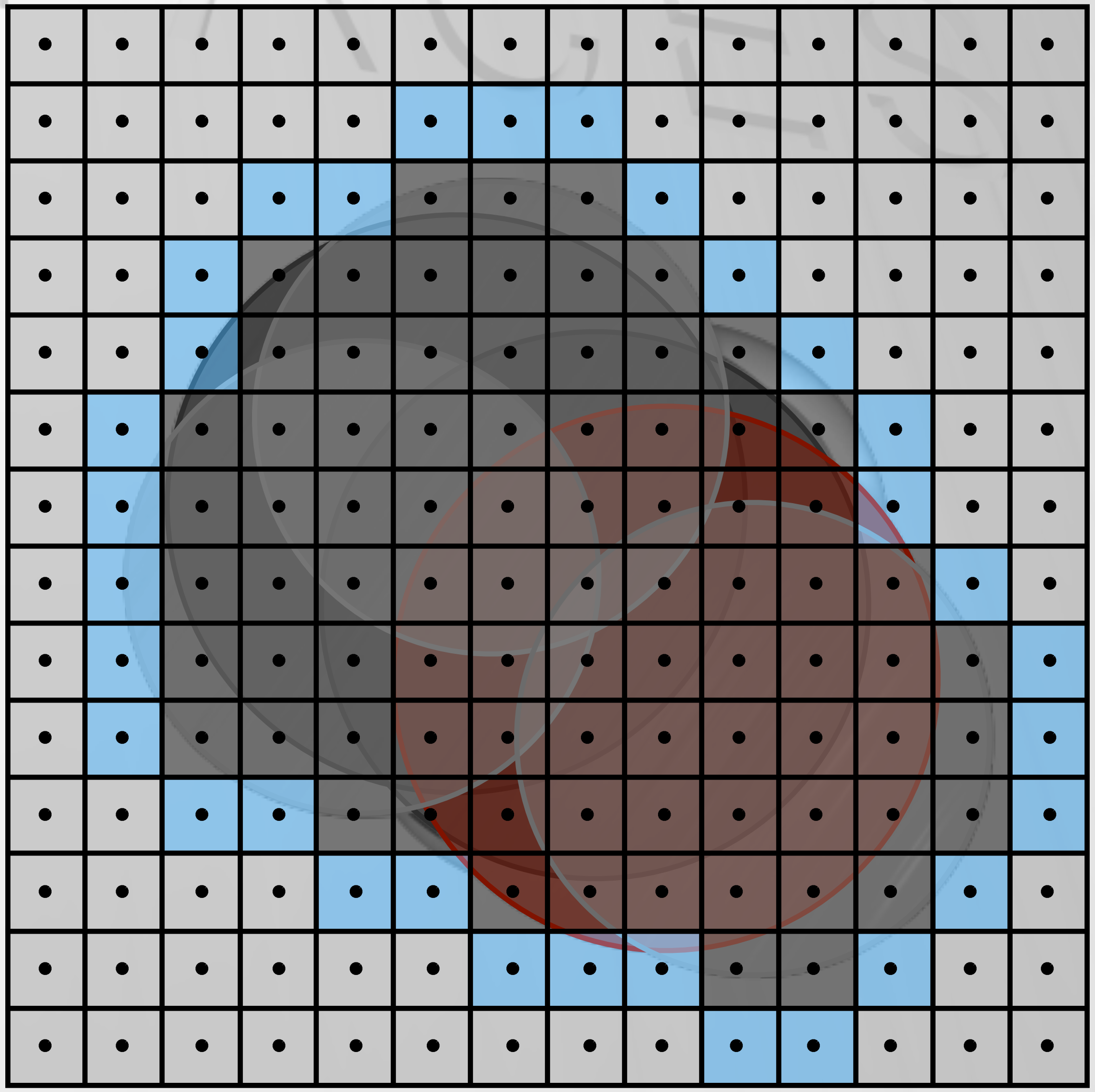


The Rolling-Ball Algorithm (Variations)

Instead of calculating circle arc lengths, a grid-based approach is used.

- Shrake-Rupley algorithm (Shrake, A. & Rupley, J. A. (1973) *J. Mol. Biol.* 79, 351–371).

- The *z*-sections are discretised with a grid. Each cell is defined as *inside* the atoms, *outside* or *in-between*.

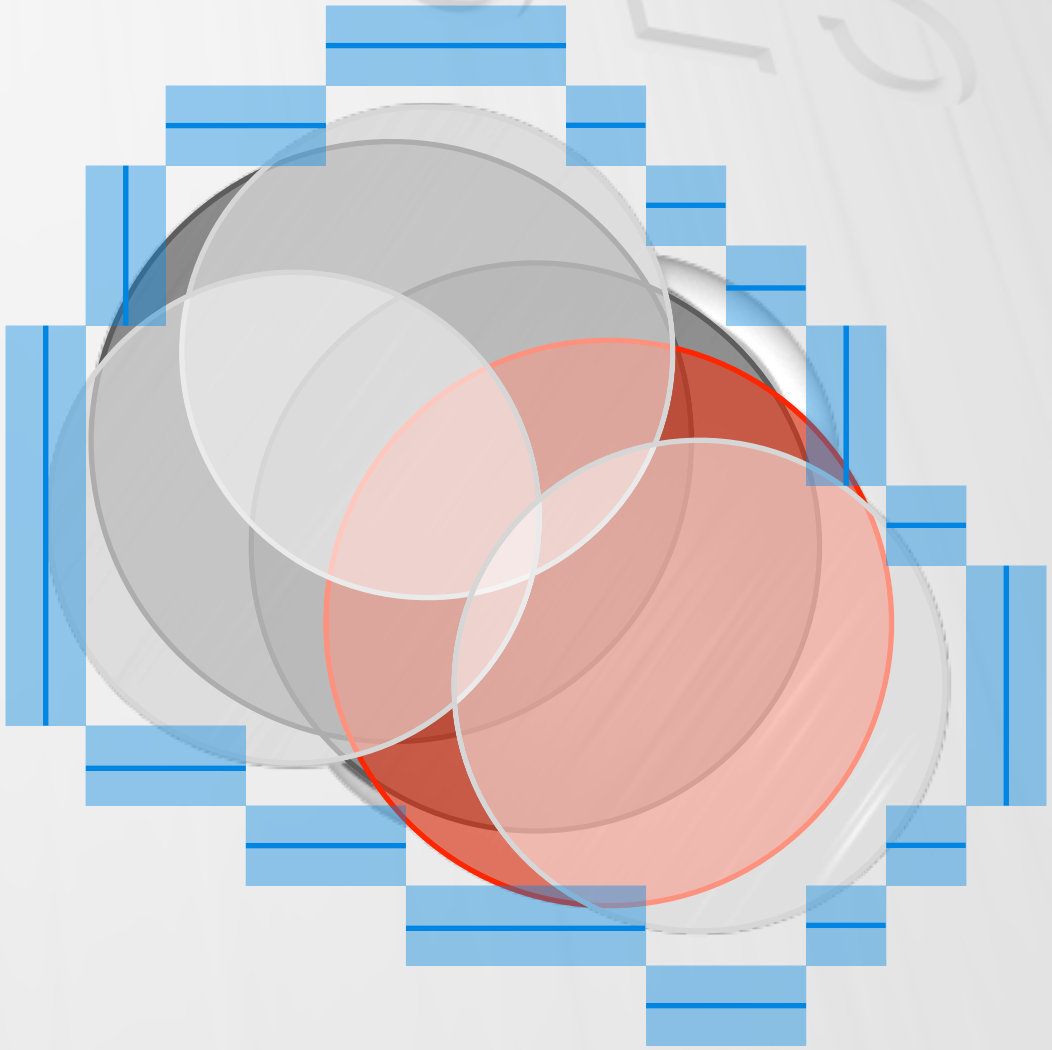


The Rolling-Ball Algorithm (Variations)

Instead of calculating circle arc lengths, a grid-based approach is used.

- Shrake-Rupley algorithm (Shrake, A. & Rupley, J. A. (1973) *J. Mol. Biol.* 79, 351–371).

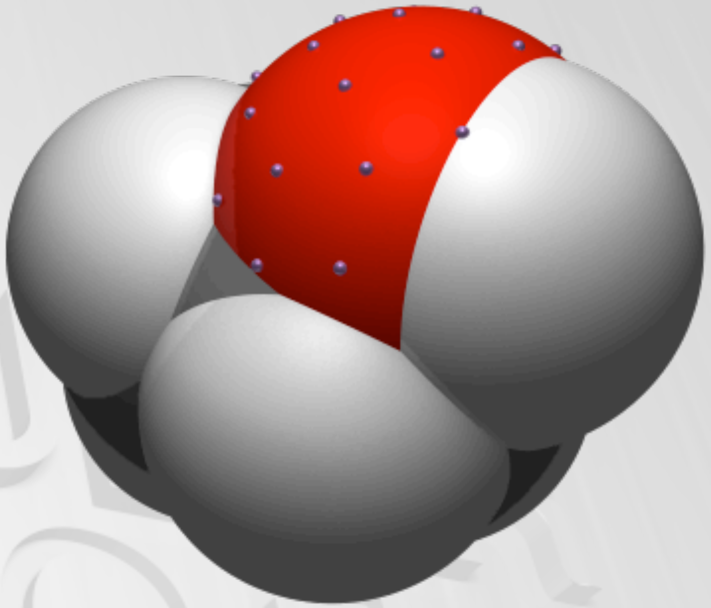
- The *z*-sections are discretised with a grid. Each cell is defined as *inside* the atoms, *outside* or *in-between*.



The Double Cubic Lattice Method (Gromacs)

- The DCLM is an algorithmic variant of the Shrake-Rupley approach (Eisenhaber, F. *et al.* (1995) *J. Comput. Chem.* 16, 273–284).

- Improvements to speed up the computation.



THE DOUBLE CUBIC LATTICE METHOD (GROMACS)

Lists of neighbouring atoms (test atoms)

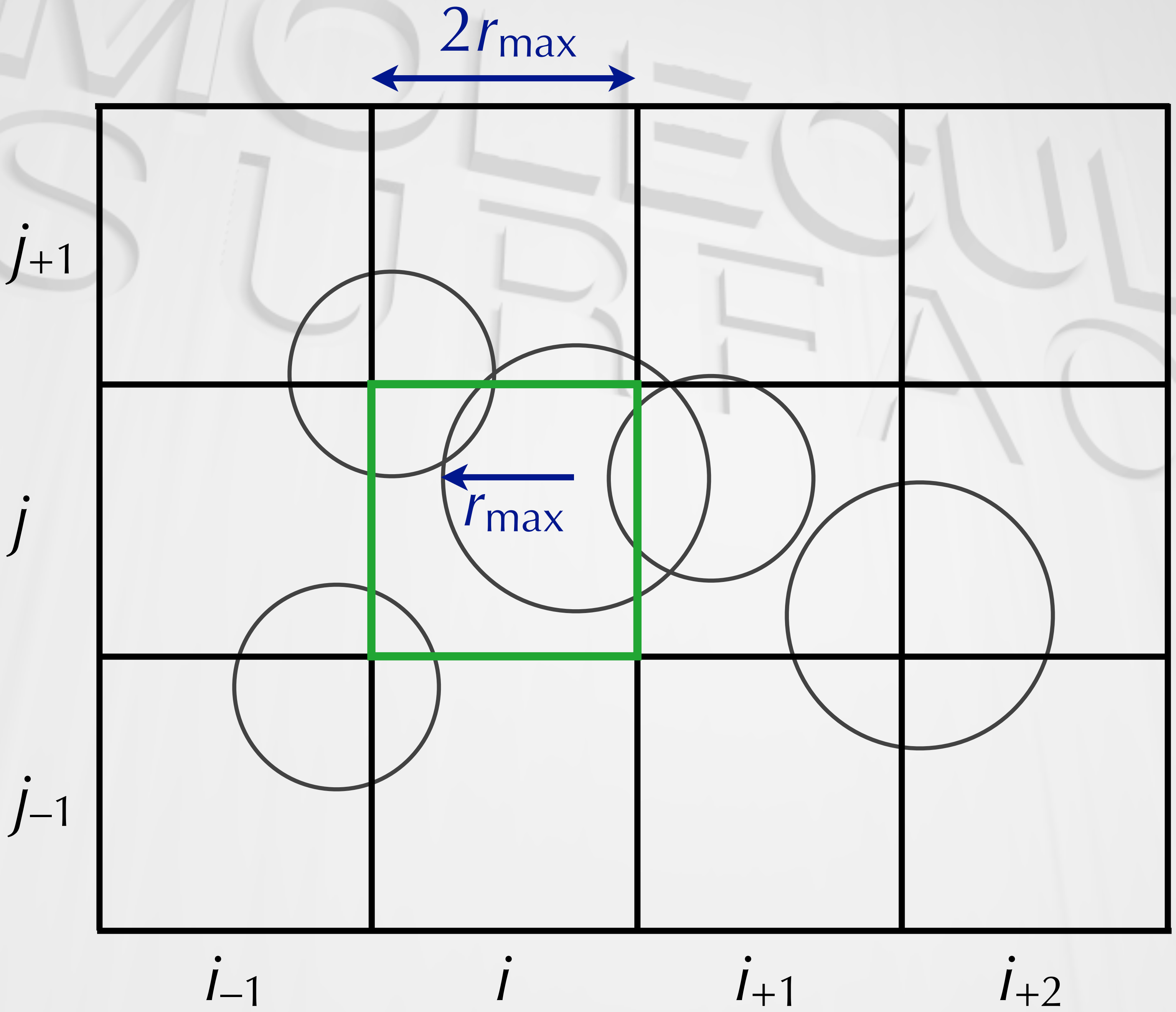
Shrake-Rupley approach

- no particular restriction in the generation of the cubic lattice
- searches through the whole grid for neighbours

DCLM

- cubic lattice with a spacing of $2r_{\text{max}}$ (r_{max}: largest extended radius in the molecule)
- limits the search in the central cell and in the cells in position ±1 (similarly to PBC/minimum image convention in MD).

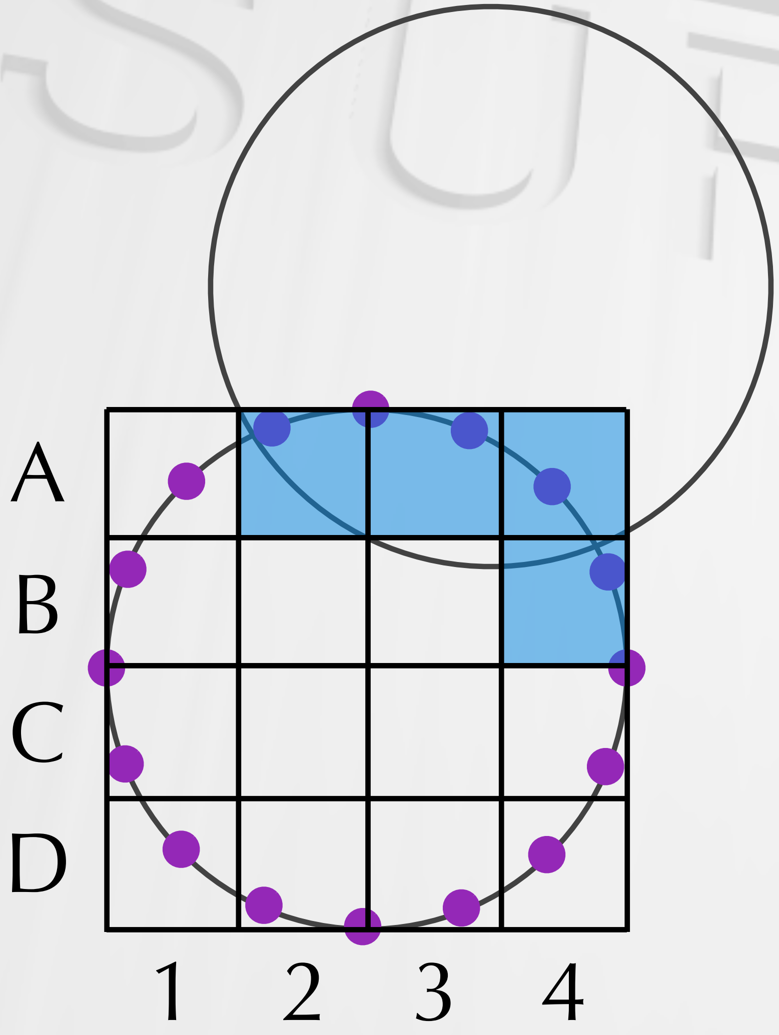
The Double Cubic Lattice Method (Gromacs)



The Double Cubic Lattice Method (Gromacs)

Shrake-Rupley approach	DCLM
Lists of neighbouring atoms (test atoms)	
- no particular restriction in the generation of the cubic lattice - searches through the whole grid for neighbours for each central atom	- cubic lattice with a spacing of $2r_{\max}$ ($r_{\max}$: largest extended radius in the molecule) - limits the search in the central cell and in the cells in position ±1 (similarly to PBC/minimum image convention in MD).
Lists of buried surface test points	
- checks all the combinations test atom–test point for each central atom	- starts from the principle that only test points lying in the area of overlap can be buried by a test atom - uses a second cubic lattice

The Double Cubic Lattice Method (Gromacs)

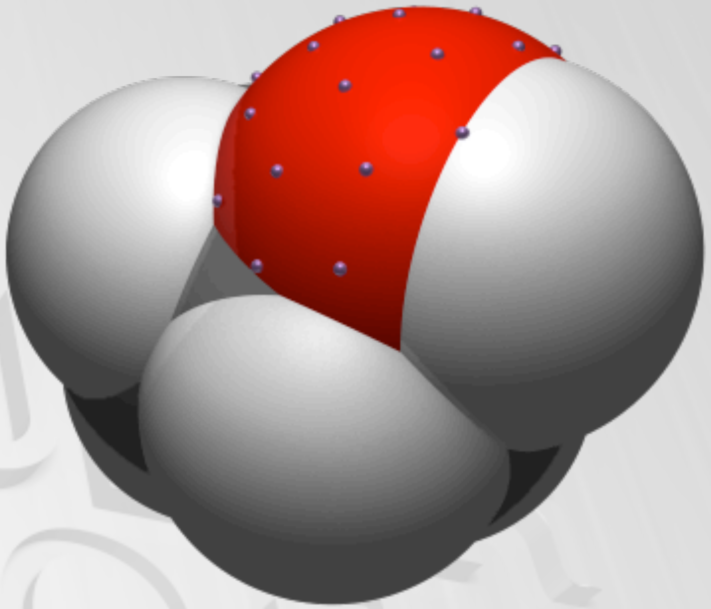


- Shrake-Rupley algorithm: checks boxes A1–4, B1, B4, C1, C4 and D1–4

- DCLM: checks only boxes A2–4 and B4

The Double Cubic Lattice Method (Gromacs)

- The DCLM is an algorithmic variant of the Shrake-Rupley approach (Eisenhaber, F. *et al.* (1995) *J. Comput. Chem.* 16, 273–284).

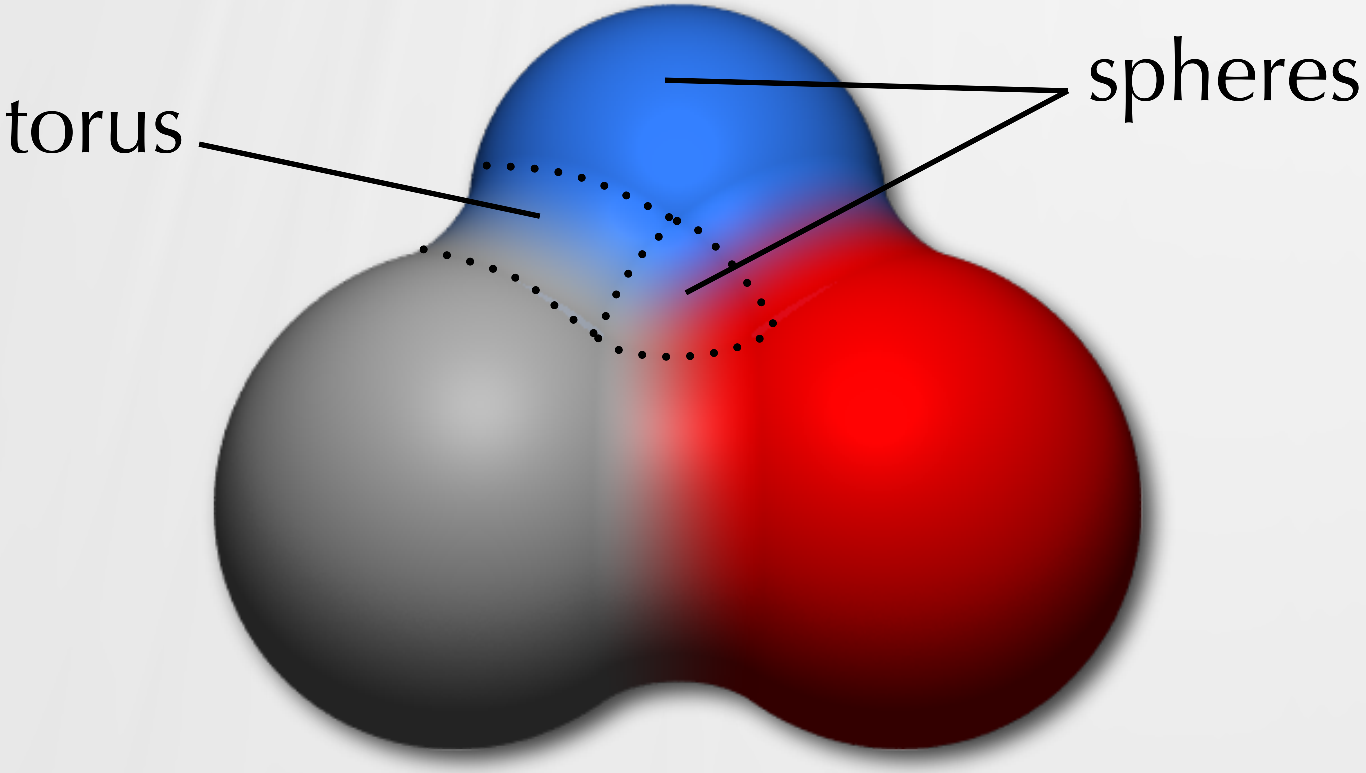


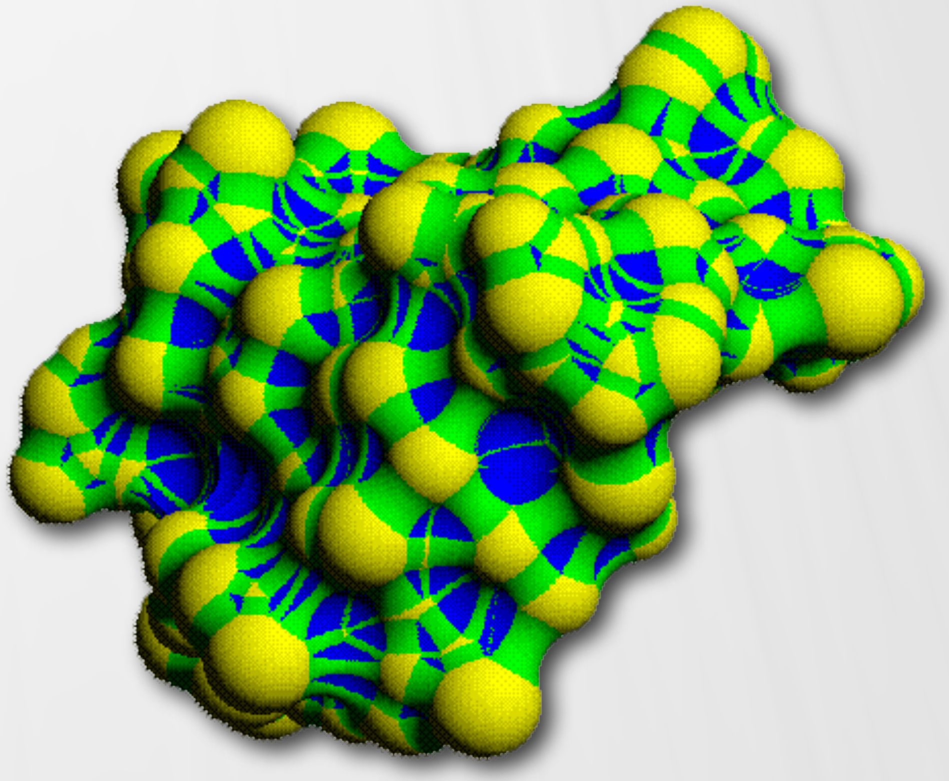
- Improvements to speed up the computation:
  - a first cubic lattice for the generation of the neighbour list (test atoms) for each central atom.
  - a second cubic lattice for the list of the occluding atoms for each central atom.
  - various optimisation methods for speed-up
    eg: reduced numbers of squares, square roots and trigonometric functions to calculate; the distance criterion *d* (or d^2) in the Shrake-Rupley algorithm is replaced by a dot product to save time.

ANALYTICAL METHODS

- Areas are calculated using equations appropriate for the shape of the surfaces:
  - A Lee-Richards surface consists of the union of convex spherical surfaces.
  - A Connolly surface is composed of:
    - convex spherical elements → contact surface
    - saddle-shaped toroidal elements
    - concave spherical elements

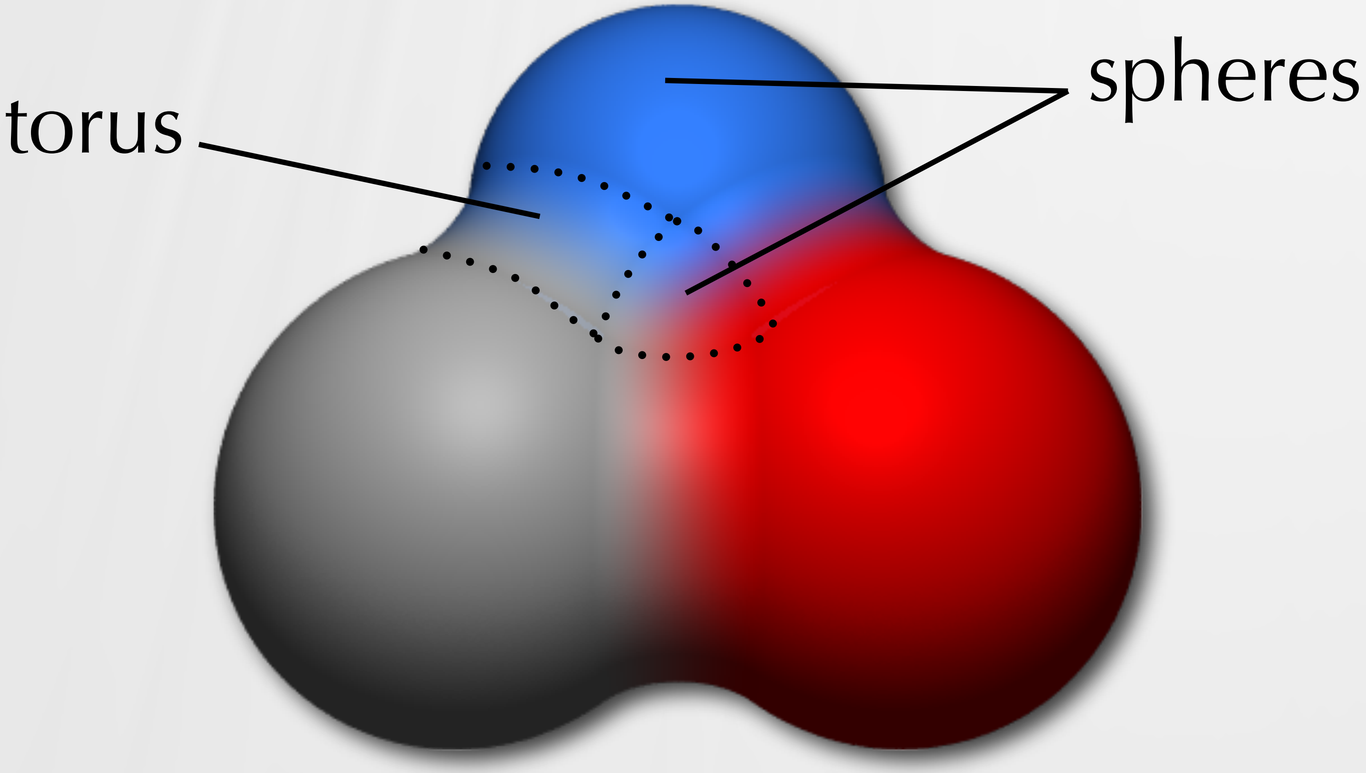
    } → re-entrant surface

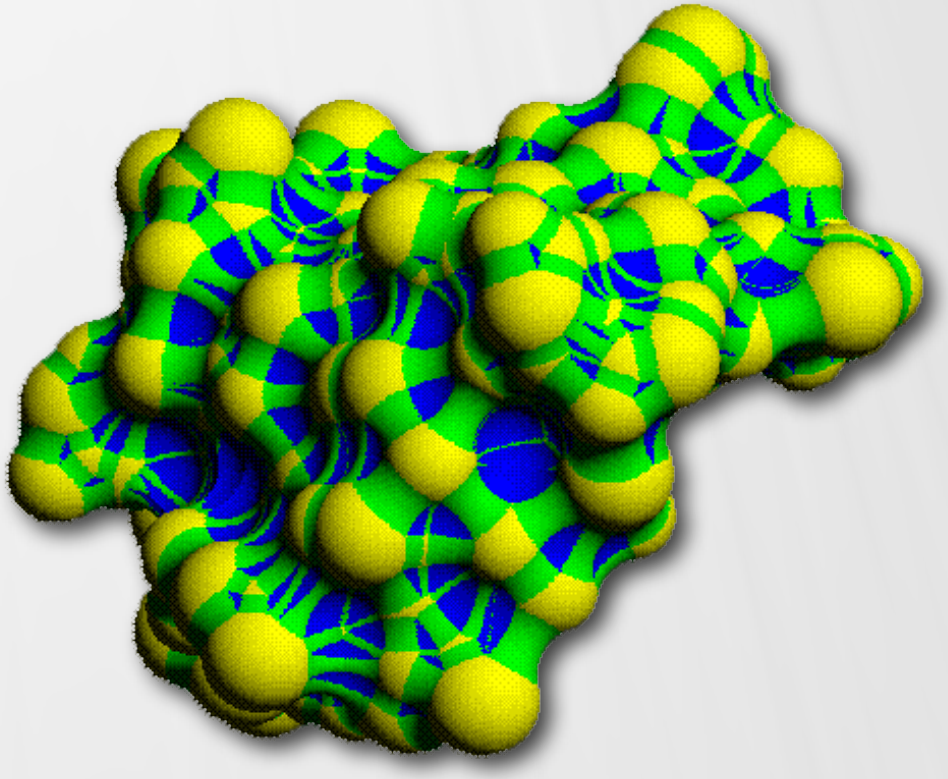




Analytical Methods

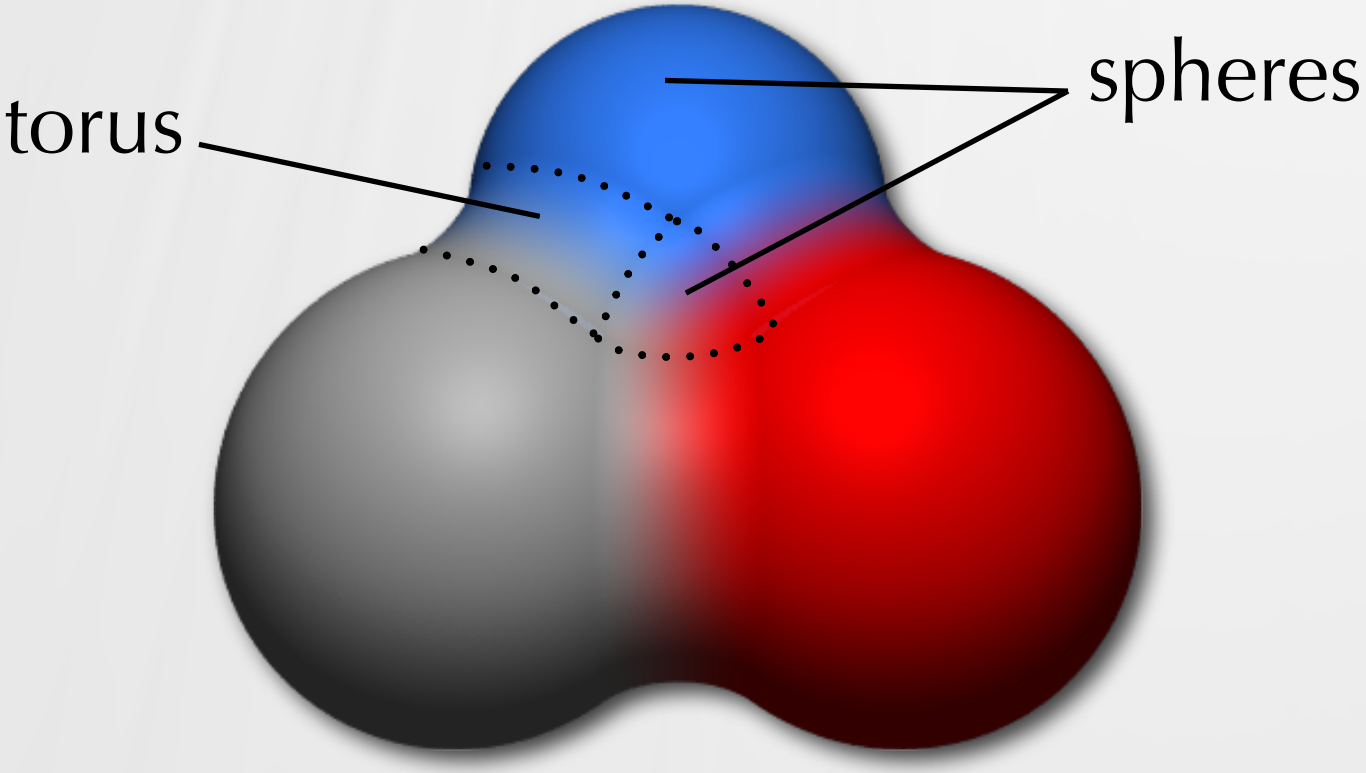
- Using the atomic coordinates, the van der Waals radii and the probe radius, a series of equations define all the geometric properties of the spherical and toroidal patches, eg:
  - the centre, the 2 radii and the axial vector of all the tori,
  - the position of the vertices and the concavity (*height*) of the concave pieces,
  - the centre and the radius of the convex pieces.
- Each element surface is defined by a set of circular arcs, for which the centre, the radius and the end points need to be determined.

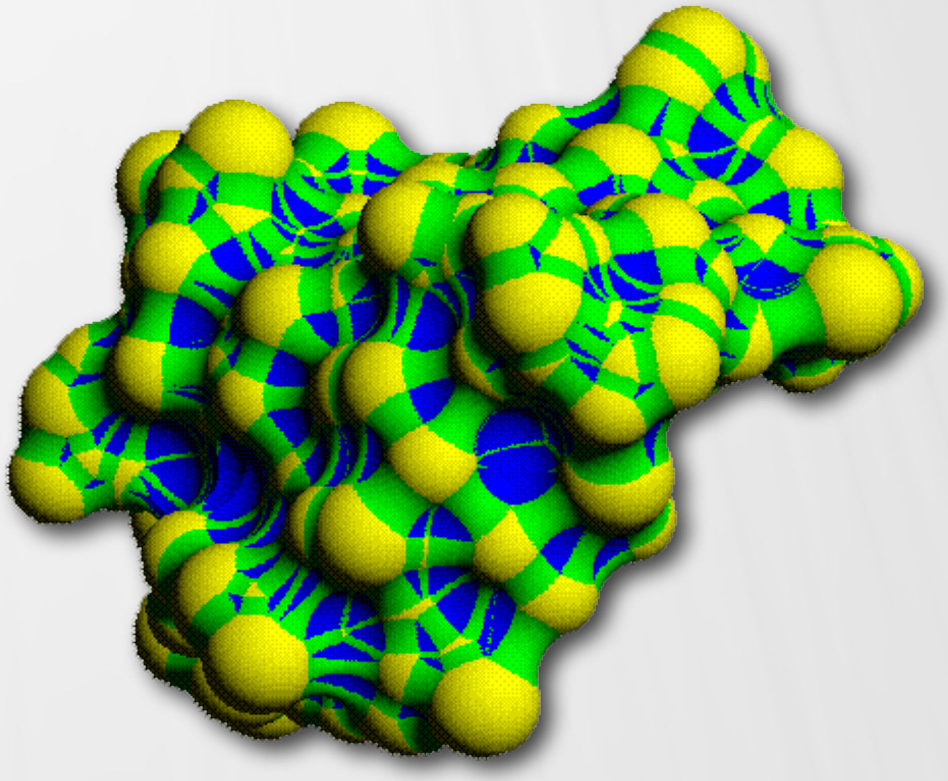




Analytical Methods

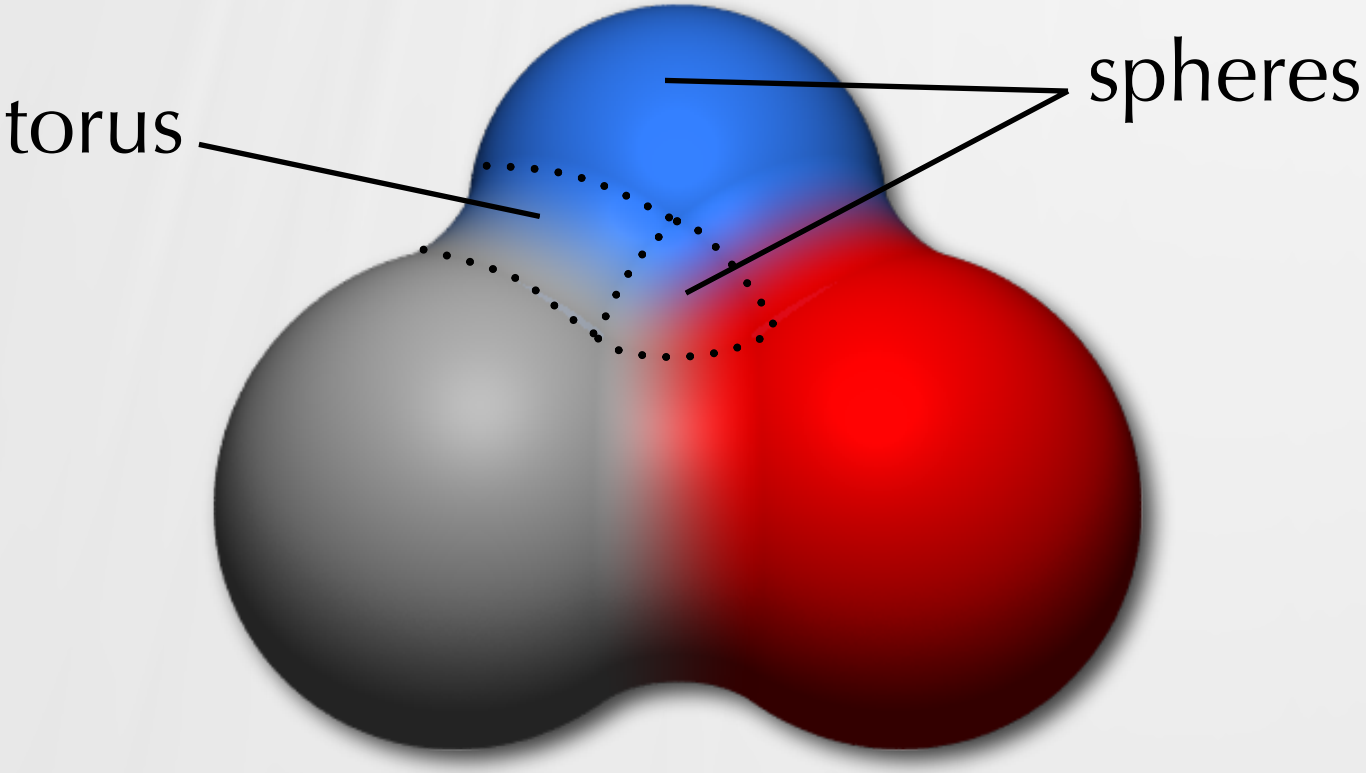
- Using the atomic coordinates, the van der Waals radii and the probe radius, a series of equations define all the geometric properties of the spherical and toroidal patches, eg:
  - the centre, the 2 radii and the axial vector of all the tori,
  - the position of the vertices and the concavity (*height*) of the concave pieces,
  - the centre and the radius of the convex pieces.
- Each element surface is defined by a set of circular arcs, for which the centre, the radius and the end points need to be determined.

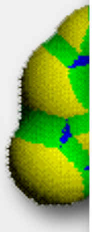




Analytical Methods

- Using the atomic coordinates, the van der probe radius, a series of equations define properties of the spherical and toroidal pa
  - the centre, the 2 radii and the axial vect
  - the position of the vertices and the conc concave pieces,
  - the centre and the radius of the convex
- Each element surface is defined by a set o the centre, the radius and the end points r





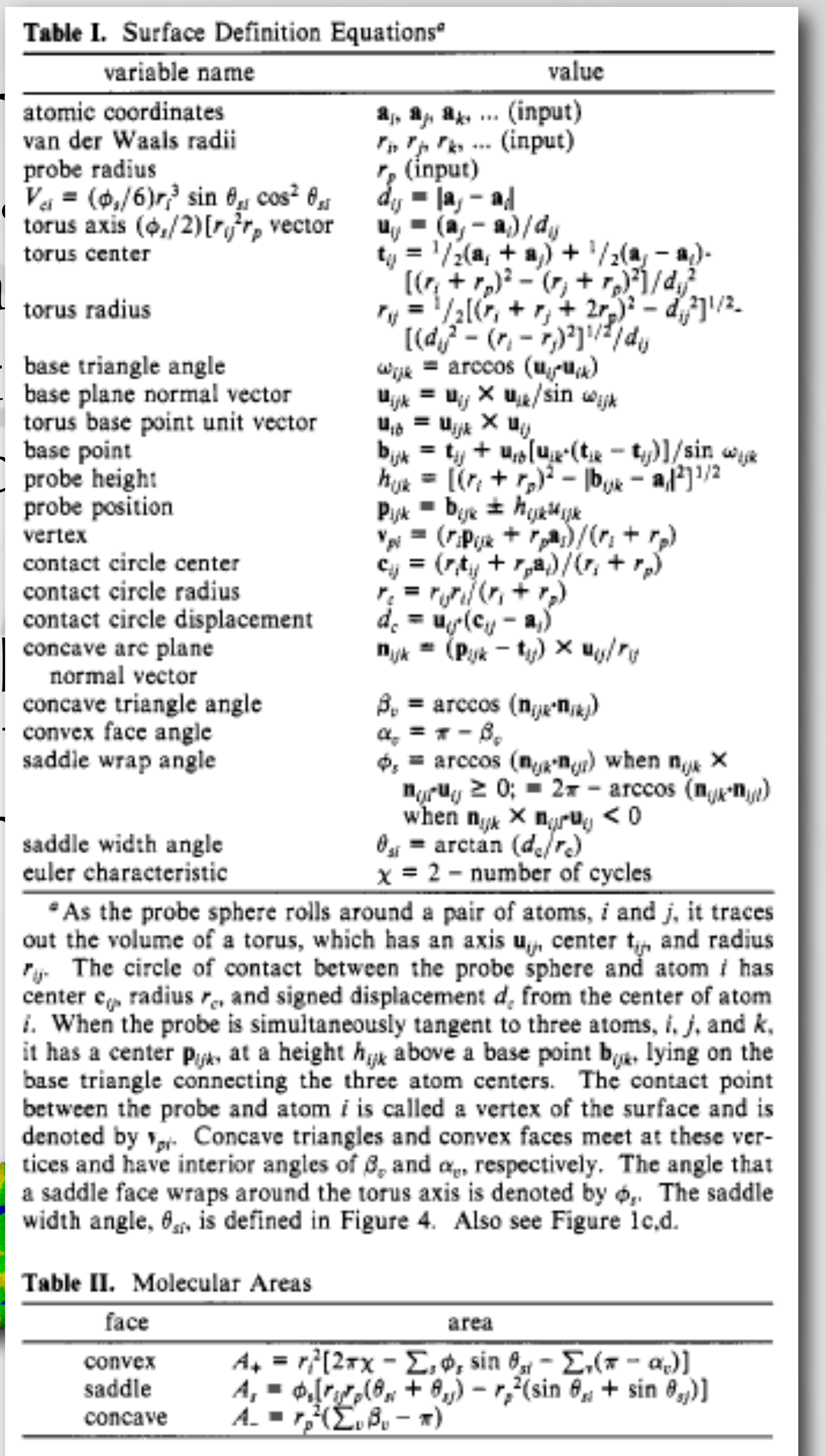


Table I. Surface Definition Equations[a]

variable name	value
atomic coordinates	$\mathbf{a}_i, \mathbf{a}_j, \mathbf{a}_k, \ldots$ (input)
van der Waals radii	$r_i, r_j, r_k, \ldots$ (input)
probe radius	r_p (input)
$V_{ci} = (\phi_s/6)r_i^3 \sin\theta_{si} \cos^2\theta_{si}$	$d_{ij} = \lvert\mathbf{a}_j - \mathbf{a}_i\rvert$
torus axis $(\phi_s/2)[r_{ij}{}^2 r_p$ vector	$\mathbf{u}_{ij} = (\mathbf{a}_j - \mathbf{a}_i)/d_{ij}$
torus center	$\mathbf{t}_{ij} = {}^1/_2(\mathbf{a}_i + \mathbf{a}_j) + {}^1/_2(\mathbf{a}_j - \mathbf{a}_i)\cdot [(r_i + r_p)^2 - (r_j + r_p)^2]/d_{ij}{}^2$
torus radius	$r_{ij} = {}^1/_2[(r_i + r_j + 2r_p)^2 - d_{ij}{}^2]^{1/2}\cdot [(d_{ij}{}^2 - (r_i - r_j)^2]^{1/2}/d_{ij}$
base triangle angle	$\omega_{ijk} = \arccos(\mathbf{u}_{ij}\cdot\mathbf{u}_{ik})$
base plane normal vector	$\mathbf{u}_{ijk} = \mathbf{u}_{ij} \times \mathbf{u}_{ik}/\sin\omega_{ijk}$
torus base point unit vector	$\mathbf{u}_{tb} = \mathbf{u}_{ijk} \times \mathbf{u}_{ij}$
base point	$\mathbf{b}_{ijk} = \mathbf{t}_{ij} + \mathbf{u}_{tb}[\mathbf{u}_{ik}\cdot(\mathbf{t}_{ik} - \mathbf{t}_{ij})]/\sin\omega_{ijk}$
probe height	$h_{ijk} = [(r_i + r_p)^2 - \lvert\mathbf{b}_{ijk} - \mathbf{a}_i\rvert^2]^{1/2}$
probe position	$\mathbf{p}_{ijk} = \mathbf{b}_{ijk} \pm h_{ijk}u_{ijk}$
vertex	$\mathbf{v}_{pi} = (r_i\mathbf{p}_{ijk} + r_p\mathbf{a}_i)/(r_i + r_p)$
contact circle center	$\mathbf{c}_{ij} = (r_i\mathbf{t}_{ij} + r_p\mathbf{a}_i)/(r_i + r_p)$
contact circle radius	$r_c = r_{ij}r_i/(r_i + r_p)$
contact circle displacement	$d_c = \mathbf{u}_{ij}\cdot(\mathbf{c}_{ij} - \mathbf{a}_i)$
concave arc plane normal vector	$\mathbf{n}_{ijk} = (\mathbf{p}_{ijk} - \mathbf{t}_{ij}) \times \mathbf{u}_{ij}/r_{ij}$
concave triangle angle	$\beta_v = \arccos(\mathbf{n}_{ijk}\cdot\mathbf{n}_{ikj})$
convex face angle	$\alpha_v = \pi - \beta_v$
saddle wrap angle	$\phi_s = \arccos(\mathbf{n}_{ijk}\cdot\mathbf{n}_{ijl})$ when $\mathbf{n}_{ijk} \times \mathbf{n}_{ijl}\cdot\mathbf{u}_{ij} \geq 0$; $= 2\pi - \arccos(\mathbf{n}_{ijk}\cdot\mathbf{n}_{ijl})$ when $\mathbf{n}_{ijk} \times \mathbf{n}_{ijl}\cdot\mathbf{u}_{ij} < 0$
saddle width angle	$\theta_{si} = \arctan(d_c/r_c)$
euler characteristic	$\chi = 2 -$ number of cycles

[a] As the probe sphere rolls around a pair of atoms, i and j, it traces out the volume of a torus, which has an axis $\mathbf{u}_{ij}$, center $\mathbf{t}_{ij}$, and radius r_{ij}. The circle of contact between the probe sphere and atom i has center $\mathbf{c}_{ij}$, radius r_c, and signed displacement d_c from the center of atom i. When the probe is simultaneously tangent to three atoms, i, j, and k, it has a center $\mathbf{p}_{ijk}$, at a height h_{ijk} above a base point $\mathbf{b}_{ijk}$, lying on the base triangle connecting the three atom centers. The contact point between the probe and atom i is called a vertex of the surface and is denoted by $\mathbf{v}_{pi}$. Concave triangles and convex faces meet at these vertices and have interior angles of β_v and α_v, respectively. The angle that a saddle face wraps around the torus axis is denoted by ϕ_s. The saddle width angle, θ_{si}, is defined in Figure 4. Also see Figure 1c,d.

Table II. Molecular Areas

face	area
convex	$A_+ = r_i^2[2\pi\chi - \sum_s \phi_s \sin\theta_{si} - \sum_v(\pi - \alpha_v)]$
saddle	$A_s = \phi_s[r_{ij}r_p(\theta_{si} + \theta_{sj}) - r_p{}^2(\sin\theta_{si} + \sin\theta_{sj})]$
concave	$A_- = r_p{}^2(\sum_v \beta_v - \pi)$

Connolly, M. L. (1985) *J. Am. Chem. Soc.* 107, 1118–1124

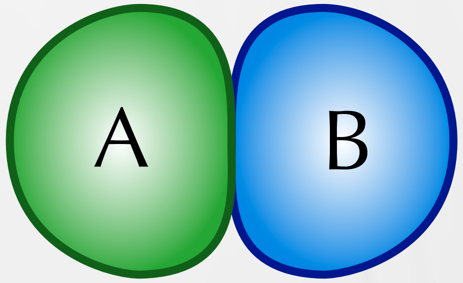
What Can You Do with These Surfaces?

- Measure the surface area (vdW, SASA, Connolly *etc*) of residues/ atoms!
  - from a simulation: variations with time
  - compare between different states (folding, binding) or proteins
  - compare with a reference (RSA, Relative Solvent Accessibility):

$$\mathrm{RSA}_{\mathrm{X}} = \frac{\mathrm{ASA}_{\mathrm{X,protein}}}{\mathrm{ASA}_{\mathrm{X,ref}}} \quad \text{(reference: Ala–X–Ala)}$$

- Calculate contact/interaction surface area between proteins, domains *etc*:

$$\mathcal{C}_{\mathrm{A,B}} = S_{\mathrm{A}} + S_{\mathrm{B}} - S_{\mathrm{A \cup B}}$$



Ideally, S_i should be Connolly surface areas but finding a program can be difficult, so SASAs may be used instead.

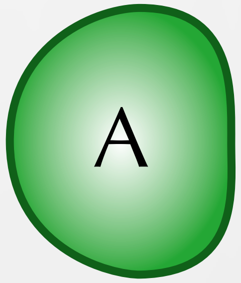
WHAT CAN YOU DO WITH THESE SURFACES?

- Measure the surface area (vdW, SASA, Connolly *etc*) of residues/atoms!
  - from a simulation: variations with time
  - compare between different states (folding, binding) or proteins
  - compare with a reference (RSA, Relative Solvent Accessibility):

$$\mathrm{RSA}_{\mathrm{X}} = \frac{\mathrm{ASA}_{\mathrm{X,protein}}}{\mathrm{ASA}_{\mathrm{X,ref}}} \quad \text{(reference: Ala–X–Ala)}$$

- Calculate contact/interaction surface area between proteins, domains *etc*:

$$\mathcal{C}_{\mathrm{A,B}} = S_{\mathrm{A}} + S_{\mathrm{B}} - S_{\mathrm{A\cup B}}$$



Ideally, S_i should be Connolly surface areas but finding a program can be difficult, so SASAs may be used instead.

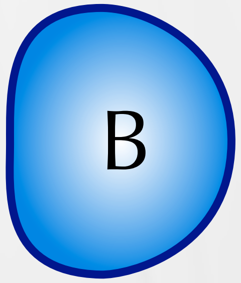
What Can You Do with These Surfaces?

- Measure the surface area (vdW, SASA, Connolly *etc*) of residues/ atoms!
  - from a simulation: variations with time
  - compare between different states (folding, binding) or proteins
  - compare with a reference (RSA, Relative Solvent Accessibility):

$$\mathrm{RSA}_X = \frac{\mathrm{ASA}_{X,\mathrm{protein}}}{\mathrm{ASA}_{X,\mathrm{ref}}} \quad \text{(reference: Ala–X–Ala)}$$

- Calculate contact/interaction surface area between proteins, domains *etc*:

$$\mathcal{C}_{A,B} = S_A + S_B - S_{A\cup B}$$



Ideally, S_i should be Connolly surface areas but finding a program can be difficult, so SASAs may be used instead.

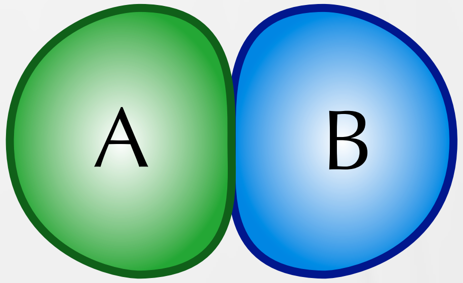
What Can You Do with These Surfaces?

- Measure the surface area (vdW, SASA, Connolly *etc*) of residues/ atoms!
  - from a simulation: variations with time
  - compare between different states (folding, binding) or proteins
  - compare with a reference (RSA, Relative Solvent Accessibility):

$$\mathrm{RSA}_{\mathrm{X}} = \frac{\mathrm{ASA}_{\mathrm{X,protein}}}{\mathrm{ASA}_{\mathrm{X,ref}}} \quad \text{(reference: Ala–X–Ala)}$$

- Calculate contact/interaction surface area between proteins, domains *etc*:

$$\mathcal{C}_{\mathrm{A,B}} = S_{\mathrm{A}} + S_{\mathrm{B}} - S_{\mathrm{A \cup B}}$$



Ideally, S_i should be Connolly surface areas but finding a program can be difficult, so SASAs may be used instead.

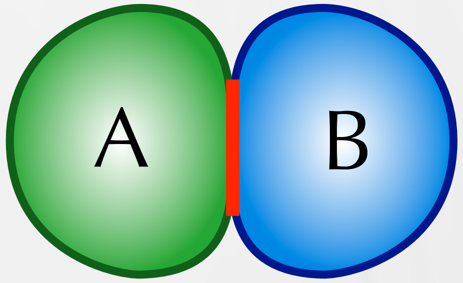
What Can You Do with These Surfaces?

- Measure the surface area (vdW, SASA, Connolly *etc*) of residues/ atoms!
  - from a simulation: variations with time
  - compare between different states (folding, binding) or proteins
  - compare with a reference (RSA, Relative Solvent Accessibility):

$$\mathrm{RSA}_X = \frac{\mathrm{ASA}_{X,\mathrm{protein}}}{\mathrm{ASA}_{X,\mathrm{ref}}} \quad \text{(reference: Ala–X–Ala)}$$

- Calculate contact/interaction surface area between proteins, domains *etc*:

$$\mathcal{C}_{A,B} = S_A + S_B - S_{A\cup B}$$



Ideally, S_i should be Connolly surface areas but finding a program can be difficult, so SASAs may be used instead.

WHY STUDYING THE SURFACE OF PROTEINS?

- Also used to study hydration
  eg in implicit solvation models: GBSA (Generalized Born with Solvent Accessibility). GBSA is a GB model which includes a solvent accessibility term:

  $$\Delta G_{\mathrm{solv}} = \sum_{i=1}^{n} a_i \, \sigma_i$$

  ΔG_{solv}: free energy of solvation of a solute (*n* atoms)
  a_i: accessible surface area of atom *i*
  σ_i: solvation parameter of atom *i* (contribution to the free energy of solvation of atom *i* per surface unit area)
  (cf Pramod)

- Connolly surfaces have been used in rational drug design and more generally in the study of protein–ligand and protein–protein interactions (eg docking of a ligand in binding pocket, identification of possible antigenic determinants on viruses).

Numerical Methods vs Analytical Methods

Analytical methods	**Numerical methods**
- Generate a *continuous* surface.	- As the surface is *discretised*, it is not continuous.
- As surfaces are represented as formulae, any mathematical (eg differentiation) can be applied to it...	
- ... but some methods have technical difficulties (eg: when the probe is tangent to 4 atoms)	
- *Accurate* calculation	- *Approximate* surface area (errors generally in the range of ±0.5–3 Å^2).
- Can be slower and/or limited by the number of atoms.	- Can be faster, depending on the discretisation level.

Things to Keep in Mind

There can be slight differences between the values of surface areas given by different programs. They can be due to:

- the method itself
- the values of the van der Waals radii used
- the value of the probe radius: some programs will take a default value of 1.4 Å for water whereas others will take 1.5 Å
- the level of discretisation used in numerical methods: Δz spacing (Lee-Richards), density of test points (Shrake-Rupley, DCLM)
- the level of description of the molecules: extended heavy atoms or use of hydrogens
- all other assumption implied by the method.

Things to Keep in Mind

There can be slight differences between the values of surface areas given by different programs. They can be due to:

- the method itself
- the values of the van der Waals radii used
- the value of the probe radius: some programs will take a default value of 1.4 Å for water whereas others will take 1.5 Å
- the level of discretisation used in numerical methods: Δz spacing (Lee-Richards), density of test points (Shrake-Rupley, DCLM)
- the level of description of the molecules: extended heavy atoms or use of hydrogens
- all other assumption implied by the method.

Does the absolute value of the surface area matter? Maybe not; often it is the change relative to a reference that is the most important or informative.

MOLECULAR SURFACES

MOLECULAR SURFACES

That's all!

MOLECULAR SURFACES

THANK YOU