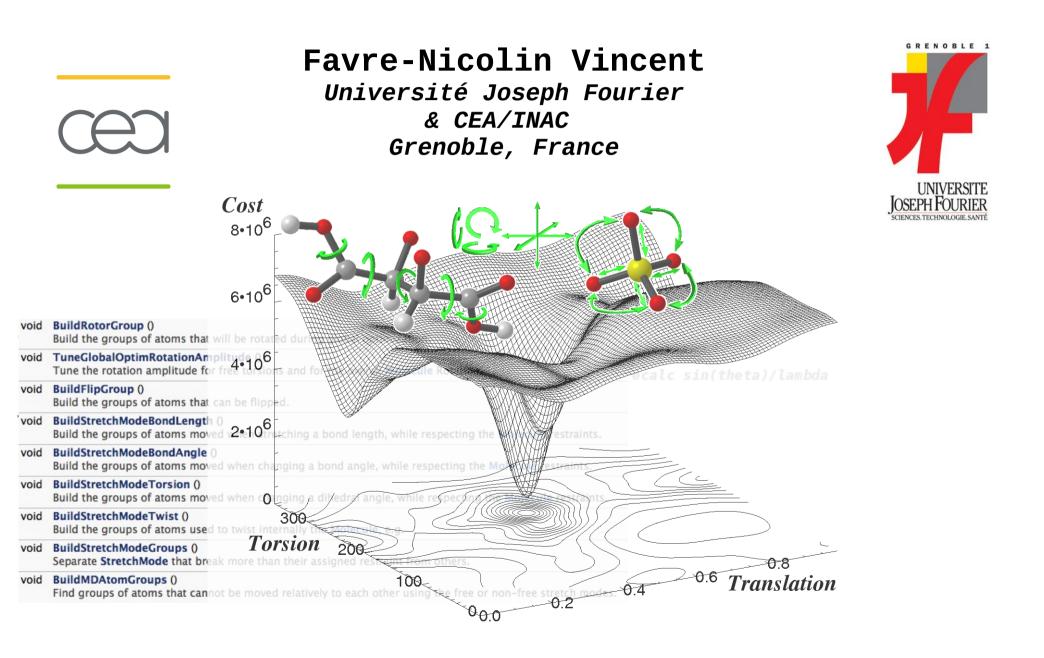
## Structure Solution in Real Space



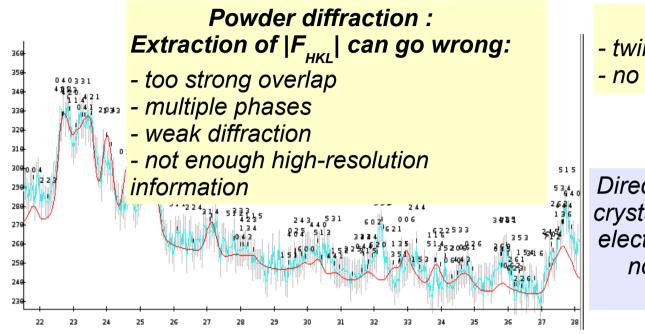
# Foreword

• Some examples in this presentation will refer specifically to powder diffraction (and some features of Fox/objcryst++)

- However: all the **algorithms** and **modeling principles** apply to:
  - *any* type of data (powder, single crystal)
  - any type of radiation (neutron/X-rays)
- Examples will refer to:
  - "small" molecules (n<sub>atoms</sub> <200)</li>
  - inorganic structures
- Algorithms also apply to macromolecules, if the number of parameters has been reduced (rigid bodies, TLS, soft modes...)

# Introduction: Structure Determination in Direct Space

#### Real (Direct)-Space Methods vs. Reciprocal-Space (Direct) Methods



#### Single Crystal:

- twinning/reflexion overlap
- no high resolution data available

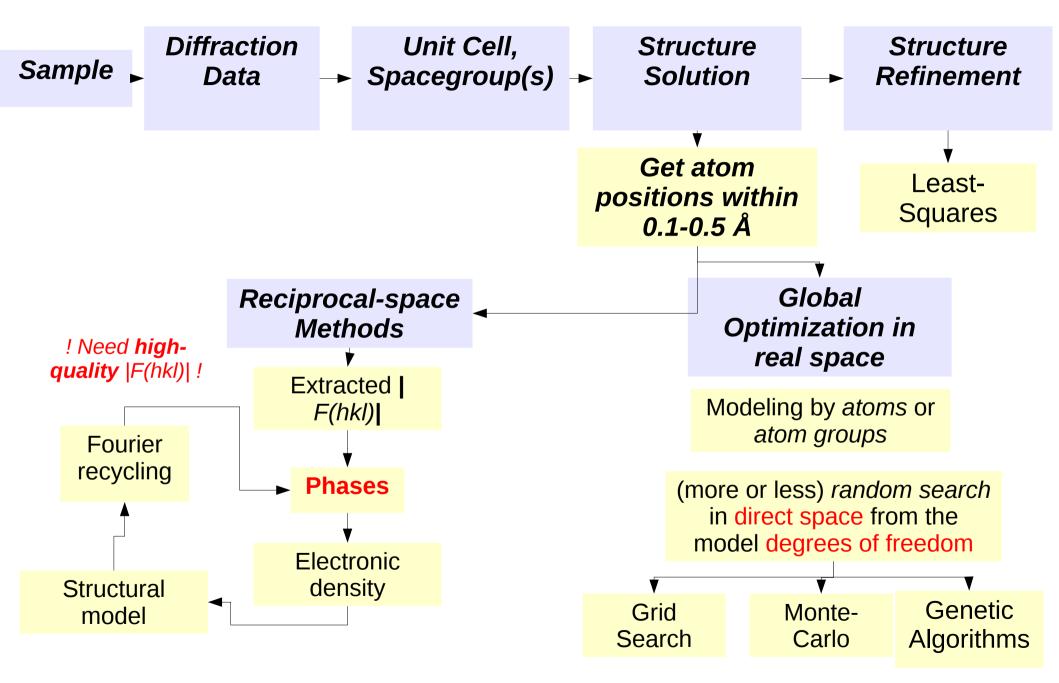
Direct methods are powerful (using full crystallographic formalism to derive the electronic density in seconds) but may not recover from bad structure factors

Real-Space structure solution: try **many** configurations until a satisfactory one is found => **brute-force approach** enabled by the increase in computing power A **basic** but **robust** approach to structure solution

#### Limited requirements on data resolution :

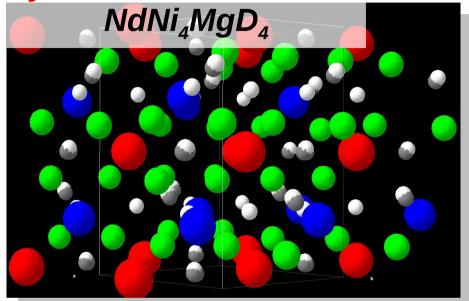
need more observed |Fhkl| than parameters (preferably many more) usually, a resolution of 2.5 Å is enough for most small molecules/inorganic structures, less if rigid bodies are used

# Solving Structures

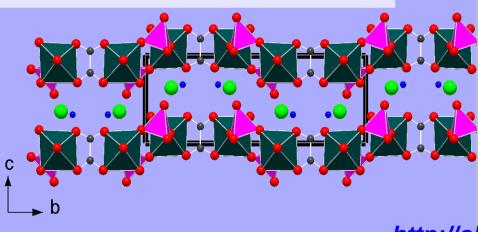


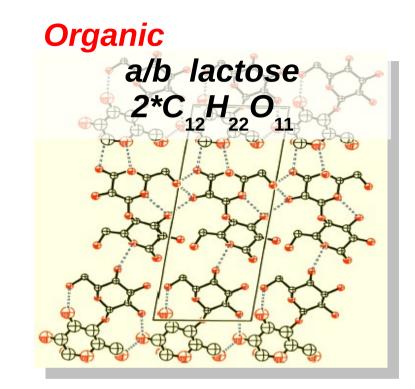
## Examples of structures solved

#### Hydrides:



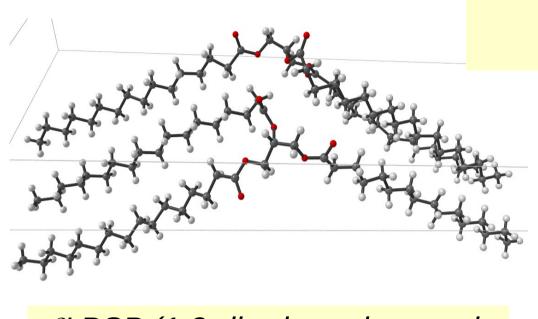
Inorganic:  $Na_2[VO(PO_4)]_2(C_2O_4).2H_2O$ 



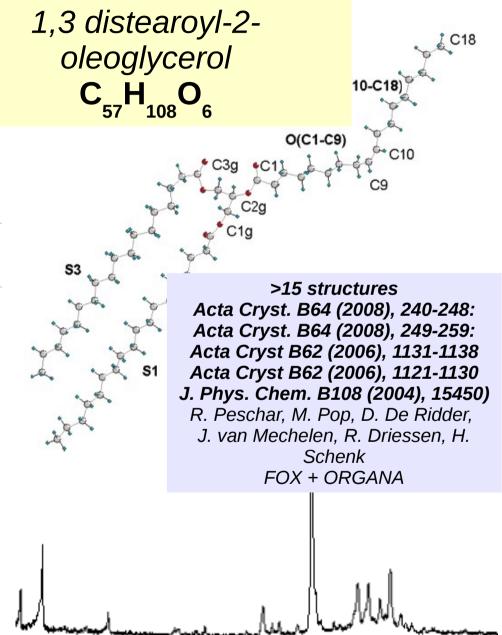


http://objcryst.sourceforge.net/Fox/FoxBiblioStructures

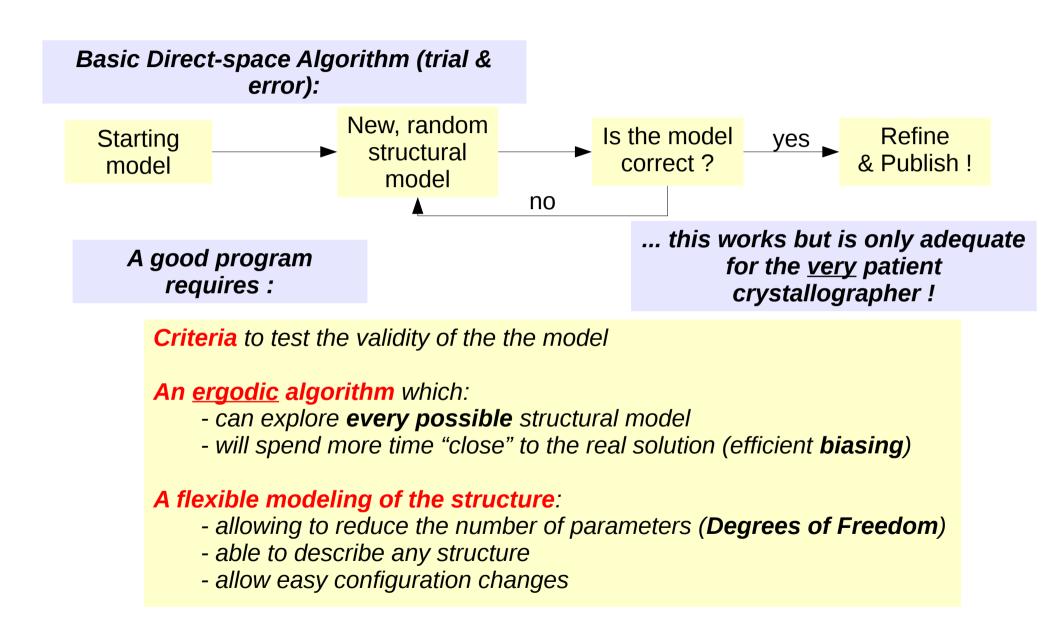
# Triglycerides



 $\beta'$  PSP (1,3-di-n-hexadecanoyl-2-n-octadecanoyl glycerol)  $C_{53}H_{102}O_{6}$ up to 56 non-H free torsion angles ! FOX > 2 months

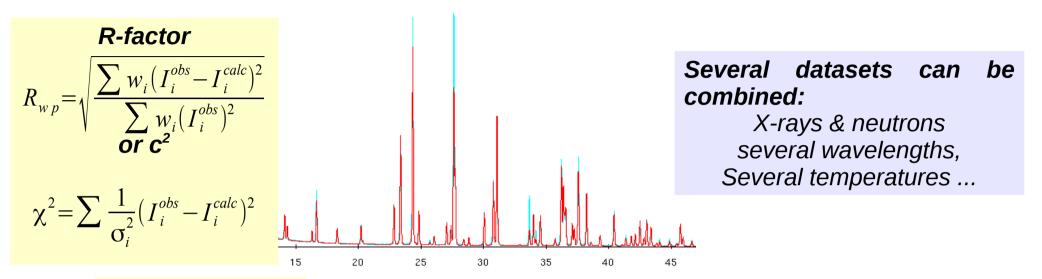


# **Real-Space Exploration ?**



# Criteria for Minimization

#### Criteria to Evaluate Trial Structural Models Diffraction Data



integrated profiles (i x² and iR<sub>wp</sub>)

Integrated profiles allow to avoid the requirement of a perfect description of profiles Why not use extracted structure factors (faster & equivalent) ?

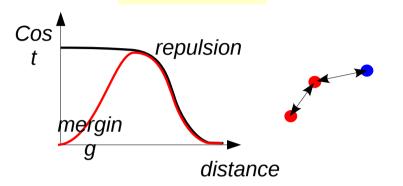
- This would require a **perfect description** of the profiles and background, which can be difficult ("real" samples, with ill profiles and multiple phases, background difficult to "guess" for close-packed reflections).

- Direct-space algorithms are necessary for samples where the extraction of structure factors is difficult

- with "integrated profiles", the full pattern is not calculated and the speed is equivalent to extracted structure factors

#### Criteria to Evaluate Trial Structural Models AntiBump Restraint

#### Anti-bump

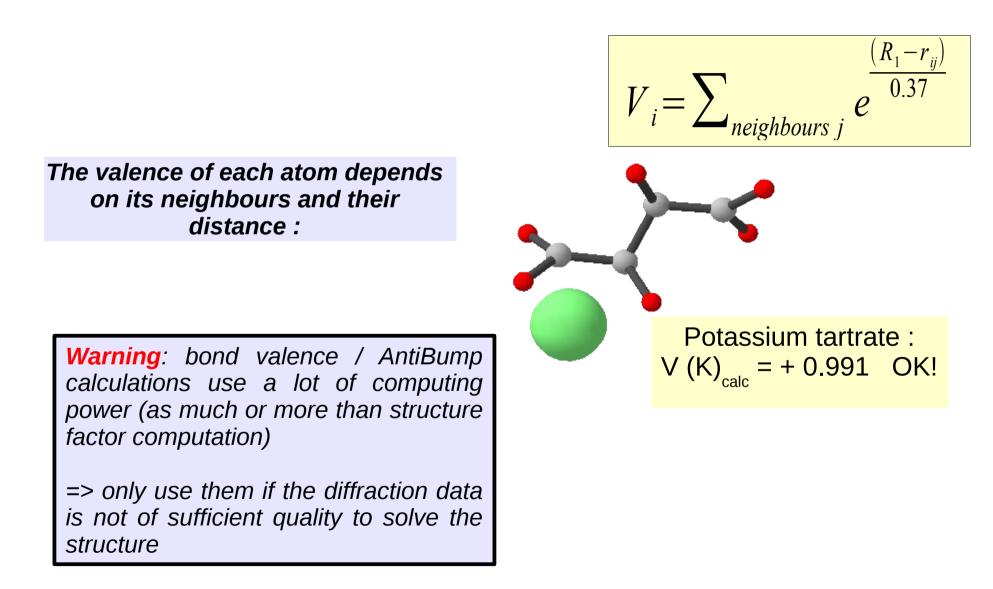


An **AntiBump** function allows the repulsion of atoms while permitting the "merging" of identical atoms on special positions or connecting several polyhedra

*Energy calculations ? Either internal energy for molecules or global for the entire unit cell* 

... But energy calculations are extremely costly from a computation point of view

## Criteria to Evaluate Trial Structural Models Bond Valence



## Criteria to Evaluate Trial Structural Models Combining Several Criteria

Problem: different criteria will have different scales !!

When combining experimental data,  $\chi^2$  can be summed:

$$\chi^{2} = \sum_{data \ 1} \frac{1}{\sigma_{i}^{2}} (I_{i}^{obs} - I_{i}^{calc})^{2} + \sum_{data \ 2} \frac{1}{\sigma_{i}^{2}} (I_{i}^{obs} - I_{i}^{calc})^{2} + \dots$$

=> avoid using R-factors which cannot be summed Fringe benefit: using  $\chi^2$  makes you ready for maximum likelihood (ML)

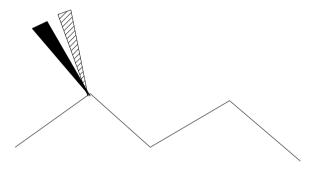
Sometimes combining « incompatible » criteria ( $\chi^2$ , energy, antibump) is necessary => finding the correct scale can be difficult. => correct scale factors can be guess if you know the 'target' values :

e.g.  $\chi^2$  should converge towards **Nobs** (Goodness-Of-Fit=1), antibump towards 0, etc..

Sometimes scaling different data sets is necessary (e.g. combine powder diffraction data from synchrotron and neutron) : statistically, no scale should be applied, but for « global optimisation » algorithms rules may be bent (see later ML slides)

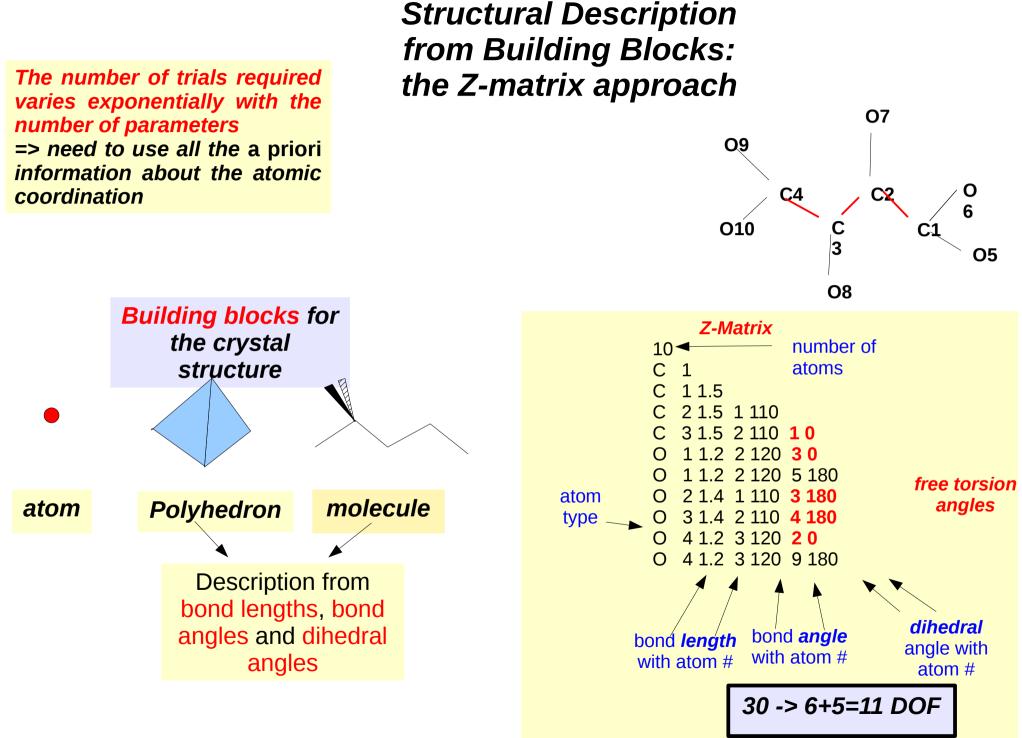
# Model Building: Real Space Parametrization

The number of trials required varies exponentially with the number of parameters => need to use all the a priori information about the atomic coordination Structural Description from Building Blocks: the Z-matrix approach



Building blocks for the crystal structure

atom



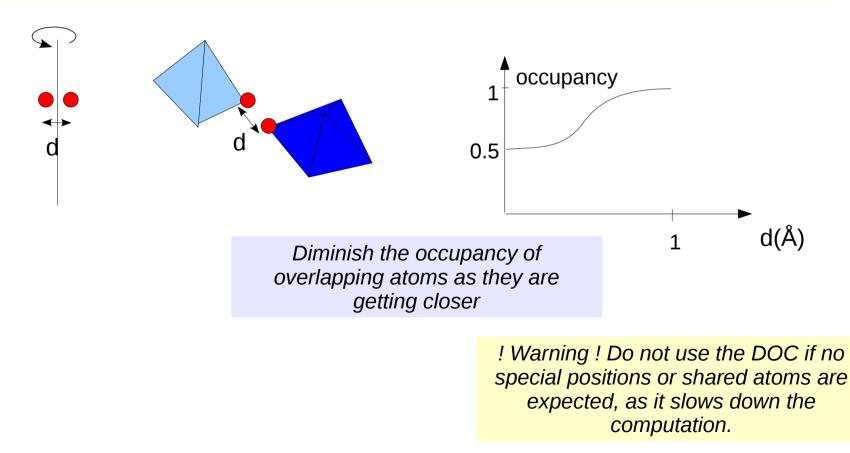
#### **Dynamical Occupancy Correction**

Inorganic structures often have atoms in **special positions**, and have **atoms common to several polyhedra**.=> New algorithm which must:

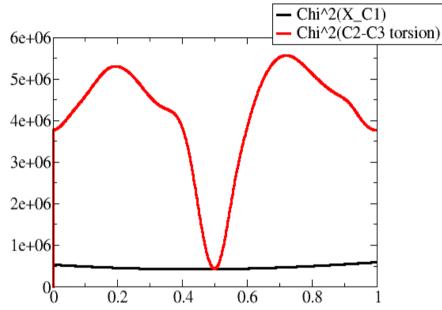
- correctly describe atoms in special positions without a priori knowledge

- allow the atoms to move continuously to and from the special positions

- not depend on the type of compound or the modelling chosen (atoms, polyhedra, molecules)

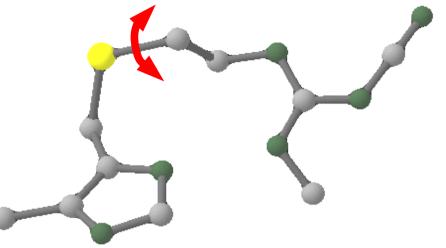


## Pitfalls of internal coordinates (zmatrix)



a **torsion angle** (moving many atoms) has a **much narrower minimum** than a translation parameter of an individual atom

=> even if the number of degrees of freedom diminishes, the global minimum is much narrower



Atoms are deduced from previous atoms => the first atoms in the z-matrix must also be the first to be found => The convergence can depend on the order of the atoms in the z-matrix

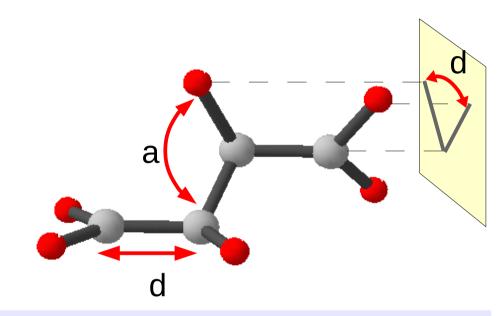
The z-matrix approach reduces the parameter space to explore, but makes it (much) more difficult to find the solution

Flexible Approach using Restraints

*idea*: keep all the coordination information, but with a *flexible approach* 

All atom positions are directly defined by their xyz coordinates and the coordination information is introduced by restraints on: - bond lengths  $\chi^2 = \frac{(d-d_0)^2}{\sigma_d^2}$ - bond angles - dihedral angles  $\chi^2 = \frac{(\alpha - \alpha_0)^2}{\sigma_d^2}$ 

The orientation of the molecule is defined by a **quaternion** (to avoid "gimbal lock" angles)

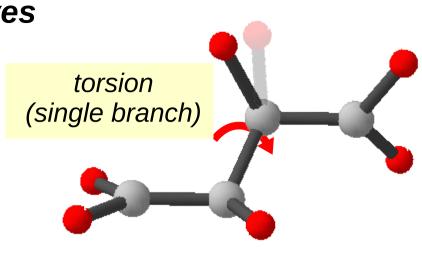


- this modelization is **independent from the order of the atoms**
- **any type of restraint** can be introduced
- any type of movement can be directly done (no need to compute complex torsions)
- any **cycle** can be defined

#### Making the Smart Moves

With atoms defined independently, it is vital to have **intelligent moves** that do not break the restraints

torsion



individual

random

moves

All torsion & flip moves that do not break restraints are **automatically identified** 

After each random move, a test is made on the total internal restraint cost to see if the configuration is kept

flip

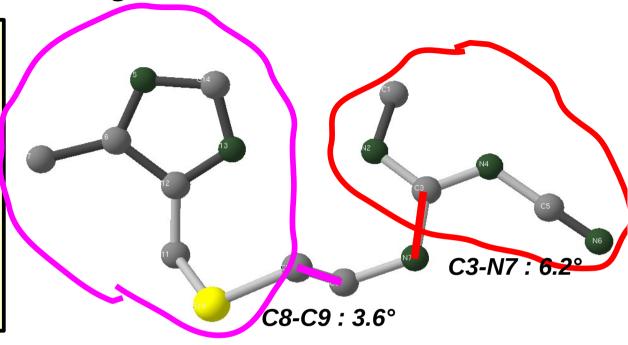
#### Adaptative Conformation Changes

Random torsion angle changes :

rotate the smallest fragment
 tune the max. rotation so that
 the average displacement is
 0.1Å.

- same for bond angle changes

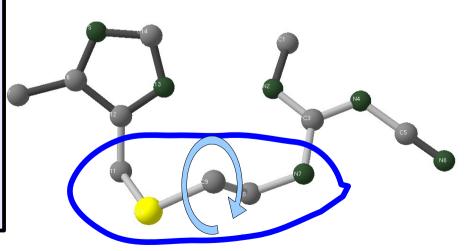
- tune global rotation of molecule



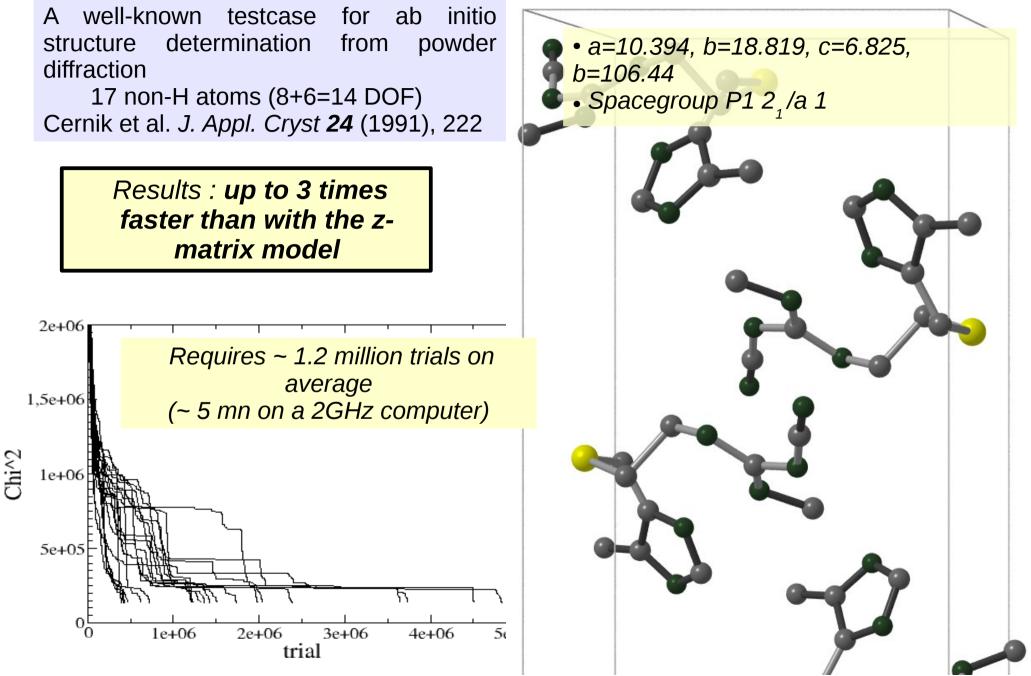
*"Twist "mode :* alter an *internal* part of a chain/cycle

=> long chains, flexible cycles

<u>TODO</u>: determine " **soft modes** " of the molecule and use them to distort the molecule (computationnally costly)

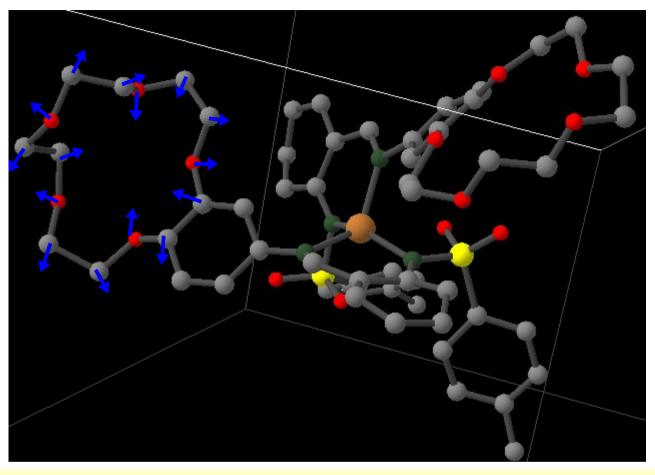


#### Cimetidine



# Model Building: Molecular Dynamics for Flexible Cycles

## **Using Molecular Dynamics**



Atoms in "restrained" groups are moved using molecular dynamics principles :

- Each atom is given a random vector speed
  - The overall Energy is E<sub>kinetic</sub> + E<sub>restraints</sub>
- Atoms are moved according to standard mechanics (force=gradient of E restraints)

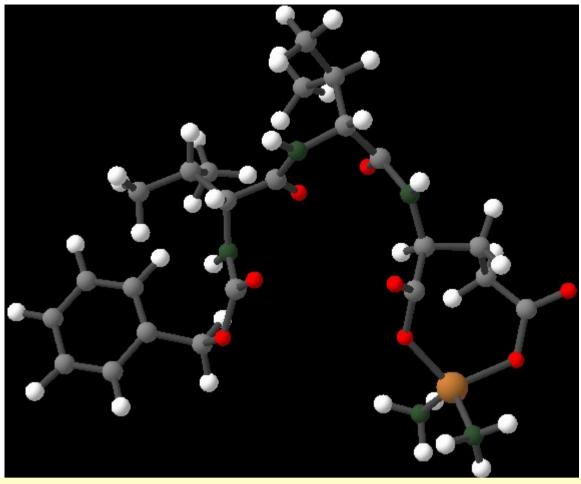
## **Using Molecular Dynamics**

Molecule	pf-9+h.fhz					٨		
File	Parameters	Formula	& Restraints	Manipulate	Geometry			
Flexibility M	Iodel Autom	atic from Re	estraints, str	ict		*		
Auto Optimi	ze Starting Co	onformatior	No 🛔					
Optimize Or	ientation Yes	5 🛓						
Rotation Ce	nter Geomet	rical center	(recommen	ded) 🛓				
_xR 🖌 L 🗆	0.432748_yR	🖌 L 🗆 0.25	5398(_zR 🖌	L 0.458472		L 🖌 1.(		
MD moves f	frequency 0.0	00000 MD n	Contraction of the second second second				2	
Center Aton	n:	No atom !	Er	ergy of Moleo	ule for Mo	lecular Dy	namics Moves	
	Name	Туре		andard amplit		all distortio	on of the Molecule)	
1	N1	N						
2	N2	N	-3.239250	-5.647540	-2.54223	III		

MD moves are **computationally expensive** => they are only tried once in a while => the **frequency** can be chosen (by default: 0=never) => the **relative energy** of the molecule can be chosen to avoid too much distortion

... But remember that SOME DISTORTION IS NECESSARY to reach the 'true' conformation of the Molecule, starting from an incorrect one...

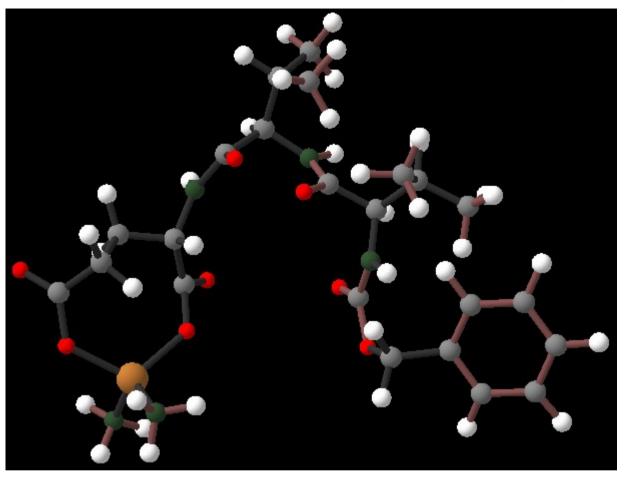
## Using Molecular Dynamics + least squares



MD moves allow to solve complex, flexible structures with large cycles... ...but it can take a long time !

Using periodic least squares greatly helps the convergence, as the least squares algorithm moves all the atoms individually (taking into account restraints) and is not limited by simple moves, or a z-matrix description

## Molecular Dynamics + least squares + rigid bodies



SOME DISTORTION IS NECESSARY... but sometimes you really want to avoid it => You can create 'rigid groups' of atoms that will only be translated/rotated as a rigid body, even during least squares.

# Model Building: New Approach: TRY

# TRY: a general purpose program for structural analysis

TRY is a computer programme which helps in crystal structure analysis of hard problems, with many atoms and few diffraction data, from the structure solution to the refinement stage.

TRY allows structural analysis by trial-and-error, random variables, genetic algorithms.

TRY has been structured so that the various constructions invoked are specified symbolically according to a conventional syntax.



Direct space search using a model

**PTRY** is a TRY evolution conceived for polypeptides structures solution and refinement.

Macromolecules **2003**, 36, 3666. J. Chem. Info. Model. **2007**, 47, 2263. J. Appl. Cryst. **2007**, 40, 1044.

*J. Appl. Cryst.* **2008**, 41, 784. *J. Appl. Cryst.* **2009**, 36, 3666.

Slide courtesy of L Erra / A Immirzi

# The model building



- Structure description is based on the internal parameter (3N -6 for molecular structures) , always chosen within a *non redundant coordinate system.*
- For linear molecules, the z-matrix method is used.
- In cyclic or polycyclic molecules the use of the z-matrix gives rise to redundancy: the variables exceed the number of degrees of freedom.
- A strictly non redundant procedure is always possible!

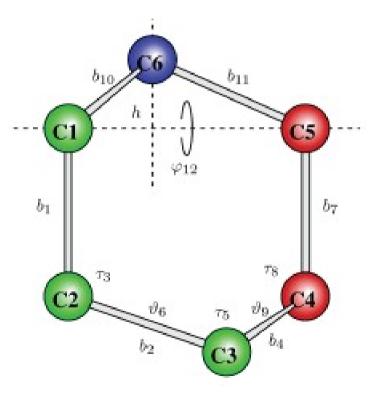
# The internal coordinates: cyclic molecules

- $3N\mathcal{J}$  6 coordinates ( $g_i$ ), being six the rigid body coordinates.
- N bond-lengths
- 2N angular coordinates.

#### Building procedure for cyclohexane

Eyring's procedure for building atoms C1, C2, . . . C5 (9 gi are employed altogether, 4 bond lengths, 3 bond angles, and 2 torsion angles).

A new machinery for building atom C6 using two bond lengths and one angle only, a so called bending angle ( $\phi$ );



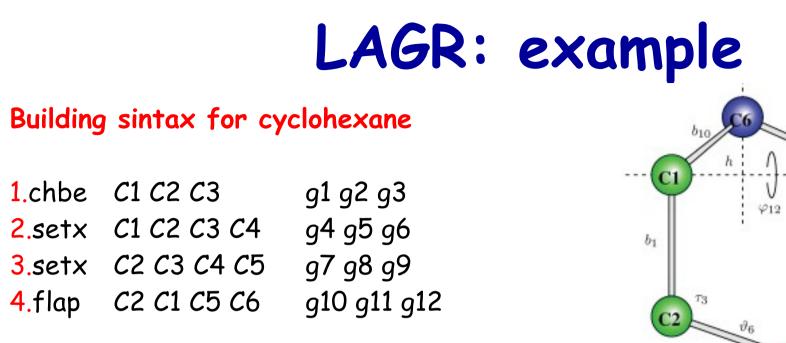
# LAGR: a general procedure to build molecular structures by using internal coordinates

All the mentioned constructions have been programmed devising an unique subroutine termed LAGR performing the whole construction at each call.

LAGR has been structured so that the various constructions invoked are specified **symbolically** according to a conventional syntax with one line of data for each construction step.

With the LAGR subroutine it is also possible to perform rotations, translations and other orthogonal transformations.

There are commands for finding the centre of gravity and the inertial axes. Special commands were designed for building linear polymers having chain symmetry viz. helices and glide-planes. Other commands allow symmetric constructions, e.g. methyl groups, aromatics, ecc.



#### 1.Build C1, C2 and C3

-g1 = bond length C1-C2	= b1
-g2 = bond length C2-C3	= b2
-g3 = bond angle C1-C2-C3	= тЗ

#### 2.Build C4

-g4 = bond length C3-C4 = b4 -g5 = bond angle C2-C3-C4 = т3 -g6 = torsion angle C1-C2-C3-C4 = Θ6

#### 3.Build C5

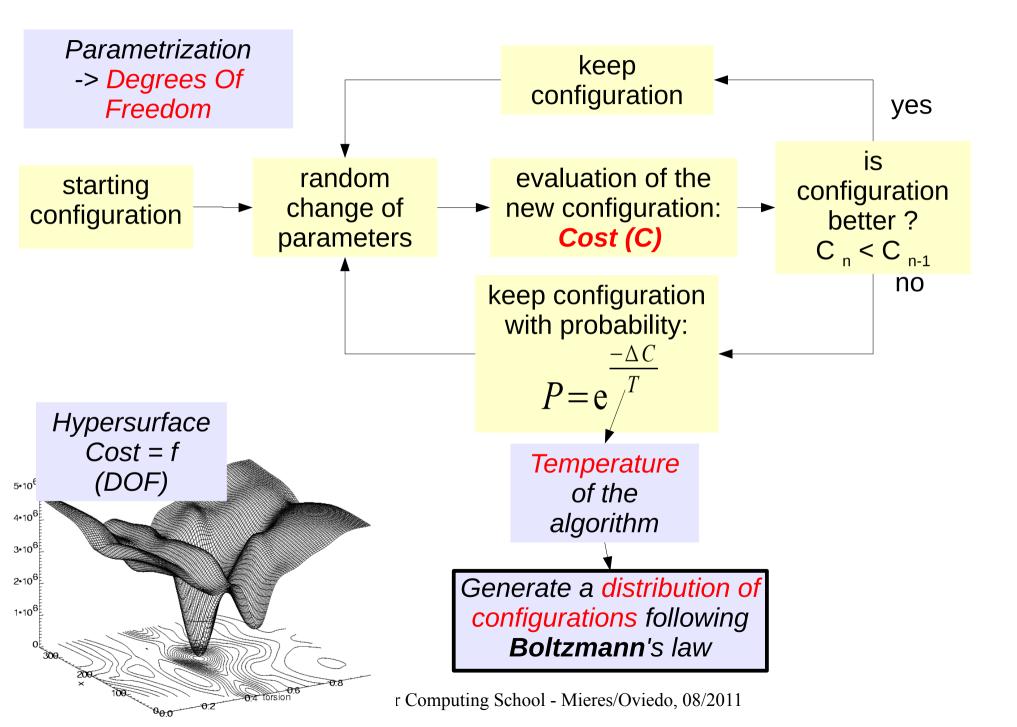
-g7 = bond length C4-C5 =	b7
-g8 = bond angle C3-C4-C5	= т8
-g9 = torsion angle C2-C3-C4-C5	= Θ9

#### 4.Build C6

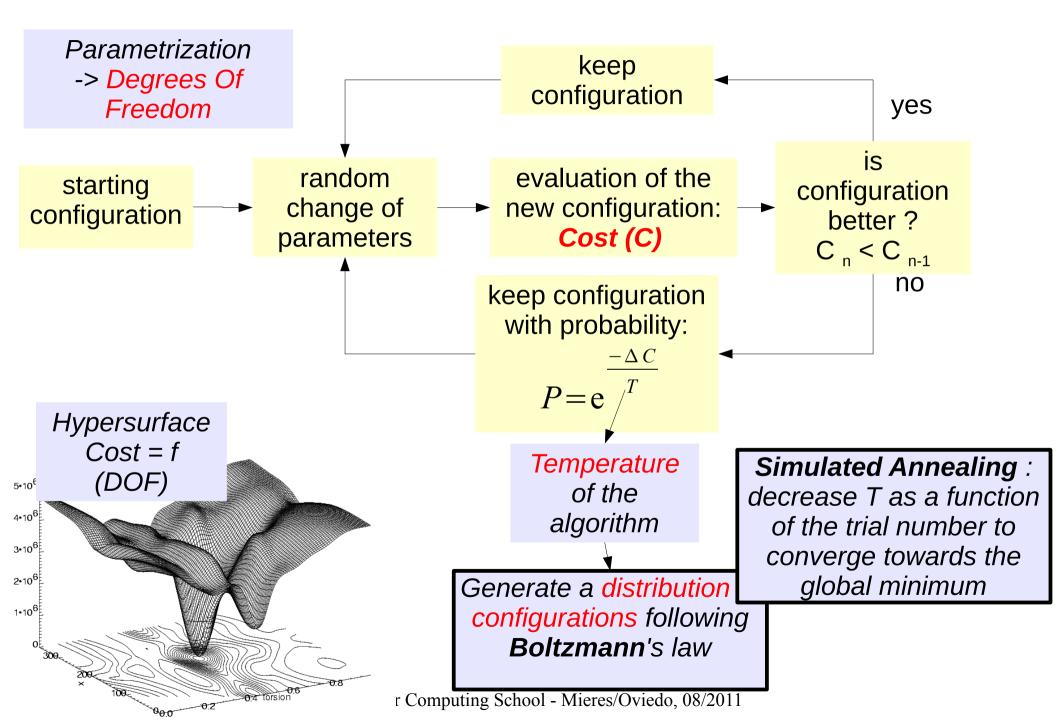
-g10 = bond length C1-C6 = b10 -g11 = bond length C5-C6 = b11 -g12 = bending angle C1-C5-C2 =  $\varphi$ 12

# Ab Initio, Ergodic Minimization Algorithms: Simulated Annealing & co

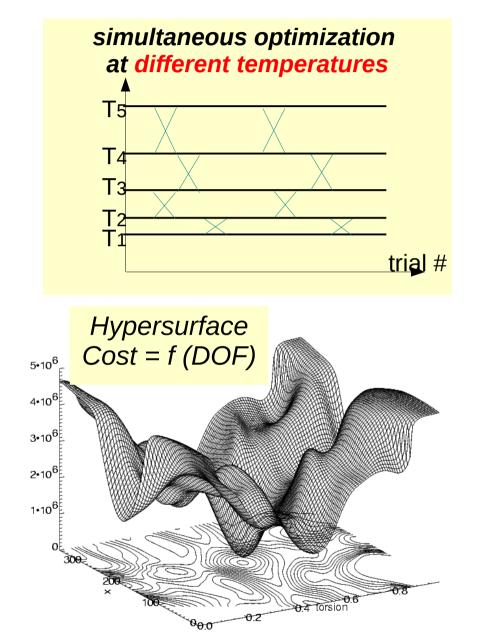
# **Reverse Monte-Carlo**



# **Reverse Monte-Carlo**



# Parallel Tempering & Annealing Temperatures



Using several parallel optimizations at different temperatures ensures that **the algorithm can get out of any local minimum**.

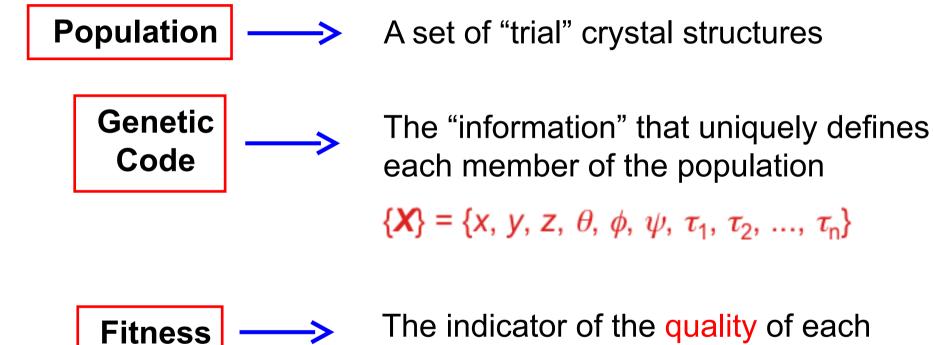
Furthermore, it does not require to predict an adequate decrease rate for the temperature.

To automatically choose the temperatures, in each parallel optimization it is the average atomic displacement per random move which is imposed, from 0.01 to 1 Å. The Temperature is then tuned so that in each "world" the acceptance rate of new configurations is from 10 to 30%.

# Ab Initio, Ergodic Minimization Algorithms: Genetic/Evolutionary

# **The Genetic Algorithm**

Technique for Global Optimization Based on the Principles of Evolution through Natural Selection

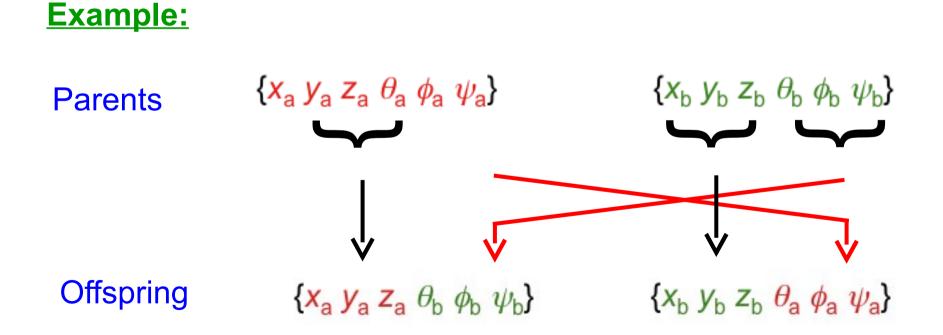


The indicator of the quality of each member of the population

Fitness =  $F(R_{wp})$ 



- Select a pair of structures (parents) from the population
- Swap parts of the genetic codes of the two parents to generate two new structures (offspring)



- Select a structure at random from the population
- Introduce random changes to parts of its genetic code to generate a new structure (mutant)

Example: {x, y, z, 
$$\theta$$
,  $\phi$ ,  $\psi$ ,  $\tau_1$ ,  $\tau_2$ ,  $\tau_3$ }  
Introduce Random Changes to  
Randomly Selected Variables  
{x,  $y^R$ , z,  $\theta^R$ ,  $\phi$ ,  $\psi$ ,  $\tau_1$ ,  $\tau_2$ ,  $\tau_3^R$ }

The new values  $y^R$ ,  $\theta^R$  and  $\tau_3^R$  could either be:

- New random values, or
- Random displacements from the previous values



The Genetic Algorithm produces a sequence of generations of the population through the processes of mating, mutation and natural selection



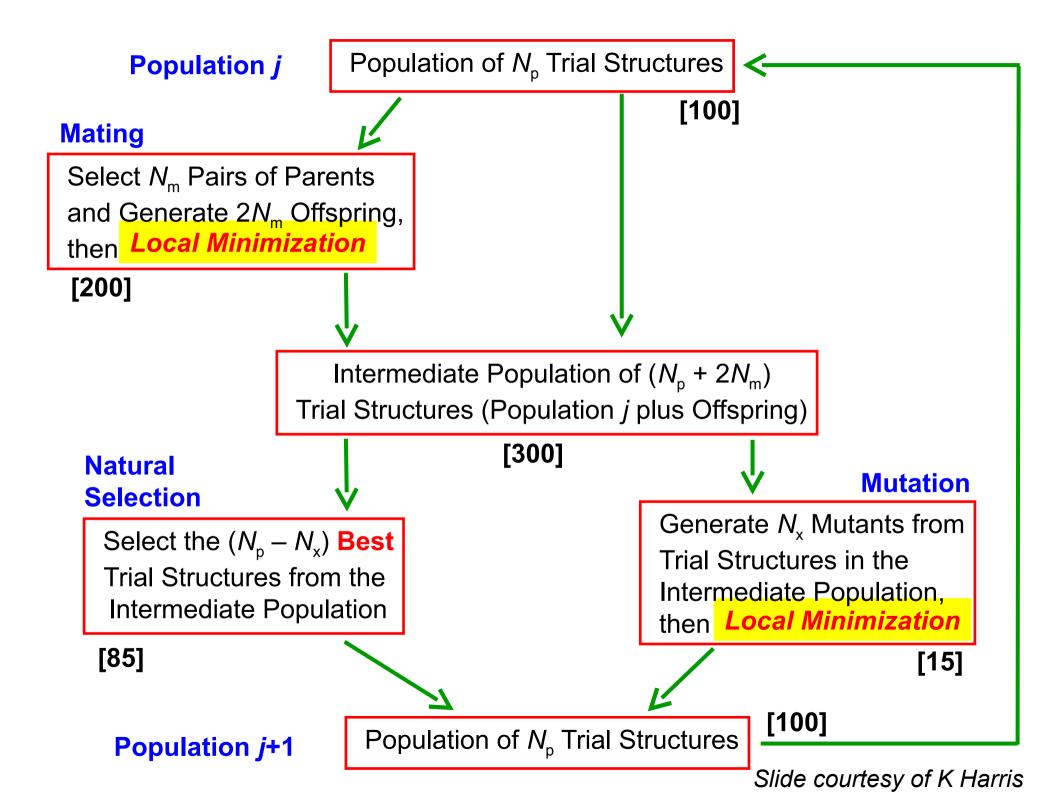
Only the best (highest fitness) structures are allowed to pass from one generation to the next generation



Each new structure created during the calculation (by mating or mutation) is subjected to local minimization of  $R_{wp}$ 

Lamarckian Evolution

G.W. Turner, E. Tedesco, K.D.M. Harris, R.L. Johnston, B.M. Kariuki, *Chem. Phys. Lett.*, **2000**, 321, 183.



# The Genetic Algorithm Technique for Structure Solution

#### **Basics:**

B.M. Kariuki, H. Serrano-González, R.L. Johnston, K.D.M. Harris, *Chem. Phys. Lett.*, **1997**, *280*, 189.

K.D.M. Harris, R.L. Johnston, B.M. Kariuki, Acta Crystallogr., 1998, A54, 632.

#### Some Developments:

G.W. Turner, E. Tedesco, K.D.M. Harris, R.L. Johnston, B.M. Kariuki, *Chem. Phys. Lett.*, **2000**, 321, 183.

S. Habershon, K.D.M. Harris, R.L. Johnston, J. Comp. Chem., 2003, 24, 1766.

S. Habershon, E.Y. Cheung, K.D.M. Harris, R.L. Johnston, *Chem. Phys. Lett.*, **2004**, *390*, 394.

Z. Zhou, V. Siegler, E.Y. Cheung, S. Habershon, K.D.M. Harris, R.L. Johnston, *ChemPhysChem*, **2007**, *8*, 650.

# Genetic Algorithms: **Differential Evolution**

 $\{x, y, z, \theta, \phi, \psi, \tau_1, \tau_2, \tau_3\}$ 

Ex: Uniform Crossover (DE/rand/1/bin)

 Crossover vector generated from randomly selected members of population.

Crossover Vector = Random 3 + F\*(Random 2 - Random 1)

 Exchange elements of a crossover vector and parents with probability of  $C_R$ .

#### Differential Evolution advantagess :

- simple parametrization (F, CR)
  robust algorithms (
  Yields not 1 but a full population close to the solution

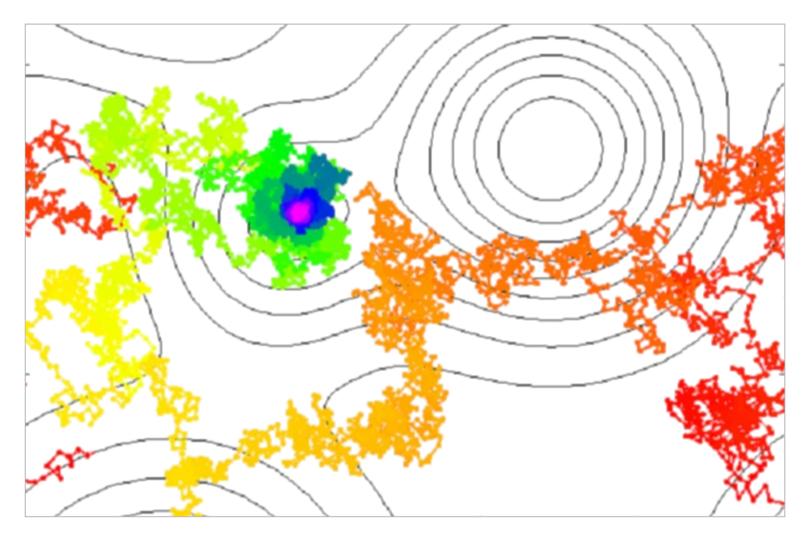
Tremayne & Seaton Acta Cryst. B58 (2002): 823-834.

IUCr Computing School - Mieres/Oviedo, 08/2011

C. Seaton / MJ Tremayne

# Ab Initio, Ergodic Minimization Algorithms: Hybrid Monte-Carlo

# Monte-Carlo Algorithm : Inefficient ?



Path followed during several SA runs : Brownian, « drunken » walk

IUCr Computing School - Mieres/Oviedo, 08/2011

B. David/A Markvardsen

# Molecular Dynamics + Monte-Carlo =Hybrid Monte Carlo

$$H(t) = \frac{1}{2} \sum_{i=1}^{N} m_i v_i^2(t) + U(\mathbf{r}(t))$$
$$H(t) = \frac{1}{2} \sum_{i=1}^{N} p_i^2(t) + \chi^2(\mathbf{r}(t))$$

a "particle" sitting on a potential energy surface

... obeying Hamilton's equations of motion

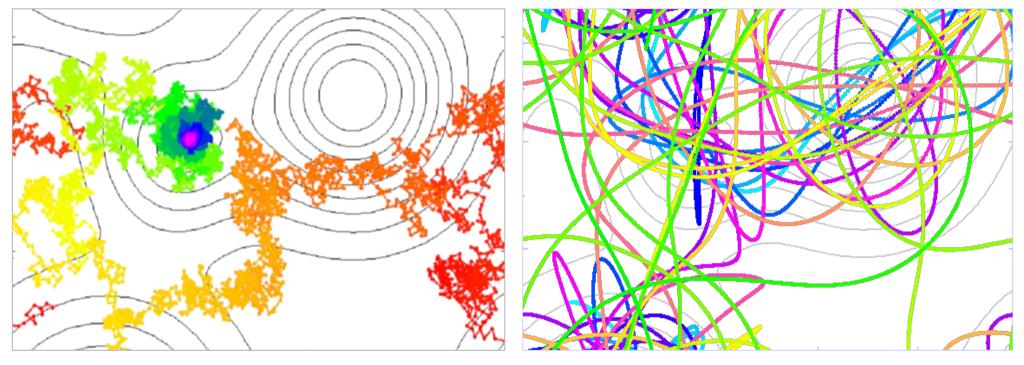
$$\frac{\partial r_i}{\partial t} = \frac{\partial H}{\partial p_i} = p_i, \qquad i = 1, \dots, N$$
$$\frac{\partial p_i}{\partial t} = -\frac{\partial H}{\partial r_i} = -\frac{\partial \chi^2}{\partial r_i}, \qquad i = 1, \dots, N$$

Molecular dynamics vs. simulated annealing

"downhill skiing" compared with "drunken walk"

Acta Cryst. **A58** (2002),441-447 J. Appl. Cryst. **38** (2005), 107-111.

# Monte-Carlo Algorithm : Inefficient ?



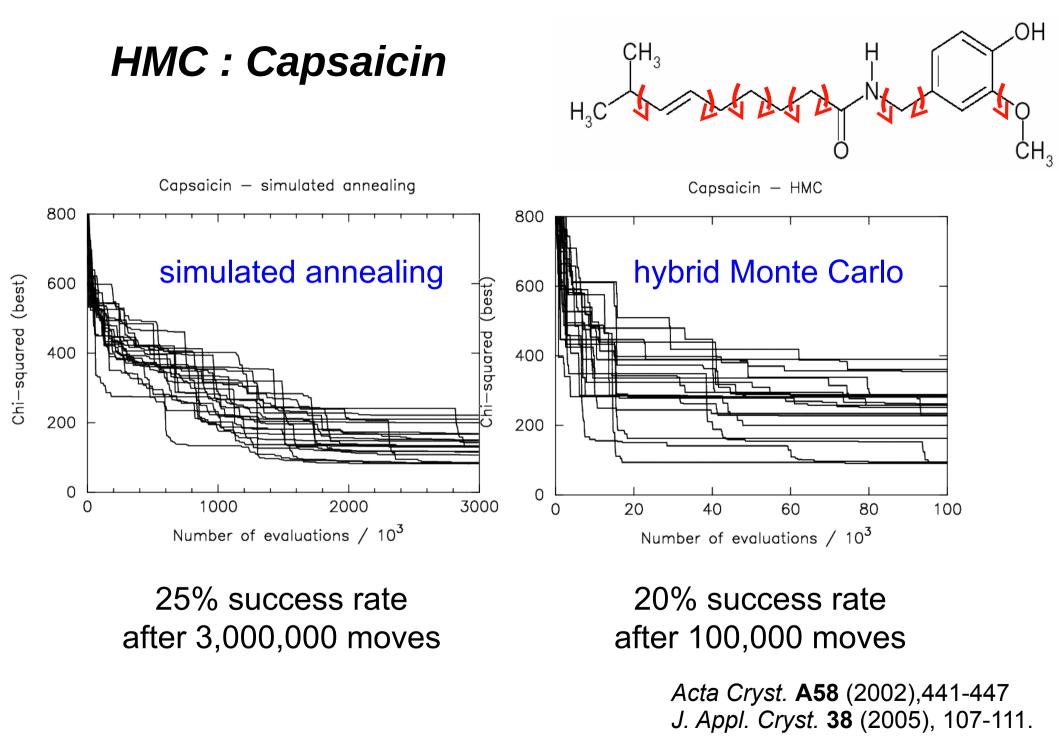
Simulated Annealing

Hybrid MC

*Acta Cryst.* **A58** (2002),441-447 *J. Appl. Cryst.* **38** (2005), 107-111.

2011/08/19

IUCr Computing School - Mieres/Oviedo, 08/2011 B. David/A Markvardsen

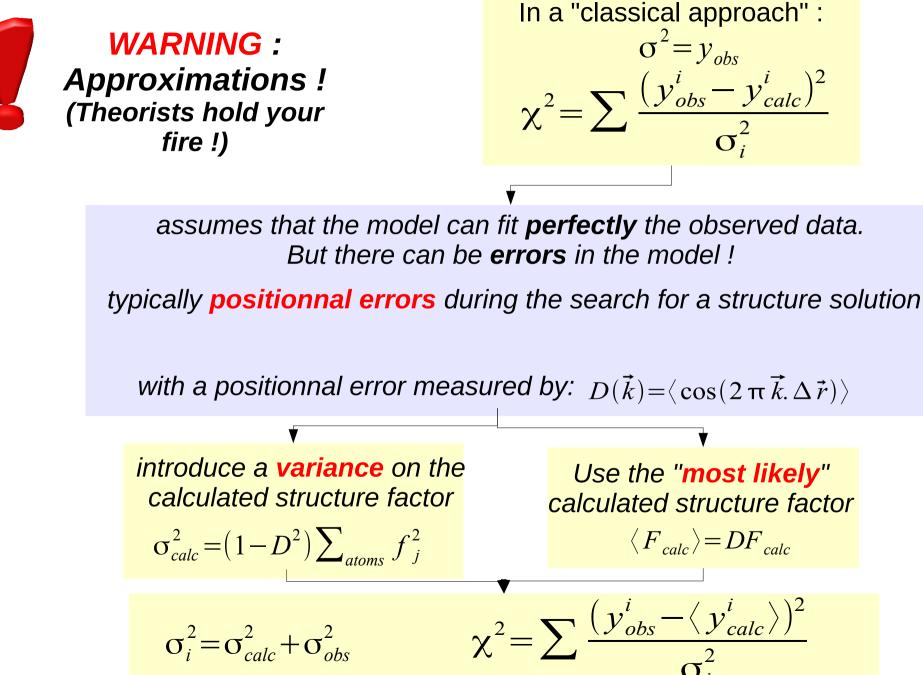


IUCr Computing School - Mieres/Oviedo, 08/2011 B David/

B. David/A Markvardsen

# Maximum Likelihood & Global Optimization

## Maximum Likelihood



2011/08/19

1001 Computing School - Mileres/Ovieuo, vo/20

### Application to Global Optimization

1<sup>st</sup> application: incomplete model

missing atoms (H's, solvant) do not contribute to the **Structure Factor** but increase the **variance** 

$$D(\vec{k}) = \langle \cos(2\pi \vec{k} \Delta \vec{r}) \rangle = 0$$

$$\langle F_{calc} \rangle = DF_{calc} = 0$$

$$\sigma_{calc}^{2} = (1 - D^{2}) \sum_{atoms} f_{j}^{2}$$
Markvardson Acta Cryst

A58(2002)

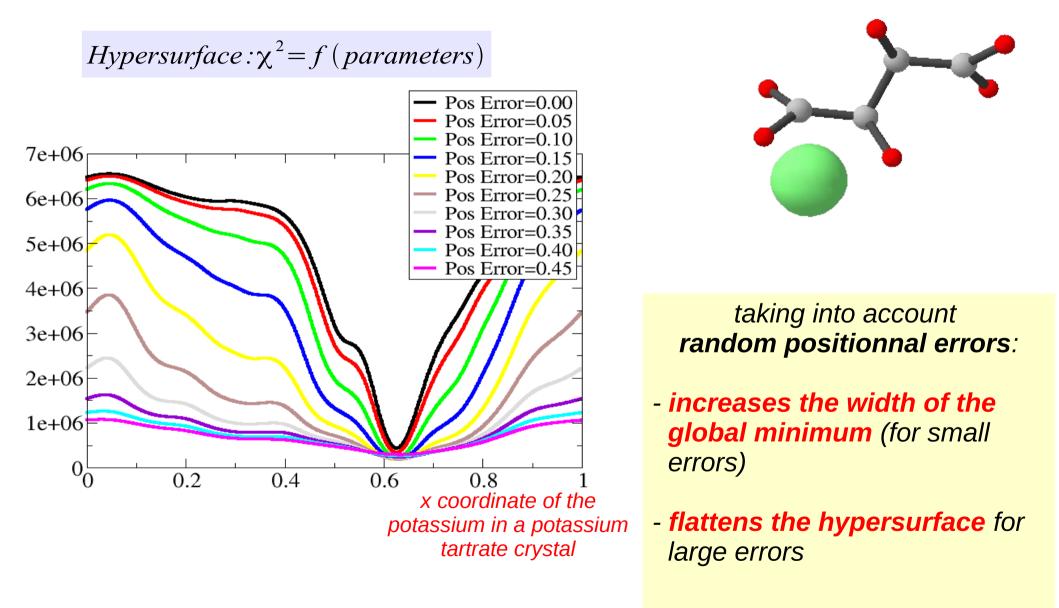
2<sup>nd</sup> application: model errors Atoms are always misplaced

Atoms are always misplaced during a global optimisation

taking into account **random positionnal errors** should yield a **better agreement between the incorrect model and the observed diffraction data** 

can it help its **convergence**?

# Hypersurface as a function of positionnal error

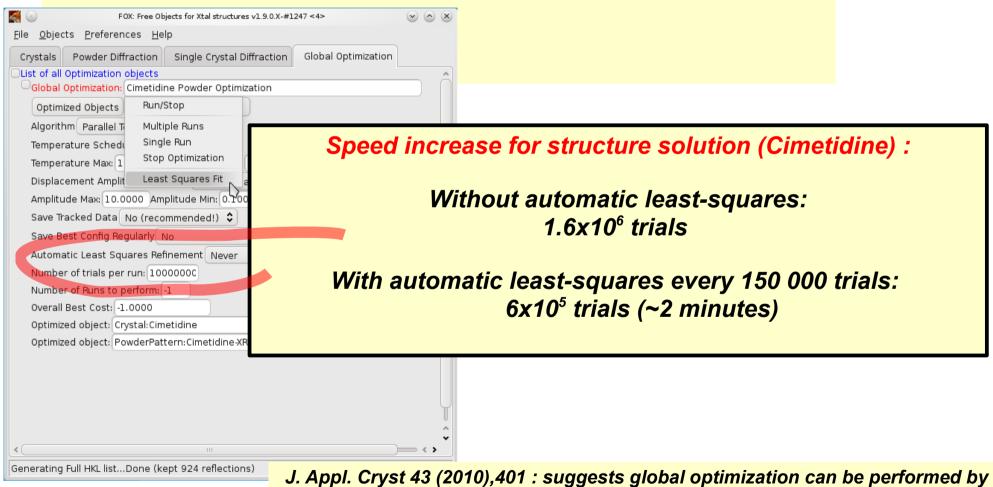


# Fox/ObjCryst++

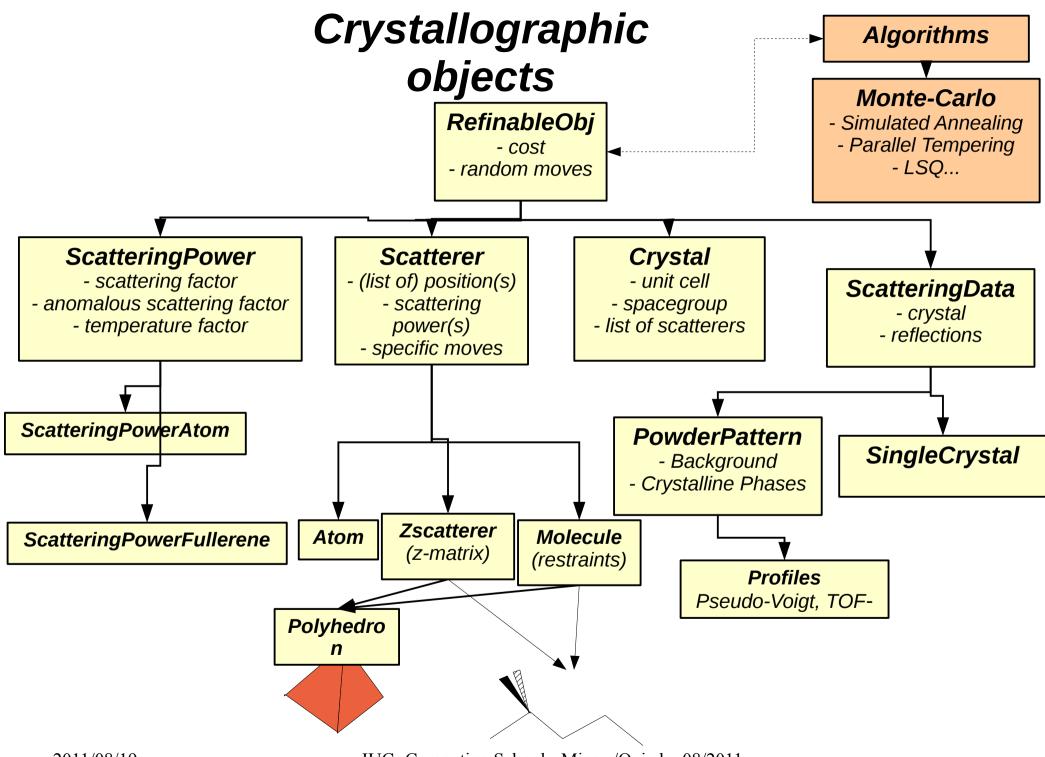
#### Least squares

Least squares refinement can be performed:

- For profile fitting
- After optimization (only the structure is refined, no parameter choice)
- Automatically during optimization



doing only downhill minimization from random starting points...



2011/08/19

# **API Documentation**

API documentation must be written along code / headers... or it will never be (who likes documenting ?)

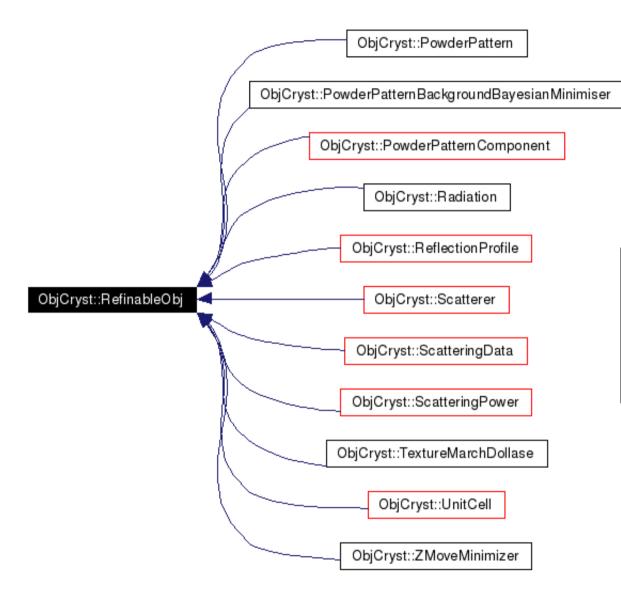
=> Use doxygen (<u>http://doxygen.org</u>): supports C++, Python,...

#### **Public Member Functions**

FOX

East Hama Daga (wiki)	
Fox Home Page (wiki)	ReflectionProfile ()
SourceForge Project	ReflectionProfile (const ReflectionProfile &old)
About FOX	virtual ~ReflectionProfile ()
Download	virtual ReflectionProfile * CreateCopy () const=0
Install	virtual CrystVector_REAL GetProfile (const CrystVector_REAL &x, const REAL xcenter, const REAL h, const REAL k, const REAL I)
Screenshots	Get the reflection profile.
Biblio: Fox References	
Biblio: Structures solved	virtual REAL GetFullProfileWidth (const REAL relativeIntensity, const REAL xcenter, const REAL h, const REAL k, const
Mailing List	Get the (approximate) full profile width at a given percentage of the profile maximum (e.g.
FAQ	virtual bool IsAnisotropic () const ObjCryst::RefinableObjClock
Using FOX	Is the profile anisotropic ?
Tutorials	mRefParListClock mClock mClock
FOX Manual (intro.)	mClockMaster mClock
- Crystal Structures	
- Powder Diffr. data	mClockLatticePar
- Single Crystal data	mClockLatticeParUpdate ObjCryst::RefinableObj ObjCryst::RefObjOpt ObjCryst::SpaceGroup
- Ontimication Alan	
FOX Development Current Development	From the Fox wiki (http://objcryst.sf.net ): mConstrainLatticeToSpaceGroup _mSpaceGroup
Features Requests	- browse the code
	- access the API (code) documentation
ObjCryst++ API	
Getting FOX from SVN	
Browse Code Repository	
2011/08/19	IUCr Computing School - Mieres/Oviedo, 08/2011

# RefinableObj Hierarchy



Each object has a **specialized cost** (Chi^2, bond/angle restraints, etc), and a **function for "random change"** of its parameters

> http://objcryst.sf.net => API doc

# **RefinableObj and Algorithms**

#### " Refinable Objects " : all objects derive from this class:

- unit cell, crystal, atoms, molecule, diffraction data, background, etc...

#### They share the same functions (inheritance):

- name of the object
- name, value and limits of the object's parameters
- cost function(s) (e.g. c<sup>2</sup>, antibump, bond valence...)
- function to generate a random move with a given amplitude
- ability to save & restore a "set" of parameters

#### A "tree" of refinable objects is used by the algorithm

- e.g. a Crystal includes atoms, molecules, scattering powers,...
- modification of parameters triggers "clocks" in each object, recursively (bottom-up)
- each computation (profile, scattering factor, etc..) is only redone if necessary

#### The algorithms have no clue about the exact nature of the refined objects:

- algorithm and structure / diffraction data are 100% independent
- any sort of refinable object can be optimized by the algorithms

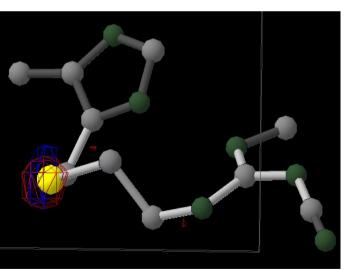
....

#### Rules to find a structure solution: check multiple solutions

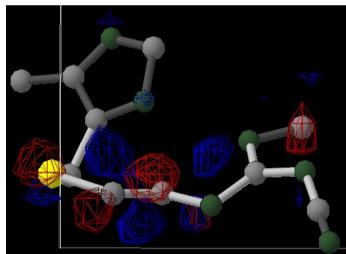
Look at multiple solutions => estimate confidence in " solution " 1) Compare the  $\chi^2$  and Rwp

2) Use **Fourier Difference** Maps to check differences (requires at least 1.5Å resolution data)

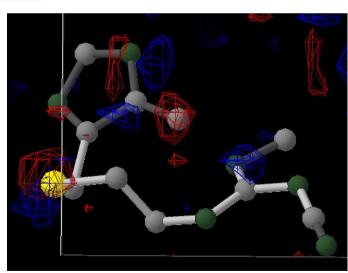
Use the same contours for all solutions Fo-Fc, +1 and -1 contours



**Correct** 



Wrong conformation of internal chain



Wrong position for side CH<sub>3</sub> group

#### Rules to find a structure solution: be a flexible User

**Restraints** must be used to reduce parameter space

... but too many restraints can slow or prevent a structure solution

e.g.: a combination of strong antibump and angular restraints can make very difficult to go from one local minimum to the global one.

Imagine the "molecule" to be solved is: a man & a chair. The "solution" is: the man, <u>sitting</u> on the chair, in the Prado Museum (Madrid)

The random starting location is: Grenoble railway station.

To "speed up " the solution, you impose as much restraints as you can, i.e. " the man must be sitted on the chair at all times "

=> Rigid groups should be used scarcely... and do not generally speed up the convergence

=> if the algorithm " distorts " your molecule during the optimization, it's for your own good (honest !)

=> the correct conformation comes from the data, not the number of restraints => NB: different rules apply if data is of <u>bad</u> quality

#### Rules to find a structure solution: no high-resolution data

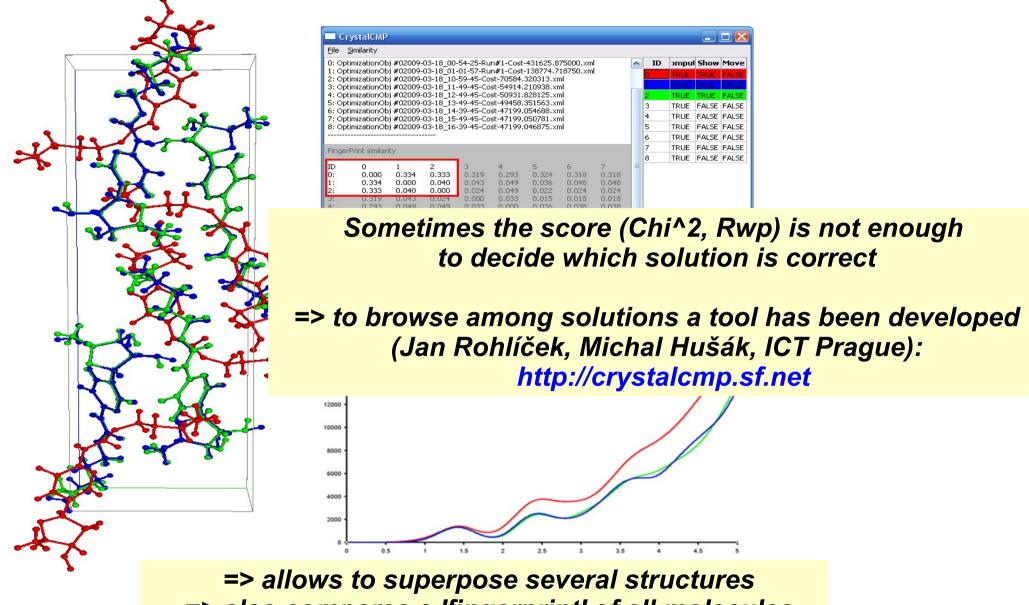
"Solving the structure " means finding all the atomic positions with an error of ~ 0.1Å. => high resolution data (1Å and higher) is not needed... increasing the resolution by 25% doubles the computing time !

Giving high-resolution data is like giving your address to a friend ....by insisting on the **exact pattern of colours from your garden's flowers** ... instead of **just giving the address & colour** of the house

> Most of the time, a 2.5 Å resolution is enough. ...sometimes 1.5Å.

Of course you still need the high-resolution data for the least squares refinement !

## Comparing crystal structures



=> also compares a 'fingerprint' of all molecules

INSTITUTE OF CHEMICAL TECHNOLOGY PRAGUE

#### Challenges ?

- More efficient algorithms (HMC, genetic)
- Solve *flexible* protein structures ab initio (reduced set of param : TLS, soft modes..)
- More efficient exploration for molecules with flexible rings
- Give the abilyty to power-users to choose their own random moves (pyobjcryst)
- Exploit power of GPU for real space structure solution (see my GPU talk)
- More user-friendly grid computing (FOX.Grid, mDASH)
- Simpler validation procedures (especially for inorganic crystals)

# (incomplete) Bibliography

[1] V. B. Zlokazov et V. V. Chernyshev, « MRIA - a program for a full profile analysis of powder multiphase neutron-diffraction time-of-flight (direct and Fourier) spectra », Journal of Applied Crystallography, vol. 25, nº. 3, p. 447-451, juin. 1992.

[2] Y. G. Andreev, P. Lightfoot, et P. G. Bruce, « Structure of the polymer electrolyte poly(ethylene oxide)3: LiN(SO2CF3)2 determined by powder diffraction using a powerful Monte Carlo approach », Chemical Communications, nº. 18, p. 2169-2170, 1996.

[3] B. M. Kariuki, D. M. S. Zin, M. Tremayne, et K. D. M. Harris, « Crystal Structure Solution from Powder X-ray Diffraction Data: The Development of Monte Carlo Methods To Solve the Crystal Structure of the γ-Phase of 3-Chloro-trans-cinnamic Acid† », Chemistry of Materials, vol. 8, nº. 2, p. 565-569, janv. 1996.

[4] M. Tremayne, B. M. Kariuki, K. D. M. Harris, K. Shankland, et K. S. Knight, « Crystal Structure Solution from Neutron Powder Diffraction Data by a new Monte Carlo Approach Incorporating Restrained Relaxation of the Molecular Geometry », Journal of Applied Crystallography, vol. 30, nº. 6, p. 968-974, déc. 1997.

[5] Y. G. Andreev, G. S. MacGlashan, et P. G. Bruce, « Ab initio solution of a complex crystal structure from powder-diffraction data using simulated-annealing method and a high degree of molecular flexibility », Physical Review B, vol. 55, nº. 18, p. 12011, mai. 1997.

[6] Y. G. Andreev et P. G. Bruce, « Solving crystal structures of molecular solids without single crystals: a simulated annealing approach », Journal of the Chemical Society, Dalton Transactions, no. 24, p. 4071-4080, 1998.

[7] W. I. F. David, K. Shankland, K. Shankland, et N. Shankland, « Routine determination of molecular crystal structures from powder diffraction data », Chemical Communications, nº. 8, p. 931-932, 1998.

[8] K. D. M. Harris, R. L. Johnston, et B. M. Kariuki, « The Genetic Algorithm: Foundations and Apllications in Structure Solution from Powder Diffraction Data », Acta Crystallographica Section A Foundations of Crystallography, vol. 54, nº. 5, p. 632-645, sept. 1998.

[9] V. V. Chernyshev et H. Schenk, « A grid search procedure of positioning a known molecule in an unknown crystal structure with the use of powder diffraction data », Zeitschrift für Kristallographie, vol. 213, nº. 1, p. 1-3, janv. 1998.

[10] H. Putz, J. C. Schön, et M. Jansen, « Combined method for ab initio structure solution from powder diffraction data », Journal of Applied Crystallography, vol. 32, nº. 5, p. 864-870, oct. 1999. [11] M. Falcioni et M. W. Deem, « A biased Monte Carlo scheme for zeolite structure solution », The Journal of Chemical Physics, vol. 110, nº. 3, p. 1754-1766, janv. 1999.

[12] A. J. Markvardsen, W. I. F. David, et K. Shankland, « A maximum-likelihood method for global-optimization-based structure determination from powder diffraction data », Acta Crystallographica Section A Foundations of Crystallography, vol. 58, nº. 4, p. 316-326, juin. 2002.

[13] J. C. Johnston, W. I. F. David, A. J. Markvardsen, et K. Shankland, « A hybrid Monte Carlo method for crystal structure determination from powder diffraction data », Acta Crystallographica Section A Foundations of Crystallography, vol. 58, n°. 5, p. 441-447, août. 2002.

[14] M. Tremayne, C. C. Seaton, et C. Glidewell, « Structures of three substituted arenesulfonamides from X-ray powder diffraction data using the differential evolution technique », Acta Crystallographica Section B Structural Science, vol. 58, nº. 5, p. 823-834, sept. 2002.

[15] D. Albesa-Jové, B. M. Kariuki, S. J. Kitchin, L. Grice, E. Y. Cheung, et K. D. M. Harris, « Challenges in Direct-Space Structure Determination from Powder Diffraction Data: A Molecular Material with Four Independent Molecules in the Asymmetric Unit », ChemPhysChem, vol. 5, nº. 3, p. 414-418, mars. 2004.

[16] V. Favre-Nicolin et R. ?erny, « A better FOX: using flexible modelling and maximum likelihood to improve direct-space ab initio structure determination from powder diffraction », Zeitschrift fur Kristallographie, vol. 219, nº. 12/2004, p. 847-856, déc. 2004.

[17] K. D. M. Harris, S. Habershon, E. Y. Cheung, et R. L. Johnston, « Developments in genetic algorithm techniques for structure solution from powder diffraction data », Zeitschrift fr Kristallographie, vol. 219, nº. 12/2004, p. 838-846, déc. 2004.

[18] A. J. Markvardsen, K. Shankland, W. I. F. David, et G. Didlick, « Characterization of a hybrid Monte Carlo search algorithm for structure determination », Journal of Applied Crystallography, vol. 38, nº. 1, p. 107-111, janv. 2005.

[19] A. J. Florence et al., « Solving molecular crystal structures from laboratory X-ray powder diffraction data with DASH : the state of the art and challenges », Journal of Applied Crystallography, vol. 38, nº. 2, p. 249-259, mars. 2005.

[20] W. I. F. David, K. Shankland, J. van de Streek, E. Pidcock, W. D. S. Motherwell, et J. C. Cole, « DASH

: a program for crystal structure determination from powder diffraction data », Journal of Applied Crystallography, vol. 39, nº. 6, p. 910-915, nov. 2006.

[21] W. I. F. David et K. Shankland, « Structure determination from powder diffraction data », Acta Crystallographica Section A Foundations of Crystallography, vol. 64, n°. 1, p. 52-64, déc. 2007. [22] A. Immirzi, L. Erra, et C. Tedesco, « Finding crystal structures of peptides by random search and evolutionary algorithms », Journal of Applied Crystallography, vol. 42, n°. 5, p. 810-814, août. 2009.

[23] Z. J. Feng, C. Dong, R. R. Jia, X. D. Deng, S. X. Cao, et J. C. Zhang, « PeckCryst

: a program for structure determination from powder diffraction data using a particle swarm optimization algorithm », Journal of Applied Crystallography, vol. 42, nº. 6, oct. 2009.

[24] A. Rapallo, « VARICELLA: A variable-cell direct space method for structure determination from powder diffraction data », The Journal of Chemical Physics, vol. 131, nº. 4, p. 044113-16, juill. 2009.

[25] S. Pagola et P. W. Stephens, « PSSP

, a computer program for the crystal structure solution of molecular materials from X-ray powder diffraction data », Journal of Applied Crystallography, vol. 43, nº. 2, p. 370-376, mars. 2010. [26] K. Shankland, A. J. Markvardsen, C. Rowlatt, N. Shankland, et W. I. F. David, « A benchmark method for global optimization problems in structure determination from powder diffraction data », Journal of Applied Crystallography, vol. 43, nº. 3, p. 401-406, mars. 2010.

# Tutorial(s)

1) Programming (python+numpy) : basic test of algorithms (simulated annealing + differential evolution), no 'real crystallography

2) (if interest) : demonstration & test of FOX ( http://objcryst.sf.net) and (informal) discussion about the internals of the algorithms

PS: anybody interested for a fast python script for triclinic powder patten indexing (dichotomy) ?