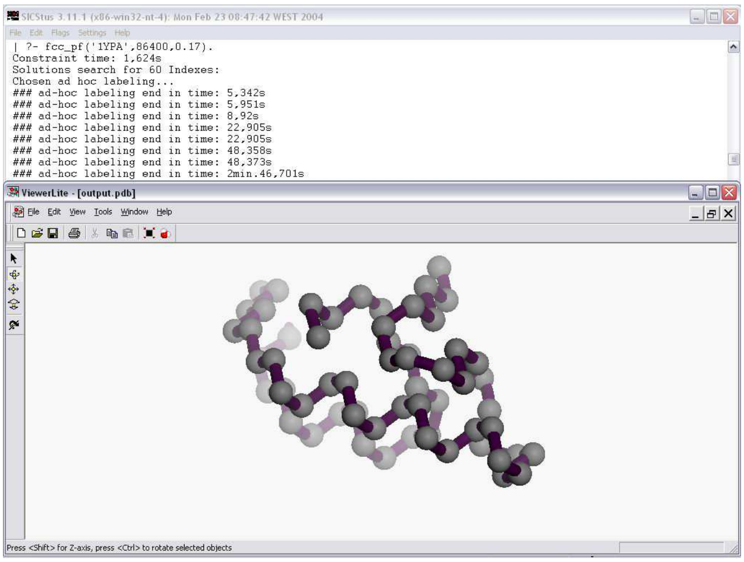
Two constraint-based tools for protein folding

Luca Bortolussi, Alessandro Dal Palù, and Agostino Dovier

Dip. di Matematica e Informatica, Univ. di Udine
Via delle Scienze 206, 33100 Udine (Italy).
(bortolus|dalpalu|dovier)@dimi.uniud.it

Introduction. A protein is a list of linked units called aminoacids. There are 20 different kinds of aminoacids and the typical length of a protein is less than 500 units. The Protein Structure Prediction Problem (PSP) is the problem of predicting the 3D native conformation of a protein, when its aminoacid sequence is known. The process for reaching this state is called the *protein folding.* It is widely accepted that the native conformation ensures a state of minimum free energy [1]. We assume some energy functions previously defined and we focus on the problem of finding the 3D conformation that minimizes them. We present two tools developed following different approaches to this problem. In the first we use Constraint Logic Programming over finite domains applied on the modeling of the Protein Structure Prediction problem on the Face-Centered Cubic Lattice [4]. In the second we develop a high-level off-lattice simulation method which makes use of Concurrent Constraint Programming [3]. The two codes are available at http://www.dimi.uniud.it/dovier/PF.

CLP($\mathcal{FD}$) minimization. A non-linear minimization problem can be easier to solve when the solution's space is finite. In this context, this can be done by setting admissible aminoacid's positions as the vertices of a lattice. We use the so-called *Face-Centered Cubic Lattice* [7], which is a good discrete model for protein's conformations. We look for the protein conformation in the lattice that minimizes a function which is the sum of the contributions of all pairs of aminoacids. Each contribution is non-zero only if two aminoacids are under a certain lattice distance and the precise value depends on their type as described in [2]. The tool is written in SICStus Prolog, using the `clpfd` library and it is based on [4]. We have added constraints obtained by secondary structure prediction (prediction of some local conformations, such as helices and sheets), which are currently very accurate, to reach acceptable computation time. We have also developed a local coordinate system to define torsional angles, which allows to link efficiently the secondary structure information to the three-dimensional folding. Moreover, we have implemented a method to dynamically prune the search tree based on the analysis of the contacts between the aminoacids during the folding process. The results we obtained allow us to effectively predict proteins up to 60 aminoacids. The actual version of the tool is much faster that the first version presented in [5]: for some proteins we have reached a speed up of more than 200 times. Anyway, time is still considerable, as can be expected from the NP completeness of the problem. Proteins of length up to 20 are folded correctly in few seconds, while for longer proteins (around 40 aminoacids) the optimal solution is reached in 3 to 10 hours. Proteins of length 60 take longer time, even if acceptable solutions are found in about 10 hours (on a PC, 3 GHz, 512MB). The user can choose the maximum search time and he/she can prune the search tree imposing a "compact" coefficient that acts on the allowed 3D structure to the protein. In figure 1 it is shown the tool while working on protein 1YPA of length 63, with time limit of 24h (86400s) and compact factor of 0.17. The solution is found in 14 hours and saved on a standard "pdb" file viewed using the program ViewerLite.



SICStus 3.11.1 (x86-win32-nt-4): Mon Feb 23 08:47:42 WEST 2004
File Edit Flags Settings Help

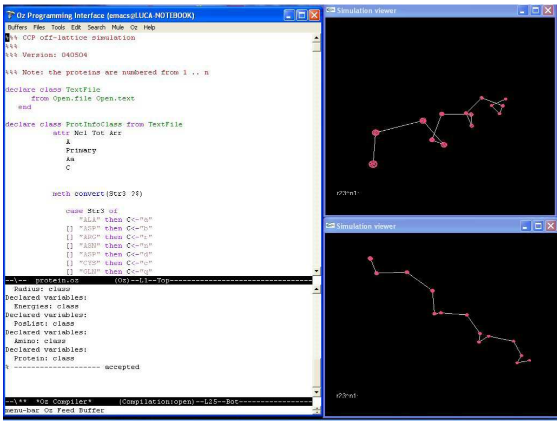
```
| ?- fcc_pf('1YPA',86400,0.17).
Constraint time: 1,624s
Solutions search for 60 Indexes:
Chosen ad hoc labeling...
### ad-hoc labeling end in time: 5,342s
### ad-hoc labeling end in time: 5,951s
### ad-hoc labeling end in time: 8,92s
### ad-hoc labeling end in time: 22,905s
### ad-hoc labeling end in time: 22,905s
### ad-hoc labeling end in time: 48,358s
### ad-hoc labeling end in time: 48,373s
### ad-hoc labeling end in time: 2min.46,701s
```

ViewerLite - [output.pdb]
File Edit View Tools Window Help
Press <Shift> for Z-axis, press <Ctrl> to rotate selected objects

Fig. 1. CLP($\mathcal{FD}$) minimizator

CCP simulation. In this tool, described in [3], we adopt an off-lattice simplified representation of a protein, where each aminoacid is represented by a center of interaction. The empirical contact energy function [2] used in the constraint-based approach is modified and augmented by local terms which describe bond lengths, bend angles, and torsion angles. Our simulation makes use of concurrent constraint programming. Basically, each aminoacid of the protein is viewed as an independent agent that moves in the space and communicates with other aminoacids. Each agent waits for a communication of the modification of other aminoacids' position; after receiving a message, it stores the information in a list and performs a move. The new position is computed using a Montecarlo simulation, based on the spatial information available to the aminoacid, which may not be the current dislocation of the protein, due to asynchrony in the communication. Once the move is performed, the aminoacid communicates its new position to all the others. The code has been implemented in Mozart [6].

We tested our system either on known proteins or on artificial sequences of aminoacids. The code works properly on sequences of Alanines, that are known to have a high tendency to form a single helix, while for more complex structures the minima not always corresponds to the real native conformation. In Figure 2 we show the tool while folding a helix from a sequence of 14 Alanines. As initial state of the protein we set each aminoacid along a line with a step of the bond distance (3.8 Å). We run the simulation for 60 seconds on a PC, 1GHz, 256MB.


Oz Programming Interface (emacs@LUCA-NOTEBOOK)
Buffers Files Tools Edit Search Mule Oz Help
Simulation viewer
r23^n1
Simulation viewer
r23^n1

```
%%% CCP off-lattice simulation
%%%
%%% Version: 040504

%%% Note: the proteins are numbered from 1 .. n

declare class TextFile
      from Open.file Open.text
   end

declare class ProtInfoClass from TextFile
           attr Ncl Tot Arr
              A
              Primary
              Aa
              C


          meth convert(Str3 ?$)

              case Str3 of
                 "ALA" then C<-"a"
              [] "ASP" then C<-"b"
              [] "ARG" then C<-"r"
              [] "ASN" then C<-"n"
              [] "ASP" then C<-"d"
              [] "CYS" then C<-"c"
              [] "GLN" then C<-"q"
--\--  protein.oz        (Oz)--L1--Top-----------------------------------
  Radius: class
Declared variables:
  Energies: class
Declared variables:
  PosList: class
Declared variables:
  Amino: class
Declared variables:
  Protein: class
% -------------------- accepted


--\**  *Oz Compiler*      (Compilation:open)--L25--Bot----------------
menu-bar Oz Feed Buffer
```

Fig. 2. CCP Simulator.

References

1. C. B. Anfinsen. Principles that govern the folding of protein chains. *Science*, 181:223–230, 1973.
2. M. Berrera, H. Molinari, and F. Fogolari. Amino acid empirical contact energy definitions for fold recognition in the space of contact maps. *BMC Bioinformatics*, 4(8), 2003.
3. L. Bortolussi, A. Dal Palù, A. Dovier, and F. Fogolari. Protein Folding Simulation in CCP. Submitted to BIOCONCUR 2004.
4. A. Dal Palù, A. Dovier, and F. Fogolari. Protein folding in $CLP(\mathcal{FD})$ with empirical contact energies. In *Recent Advances in Constraints*, volume 3010 of *Lecture Notes in Artificial Intelligence*, pp. 250–265, 2004.
5. A. Dovier, M. Burato, and F. Fogolari. Using secondary structure information for protein folding in $clp(\mathcal{FD})$. *Proceedings of the 11th International Workshop on Functional and (Constraint) Logic Programming*, 76, 2002.
6. Univ. des Saarlandes, Sweedish Inst. of Computer Science, and Univ. Catholique de Louvain. The Mozart Programming System. www.mozart-oz.org.
7. G. Raghunathan and R. L. Jernigan. Ideal architecture of residue packing and its observation in protein structures. *Protein Science*, 6:2072–2083, 1997.