A NEW SMOOTHING-BASED GLOBAL OPTIMIZATION ALGORITHM FOR PROTEIN CONFORMATION PROBLEMS *

AQIL M AZMI, † AND RICHARD H BYRD, ELIZABETH ESKOW, ROBERT B SCHNABEL. ‡

Abstract. To help solve difficult global optimization problems such as those arising in molecular chemistry, smoothing the objective function has been used with some efficacy. In this paper we propose a new approach to smoothing. First, we propose a simple algebraic way to smooth the Lennard-Jones and the electrostatic energy functions. These two terms are the main contributors to the energy function in many molecular models. The smoothing scheme is much cheaper than the classic spatial averaging smoothing techniques. In computational tests on the proteins polyalanine with up to 58 amino acids and metenkephalin, smoothing is very successful in finding the lowest energy structures. The largest case of polyalanine is particularly significant because the lowest energy structures that are found include ones that exhibit interesting tertiary as opposed to just secondary structure.

Key words. Global Optimization, Molecular Chemistry, Polyalanine, Metenkephalin, Smoothing Techniques, Protein Folding

1. Introduction. The topic of this paper is the development of new smoothing methods for large-scale global optimization problems which frequently arise in molecular chemistry applications. The prediction of molecular conformation by energy minimization gives rise to very difficult global optimization problems. To aid in solving these large problems, both the chemistry and the optimization communities have relied on different techniques, smoothing being one of these techniques. Our overall goal is to develop new smoothing techniques that are both inexpensive and effective in the molecular context, and to integrate them into sophisticated global optimization algorithms. This paper is an extension of an earlier work that dealt with smoothing for molecular cluster configuration problems [24].

The protein folding problem is defined in [2] as the problem of finding the native state of a protein in its normal physiological milieu. In other words, we want to determine the three-dimensional structure of a protein, called its "tertiary structure," just from the sequence of amino acids that it is composed of (its "primary structure"). Under the assumption that in the native state the potential energy of a protein is globally minimized, the protein folding problem can be regarded as equivalent to solving the
problem

$$\min_{x \in \mathbb{R}^n} E(x)$$

(1.1)

where $E(x)$ is the value of the potential energy function (2.5) for a configuration of an $n$ atom protein described by the $3n$ dimensional vector $x$.

To handle this problem, an optimization algorithm has to solve problems with many variables, since even the smallest proteins have a large number of free variables. Apart from that, the potential energy function is known to have many local minimizers. For proteins, it is speculated that the potential energy function has at least $3^n$ local minima, $n$ being the number of free variables [15] [28]. Thus, the protein folding problem belongs to the class of NP-hard problems.

A large scale global optimization algorithm that does not utilize the solution structure of the cluster has been developed by some of the authors in the past few years [8]. It has been successfully used to solve Lennard-Jones molecular cluster problems with up to 76 atoms as well as more complex water cluster problems of up to 21 molecules. However, in practice, it is expected that it will generally be too expensive – if not impossible – to solve problems of such size, such as polyalanine, by using the same global optimization algorithm directly. In fact [26] was unsuccessful in solving for polyalanine $N > 40$ using the global optimization algorithm of [8]. This realization motivates approaches that seek to improve the effectiveness of global optimization algorithms via transformations of the objective function.

One transformation method is to use a parameterized set of smoothed objective functions. The smoothed functions are intended to retain coarse structure of the original objective function, but have fewer local minima. By selecting different smoothing parameters, objective functions with different degrees of smoothness can be derived. However, it is quite possible that as we vary smoothing parameters, the trajectory from the global minimizer of the smoothed problem will not lead to the global minimizer of the original problem. Indeed, the later situation which we term order flips appears to be a common and fundamental problem in smoothing, and one that we must deal with. The algorithm described in this paper handles this problem by applying a global optimization algorithm to the smoothed function, and following the trajectories of several of the best local minimizers.

In the rest of this paper we will describe in Section 2 the nature of the problem and the structure of the potential energy function. In Section 3 we will describe our efficient approach for smoothing the energy function. How the smoothed function is used as part of a global optimization strategy is explained in Section 4. Section 5 will present some computational results of this approach for two different proteins.
2. Protein Structure and Potential Energy. A naturally occurring protein is a bonded chain of different amino acids. All amino acids (except proline) have the same underlying structure. A central carbon atom ($C_\alpha$), to which are attached: a hydrogen atom ($H$), an amino group ($NH_2$), a carboxyl group ($COOH$) and a residue $R$. Residues are what distinguishes one amino acid from another. Figure 1 is the primary structure of a protein, the repeating chain $-NC_\alpha C'$ is known as the backbone. Overall there are twenty possible different residues, thus we have that many amino acids.

\[
\begin{array}{ccccccc}
H & H & O & H & H & O \\
\mid & \mid & \mid & \mid & \mid & \mid \\
H- & \cdots- & N- & C_\alpha - & C'- & N- & \cdots- & C'- & \cdots- & OH \\
\mid & \mid & \mid & \mid & \mid & \mid & \mid & \mid & \mid & \mid \\
R_i & & R_{i+1} & & & & & & & \\
\end{array}
\]

Fig. 1. The primary structure of a protein. The peptide bond links two amino acids.

2.1. Protein Geometry. There are two different but equivalent coordinate systems used to describe the conformation of a protein: internal and the external (or Cartesian) coordinates. The position of any group of atoms in space can be equally specified in either one of these representations. In external coordinates, each atom is represented by its $x, y$ and $z$ coordinates. In the internal coordinates system, which is more closely related to the structure of a protein, we use the bond length, bond angle and dihedral angle notions to specify the coordinates of the atom. The bond length is simply defined as the Euclidean distance between two consecutive atoms and bond angle is the angle between three consecutive atoms, assuming the atoms are located in a single plane. In the sequence of four atoms, the dihedral angle is the angle between the plane defined by the first three atoms in sequence and the last three atoms in the sequence. This is known as the proper dihedral angle. Ramachandran [20] has showed that there isn't a lot of variation in the values of the bond angles and the bond lengths. In fact, for many proteins, we have only three free variables per amino acid, corresponding to the three dihedral angles in an amino acid. Furthermore the peptide bond (Figure 1) is very rigid, and may be kept fixed. This leaves us with just two free variables per amino acid. So for proteins, use of internal coordinates is computationally more efficient than the external coordinate system. It is easier to keep bond lengths and bond angles fixed in internal coordinates than in external coordinates.

This paper considers the performance of new smoothing methods to solve the molecular conformation of two different proteins: an artificial protein known as polyalanine [9] [21], and metenkephalin [17]. Polyalanine
is a molecular structure consisting of $N$ alanine (ALA) amino acids, which amounts to $10N - 8$ atoms. Polyalanine is a good molecular conformation test problem for two reasons. First, the problem is a difficult global optimization problem due to its sheer size, polyalanine 58 consists of 572 atoms. Secondly, some of the larger polyaniline structures are known to possess more than a single optimal structure [13]. The minimum energy state of polyaniline 58 is known both as straight and bent with very close minimum energy values (Figure 2). On the other hand, metenkephalin is a much smaller naturally occurring protein with just 75 atoms. But unlike polyaniline, metenkephalin consists of four different amino acids. The objective function for proteins contains two terms common to many molecules, namely the Lennard-Jones potential and the electrostatic potential energy function. Thus for this reason, the techniques developed in this paper can be extended to cover smoothing in a wide range of chemical problems.

2.2. Modeling the Potential Energy. As with molecular cluster problems, it is believed that in the native state of a protein, the potential energy function of the protein is in its global energy minimum [1]. Therefore, we need a potential energy function to model the energy of a protein. Several potential energy functions have been developed to model proteins. Three of the most widely used are ECEPP [16] [18], AMBER [27] and CHARMM [4].

For this research we used the CHARMM and later AMBER potential energy functions. Due to differences in constants and other fine details, each reports a different energy for the same configuration. The CHARMM potential energy function is given by

$$E_{\text{CHARMM}} = E_b + E_\phi + E_\psi + E_{\text{vdw}} + E_{\text{es}}$$

(2.1)

in which the first four terms are the separable internal coordinate terms,
and the last two are the pairwise nonbonded interaction terms. The formula for each of these terms, as well as the relevant definitions, are given below.

- \( E_b \) is the bond potential, which equals \( \sum k_b (r - r_0)^2 \), where \( k_b \) is a bond force constant, dependent on the type of the atoms involved in the bond. The actual bond length is \( r \), while \( r_0 \) is the equilibrium bond length, i.e., the ideal bond length for the type of atoms involved in the bond.

- \( E_\theta \) is the bond angle potential, which equals \( \sum k_\theta (\theta - \theta_0)^2 \), such that \( k_\theta \) is an angle force constant which depends on the type of the atoms that constitutes the angle. \( \theta \) and \( \theta_0 \) are the actual bond angle and the equilibrium bond angle respectively.

- \( E_\psi \) is the proper dihedral angle potential. It is given by \( \sum [k_\psi - k_\psi \cos(n \psi)] \), where \( k_\psi \) is a dihedral angle force constant, dependent on the type of the atoms that constitutes the proper dihedral angle, \( n \) is a multiplication factor that can have the values 2, 3, 4 or 6, and \( \psi \) is the actual proper dihedral angle.

- \( E_{\psi} \) is the improper dihedral angle potential, which equals \( \sum k_{\psi} (\omega - \omega_0)^2 \), where \( k_{\psi} \) is an improper dihedral angle force constant that depends on the types of the atoms that constitute the improper dihedral angle (i.e., dihedral angle involving 3 backbone atoms and one residue atom). \( \omega \) and \( \omega_0 \) are the actual and the equilibrium improper dihedral angle, respectively.

- \( E_{\text{vdW}} \) is the van der Waals potential, which is a repulsive-attractive force that is very repulsive at very short distances, most attractive at an intermediate distance, and a very weak attractive force at longer distances. We represent these pairwise interactions between atoms \( i \) and \( j \) using the Lennard-Jones potential of the form

\[
e_{ij} \left[ \left( \frac{\sigma_{ij}}{d_{ij}} \right)^{12} - 2 \left( \frac{\sigma_{ij}}{d_{ij}} \right)^6 \right] \quad (2.2)
\]

where \( d_{ij} \) is the Euclidean distance between atoms \( i \) and \( j \), \( \sigma_{ij} = \sigma_i + \sigma_j \) where \( \sigma \) is the van der Waals radius, and \( \epsilon_{ij} = \sqrt{\sigma_i \sigma_j} \) where \( \epsilon_i \) is the potential well depth.

In this formulation, the Lennard-Jones pairwise equilibrium distance (the distance of greatest attraction) is scaled to \( \sigma_{ij} \), and its minimum energy is scaled to \(-\epsilon_{ij}\).

- \( E_{\text{elec}} \) is the (Coulomb-Fekete) electrostatic potential. Two atoms with charges of same sign repel, and attract if the charges have opposite signs. The pairwise electrostatic energy for atoms \( i \) and \( j \) is given by

\[
\frac{q_{ij}}{4\pi\epsilon_0 d_{ij}} \quad (2.3)
\]
where \( \epsilon_0 \) is the vacuum permittivity, and \( q_{ij} = q_i q_j \) where \( q \) is the charge. Usually, \( C \) is used to denote the constant term \( 1/4\pi \epsilon_0 \).

\( E_b \) is summed over all pairs of bonded atoms. \( E_\theta \) is summed over all bond angles. \( E_\phi \) and \( E_\omega \) are summed over all proper dihedral angles, and all improper dihedral angles respectively. Finally, \( E_{\text{nw}} \) and \( E_{\text{ns}} \) are summed over all pairs of nonbonded atoms. The first four terms force the local structures (bond length, bond angle, proper dihedral angle and improper dihedral angle) into their ideal values. The last two terms account for the long range attractive and repulsive interaction forces.

Our main objective is to find the minimum energy structure of a protein using the CHARMM potential energy function. If we define the position of the protein structure by

\[
x = (x_1, x_2, \cdots, x_n)
\]

where \( x_i \) is a three dimensional vector denoting the coordinates of the \( i \)-th atom, then the overall potential energy function is

\[
E(x) = E_b + E_\theta + E_\phi + E_\omega + E_{\text{nw}} + E_{\text{ns}}
\]

\[
= E_\bullet + \sum_{i \neq j} e_{ij} \left[ \left( \frac{\sigma_{ij}}{d_{ij}} \right)^{12} - 2 \left( \frac{\sigma_{ij}}{d_{ij}} \right)^6 \right] + C q_{ij} \frac{q_i q_j}{d_{ij}}
\]

where \( d_{ij} = \|x_i - x_j\|_2 \) is the distance between atoms \( i \) and \( j \). The term \( E_\bullet \) denotes the sum of bonded interactions, \( E_b + E_\theta + E_\phi + E_\omega \).

In equation (2.5) there are two points worth noting. First, the \( \sigma_{ij}, e_{ij} \) and \( q_{ij} \) occur in tuples (Table 2.1 of [3]). These tuples take completely different values depending on whether there is an attractive or a repulsive force between atoms \( i \) and \( j \). Also, different proteins will have different entries. Secondly, the total contribution of \( E_b + E_\theta + E_\phi + E_\omega \) does not exceed 5-6% of the total potential.

In Figure 3 we show the total pairwise interaction \( E_{\text{nw}} + E_{\text{ns}} \) for two different pairs with an attractive electrostatic potential. Each of the two plots corresponds to one \( \sigma_{ij}, e_{ij} \) and \( q_{ij} \) tuple. As we have quite a few \( \sigma_{ij}, e_{ij} \) and \( q_{ij} \) tuples, the pairwise equilibrium distance would vary accordingly.

For the attractive electrostatic, near distance \( d = 0 \), the Lennard-Jones potential and the electrostatic forces are in opposing directions. But as we can see from Figure 3, the Lennard-Jones is the dominant term due to its higher growth rate. For \( d \gg 0 \) the situation is reversed; the electrostatic potential drops more slowly than the Lennard-Jones potential and thus is the dominant force.

The interaction of Lennard-Jones with a repulsive electrostatic is simple. The sum of the two terms is repulsive for all \( r \) so that there is no finite stable distance between two such atoms.
3. A Smoothing Technique and its Behavior. The basic idea of smoothing a function is to modify it by reducing abrupt function value changes and fine grain fluctuations, while retaining the large scale structure of the original function. As a result, nearby minimizers should merge after sufficient smoothing is applied to remove the barriers between them. Therefore, smoothing reduces the total number of minima in the problem.

The Lennard-Jones and electrostatic potential energy functions for a pair of atoms have a pole at distance zero, and thus very large derivative values for distances near zero. The poles and large gradient values create huge barriers that separate similarly structured minimizers in the overall potential energy function (2.5). That is a fundamental reason why this and similar problems that include Lennard-Jones potential and/or electrostatic potential, have so many minima, and are so difficult to solve. A technique that smoothes Lennard-Jones should be able to remove these barriers in some effective way.

Most smoothing techniques generate a family of smoothing functions that is parameterized by one or more smoothing parameters. Such a family can be represented as

$$ \tilde{E}_{s} : \mathcal{D} \rightarrow \mathbb{R}, \quad \exists \mathcal{D} \subset \mathbb{R}^{n} \text{ and } s \in \mathbb{R}^{d} \quad (3.1) $$

where $D$ is some closed region and $d$ is the number of smoothing parameters. By varying the smoothing parameters, one can create a series of functions that gradually smooths the original function. In our case, a family of smooth problems can be constructed

$$ \min_{x \in \mathcal{D}} \tilde{E}_{s}(x) \quad (3.2) $$
where $s$ is some smoothing parameter set and $\tilde{E}_s$ is a smoothed potential function (2.5). The number of minima should be reduced gradually as the objective function becomes smoother.

One general smoothing technique is spatial averaging, which has been widely studied [10] [11] [12] [14] [23] [29]. The fundamental idea of this technique is that the smoothed function value at each point is given by a weighted average of the energy function in a neighborhood of the point using a distribution function centered at this point. The Gaussian distribution function is commonly used to provide the weighting. In this case, the smoothing transformation is

$$\tilde{f}_\gamma,\lambda(x) = \int H(f(\hat{x}), \gamma)e^{-\|x-\hat{x}\|^2/\lambda^2} d\hat{x}$$ (3.3)

where $\lambda$ and $\gamma$ are the smoothing parameters. The parameter $\lambda$ determines the scale of the Gaussian distribution, while the parameter $\gamma$ is used with the function $H$ to transform the original function $f(x)$ into a function with no poles. The transformation $H(f, \gamma)$ is necessary to make the function integrable, and also further damps the function. This transformation takes on different forms, in the case of [14] it consists of approximating $f(x)$ by a sum of Gaussian functions, while in the work of [29] the transformation consists of truncating $f(x)$ to some fixed maximum value.

In this section, we will discuss a new family of smoothing functions. As opposed to spatial averaging techniques, this family of smoothing functions does not involve integration, but makes an explicit algebraic modification to the more wildly varying terms of the (2.5) potential energy function, and removes poles algebraically. The advantage of this algebraic approach compared to spatial averaging is simplicity and low evaluation cost.

3.1. A New Smoothing Scheme. To algebraically smooth the potential energy function we focus on the terms in (2.5) containing poles, and replace the term $\sigma/d$ with an expression that is finite at zero. We also vary the Lennard-Jones exponents. As a result (2.5) is changed to the smoothed function

$$\tilde{E}_{\gamma,P}(x) = E_\bullet + \sum_{i\neq j}^n e_{ij} \left[ \left( \frac{1 + \gamma}{r_{ij}^2 + \gamma} \right)^P - 2 \left( \frac{1 + \gamma}{r_{ij}^2 + \gamma} \right) \right]$$

$$+ C \frac{q_{ij}}{\sigma_{ij}} \frac{1 + \gamma}{r_{ij}^2 + \gamma}$$ (3.4)
smoothed potential is at distance zero. The other smoothing parameter, $P$, is used to widen (stretch) the minimum’s region of attraction. Note that (3.4) reverts to equation (2.5) if we pick $\gamma = 0, P = 6$, in other words, we turn off the smoothing. The smoothing of the Lennard-Jones term is similar to the smoothed Lennard-Jones potential proposed in [7], but the algebraic form is different to make it more compatible with the electrostatic term here. We also tried a variant of (3.4) where the smoothing was applied to the Lennard-Jones term only, and which is discussed in the Computational Results section.

Examples of (3.4) with $\gamma = 0.1$ are shown in Figure 4. The curves for $P = 3, 4, 5$ and 6 show smoother behavior than the unsmoothed function. For $P = 2$, however, there is a difficulty. The pairwise smoothed potential curve for smoothing $\gamma = 0.1, P = 2$ has a local minimum at $d = 0$; thus the pairwise local minimum of this term is for two atoms to coincide. This difficulty is due to oversmoothing of the Lennard-Jones function, and we can avoid such cases by insisting that pairwise smoothed curve must have a minimum for $d > 0$, which can be assured by having a maximum at $d = 0$.

We develop conditions on the smoothing parameters that must be satisfied in order to have a maximum at zero. Consider two atoms that are distance $d$ apart. Let

$$y = \frac{r^2 + \gamma}{1 + \gamma},$$

(3.5)

where $r = d/\sigma$. Then we can re-write a single pairwise interaction of the smooth potential in terms of $y$

$$\tilde{E}(r) = e \left[ -y^P - 2/y^{P/2} \right] + \frac{C}{\sigma} \sqrt{\frac{r}{y}}$$

(3.6)

It is easily shown that at $d = 0$, $\partial \tilde{E}/\partial r = 2\sigma/(1 + \gamma)\partial \tilde{E}/\partial y|_{r=0} = 0$. 

Fig. 4. Pairwise smoothed potential energy curves between two atoms of opposite charges on different setting of $P$ at $\gamma = 0.1$. Two different vertical scales are shown.
Hence we always have a critical point at zero. The second derivative at zero is

$$\frac{\partial^2 \hat{E}}{\partial r^2} \bigg|_{r=0} = \frac{2}{1 + \gamma} \left[ \frac{\partial \hat{E}}{\partial y} + \frac{2r^2}{1 + \gamma} \frac{\partial^2 \hat{E}}{\partial y^2} \right] \bigg|_{r=0}$$

(3.7)

$$= \frac{2}{1 + \gamma} \frac{\partial \hat{E}}{\partial y} \bigg|_{r=0}$$

(3.8)

$$= \frac{2}{1 + \gamma} \left[ \frac{eP}{y_0^{P/2+1}}(1 - y_0^{-P/2}) - \frac{1}{2}C\frac{q}{\sigma}y_0^{-3/2} \right]$$

(3.9)

$$= -2 \frac{1}{1 + \gamma} \left[ eP\beta y_0^{-P-1} + \frac{1}{2}C\frac{q}{\sigma}y_0^{-3/2} \right]$$

(3.10)

where $y_0 = y \big|_{r=0} = (1 + \gamma)^{-1}$ and $\beta = -y_0^{P/2}(1 - y_0^{-P/2}) = 1 - (1 + \gamma)^{P/2}$.

The possibility of a local minimum at zero is ruled out if this quantity is negative, or equivalently if

$$\frac{1}{\gamma} > \left[ \frac{-Cq}{2\sigma eP\beta} \right]^{1/(P-0.5)} - 1.$$  

(3.11)

Since we are considering smoothing parameter values in the range $0 \leq \gamma < 1$ only, then $\beta = 1 - (1 + \gamma)^{P/2} \geq 1 - 2^{-P/2}$, and (3.11) holds if

$$\frac{1}{\gamma} > \left( \frac{1}{2(1 - 2^{-P/2})P} \right)^{1/(P-0.5)} \times \left( \frac{-Cq}{\sigma e} \right)^{1/(P-0.5)} - 1$$

(3.12)

We want (3.12) to hold for $q$ and $\sigma$ corresponding to all atom pairs $(i, j)$ for the protein under consideration. This is the case if (3.12) is satisfied for the pair $(i, j)$ for which $q_{ij}/(\sigma_{ij}e_{ij})$ is largest, since $q < 0$. The values corresponding to the largest ratio, which is 366.703 for polyalanine 58, are: $C = 332.167$ (a constant), $q = -0.1375$, $\sigma = 1.248$ and $e = 0.0998$ (Table 2.1 of [3]). Substituting these values into (3.12) we get

$$\frac{1}{\gamma} > \left( \frac{183.3515}{(1 - 2^{-P/2})P} \right)^{1/(P-0.5)} - 1.$$  

(3.13)

The bound (3.13) is specific to polyalanine 58 and may vary for other proteins. It is worth noting that we only consider attractive electrostatic potentials in deriving the condition (3.12). This should be clear, since when the electrostatic force is repulsive the smoothed Lennard-Jones potential cannot allow a minimum at zero. In fact $\max\{-Cq/\sigma e\}$ provides a guarantee that this is the lowest interception point (with the potential energy axis) we can have, and any other pair of attractive atoms will have
an interception point that is greater or equal to this. Equation (3.13) is a sufficient condition for a maximum at distance zero.

Figure 5 plots the maximum $\gamma$ for a given $P$ based on (3.13). On the other hand, in (3.12) if we take $\min\{-Cq/\sigma e\}$ then we have a necessary condition for any pair to have a maximum at zero distance.

![Graph](image)

**Fig. 5.** Maximum $\gamma$ for a given $P$ based on equation (3.13) for pairwise local optimization to converge. $\gamma_A$ and $\gamma_B$ are the maximum value of $\gamma$ for sufficient and necessary conditions, respectively.

For example, in Figure 4 it is clear that $\gamma = 0.1, P = 2$ is not a good choice (since it has a minimum point at zero). In that case, according to (3.13) we must have $\gamma < 0.032$ in order to change it to a maximum at distance zero.

For problems with many atoms this type of smoothing also has the effect of simplifying the objective function and reducing the number of local minimizers. This is demonstrated in [24] and [7], and in Section 5. The smoothing is also simple and inexpensive to perform. However, the problem is still not easy, because the global minimum of the smoothed function may not correspond to a local minimizer of the original objective function that is far from global. Indeed, as the experiments in [24] demonstrate, varying the smoothing parameters can cause the order of function values of local minimizers to change extensively. There is no reason to expect this effect to be different for our smoothing technique than for others. Therefore because of these order flips, we still require an effective global optimization algorithm to use with the smoothed energy function.

4. Using Smoothing in a Global Optimization Algorithm. In this section, we discuss a new global optimization algorithm that incorporates the smoothing function described in Section 3. It is based on the global optimization approach proposed in [8], which has been successfully tested on other molecular conformation problems, including water clusters [5] and proteins [7]. The algorithm described here is a modification of that

The essential idea of our approach is to use the global optimization algorithm of [8] on a smoothed version of the potential energy function to find several local minimizers, and then find related minimizers of the original function by local optimization. Although it is possible to work with several different levels of smoothing, we have found it effective to work with simply a single smoothed function, and the original function. The global optimization algorithm in [8] consists of two phases. Phase I (sample generation phase) uses random sampling and local minimization to build an initial conformation of the protein. In phase II we improve upon the initial conformation. This phase is where most of the computational effort lies. All the local minimizations were performed using the BFGS method in the UNCMIN package [22].

As mentioned earlier, we have two different schemes to represent a protein, the internal and the Cartesian coordinate systems. In this algorithm, we use internal coordinates which simplifies the task of fixing the values that are natural to the proteins, such as bond lengths and bond angles. Also, we reduce the dimensionality of the problem by about \(1/15\) (in case of polyalanine 58) of the original problem*, without limiting the folded states a protein can attain.

The first phase of the algorithm starts by generating several initial sample conformations, using a smoothed potential energy function with a reasonable value of the smoothing parameters. The values used here were determined by trial and error, but we have had some success in applying the same values to different proteins. The conformation of the protein is built up one dihedral angle at a time. Each dihedral angle is sampled a number of times, and for every sample the corresponding atoms are added to the polymer and the partial (smoothed) energy is evaluated. The dihedral angle resulting in the best partial energy is chosen. Sampling is then continued with the next dihedral angle. From these sample points, start points for local minimizations are selected and a local minimization is performed from each selected sample point using the smoothed energy. In selecting the start points, we pick the \(\ell\) sample points that are lowest in potential energy. All the smooth minimizers generated by phase I undergo another local minimization, this time to remove the smoothness (desmoothing). The \(m (~ 10)\) lowest unsmooth local minimizers found are passed on to phase II.

---

*This is achieved by fixing all internal parameters (the bond lengths, bond angles, improper dihedral angles and the third proper dihedral angle in each amino acid), leaving as variables the two proper dihedral angles per amino acid that give the backbone of the protein its shape.
Algorithm 4.1 - Framework of the Large-Scale Global Optimization Algorithm for Protein Conformation

1. **Phase I (Sample Generation)**
   
   (a) **Protein sample point buildup:** Build up \( k \) sample configurations from one end of the protein to the other by sequentially generating each dihedral angle in the protein. Randomly sample the current dihedral angle a fixed number of times and select the dihedral angle that gives the lowest energy function value (in smooth domain) for the partial protein generated so far.
   
   (b) **Start point selection:** Select \( \ell < k \) sample points from step 1a to be start points for local minimizations.
   
   (c) **Full-Dimensional local minimizations (smooth):** Perform a local minimization from each start point selected in step 1b.

2. **Desmooth Minimizers:** Do a full-dimensional local minimization for each generated smooth minimizer in step 1c. Collect \( m \) (usually 10) of the best of these non-smooth minimizers to initialize a list \( \mathcal{L} \) for improvement in Phase II.

3. **Phase II (Improvement of Local Minimizers):** For some number of iterations, do:

   (a) **Select a minimizer:** From list \( \mathcal{L} \), select one conformation, and a small subset of \( \sim 5 \) dihedral angles from that local minimizer to be optimized.

   (b) **Global optimization (smooth) on a small subset of variables:** Apply a fairly exhaustive small-scale global optimization algorithm to the smoothed energy of the selected configuration using the selected small subset of the dihedral angles as variables, and keeping the remaining angles temporarily fixed.

   (c) **Full-Dimensional local minimizations (smooth):** Apply a local minimization procedure, with all dihedral angles as variables, to the lowest \( \tilde{m} \) (\( \sim 25 \)) configurations that resulted from the global optimization of the step 3b.

   (d) **Merge the new local minimizers:** Merge the new lowest configurations into the existing list of local minimizers \( \mathcal{L} \).

   (e) **Retire the selected minimizer:** Remove the selected conformation (the one we picked up at step 3a) from \( \mathcal{L} \).

4. **Desmooth Minimizers:** Do a full-dimensional local minimization of the unsmeothed energy for each generated smooth minimizer in step 3d.
In the second phase, we use the \( m \) initial conformations (out of phase I) to initialize a working list, and try to improve them. Note, in this phase as in phase I, all function evaluations and local minimizations including the full-dimensional are done in the smooth domain. A heuristic (described shortly) is used to select a configuration for improvement from the list. Next, we select a number of dihedral angles. The selection of these dihedral angles is based on another heuristic. Then, rather than just sampling on the selected dihedral angles as in the sample point improvement procedure in phase I, a complete global optimization algorithm is applied to find the best new positions for the selected dihedral angles within the selected configuration, with the remainder of the configuration temporarily fixed. The global optimization method used is the stochastic method of [6] [25], which is a very effective global optimization method for problems with small numbers of variables. When doing the global optimization on the selected dihedral angles, the rest of the dihedrals are kept fixed. So generally, these best new positions found for the selected dihedral angles by the small global optimization algorithm lead to configurations in the basin of attraction of new local minimizers for the entire problem. Therefore, a full-dimensional local minimization algorithm is then applied to “polish” the \( m (\sim 25) \) best of these new configurations. The reason for choosing the \( m \) best rather than just the best is because we have found that sometimes the best polished solution does not come from the best unpolished solution. These new full-dimensional local minimizers are then merged into the working list and the entire process is repeated a fixed number of times. In our experience, this phase is able to identify significantly improved local minimizers and leads to the success of the method. Finally, all the smooth minimizers generated by this phase are passed on for a full-dimensional local minimization in the non-smooth domain.

The heuristic used to determine which configuration to select at each iteration of the second phase is the following. We consider an initial configuration and any configurations generated from it to be related, such that the latter is a “descendent” of the former. For some fixed number of iterations, the work in this phase is balanced over each of the \( m \) sets of configurations consisting of an initial minimizers and all of that minimizer’s descendants. In this “balancing phase”, the same number of minimizers is selected from each set, and within a set we select the minimizer with lowest potential energy that has not been selected before. The remaining iterations of the local minimizer improvement phase constitute the ”non-balancing phase” in which we select the best (lowest potential energy) configuration that has not been selected before, regardless of where it is descended from. We have found that the combination of the breadth of search of the configuration space that the balancing phase provides with the depth of search that the non-balancing phase allows is useful to the success of our method.

In our experiments we tested several different criteria for selecting the dihedral angles to be varied in the small-scale global optimizations of Phase
II. These criteria are described in [26]. The method described below was proven to be the most successful for our target, and is the one we ended up using in all of the runs. The main effect of varying a given dihedral angle is on the interaction energies between atoms to the right and to the left of the given angle. Therefore, in this heuristic we compute for each backbone dihedral the total left-right interaction energy and normalize this energy value by the product of the number of atoms to the left times the number to the right. Some specified number (generally five) of dihedral angles with the highest normalized interaction energies are then selected.

The complete framework for the global optimization algorithm for protein conformation is outlined in Algorithm 4.1. For further information on the algorithm and details on the criteria to select dihedral angles, see [26].

5. Computational Results. In this section we report on some of the experiments we conducted as a first step in assessing the effectiveness of the new smoothing algorithm presented in this paper. These experiments give a preliminary indication of the effectiveness of integrating smoothing techniques within a powerful global optimization algorithm. They also provide some insight on the choice of the smoothing parameters $\gamma$ and $P$. We will study the effect of smoothing in phase I and in phase II of the algorithm. The smoothing has been tested on two different proteins: polyalanine and metenkephalin. We report the effect of smoothing in terms of individual phases for polyalanine. Later we repeat the same for metenkephalin.

5.1. Smoothing polyalanine in phase I. Polyalanine-$N$ is a molecular structure that consists of $N - 2$ alanine amino acids, which translates into $10N - 8$ atoms. The phase I algorithm was applied to polyalanine of sizes 3, 5, 10, 20, 30, 40 and 58 amino acids. We will discuss the smoothing in terms of polyalanine of sizes larger than 20 amino acids. This is because our global algorithm without smoothing [26] was able to find the optimal configuration for polyalanine of sizes up to 20 in phase I (albeit not so easily) without employing any of the smoothing techniques.

Our procedure was to generate 1200 sample points and then save the 60 configurations with best smoothed energy. Since the ordering of smooth energy values differs from that of unsmooth energy values, we do a full scale local minimization on the unsmoothed function with these smoothed minimizers as starting points. We then pick the best $m$ configurations based on the unsmoothed values (usually 10) as input to phase II algorithm.

The results for phase I are in Table 1. For each of the tabulated $\gamma, P$ combinations, the results are based on five runs (300 smooth minimizations total). The $(\gamma, P)$ combinations shown were chosen based on prior experience with smoothing Lennard-Jones clusters [7]. In the last column, we consider two unsmooth minimizers to be the same if the difference is less than $10^{-5}$. This last column indicates that number of minimizers found, given 300 starting points, tends to decrease as we increase smoothing (although not monotonically in $\gamma$. This fact would seem to indicate that
smoothing does indeed tend to decrease the number of local minimizers, as was clearly shown for Lennard-Jones clusters in [7]. It is noteworthy that for these values of $N$, phase I without smoothing is unable to find the best known minimizer, while it is found several times using smoothing. For polyalanine size 58, the best minimizer found, with unsmooth value of $-1559.494$ kcal/mole is the best known minimizer with a straight $\alpha$-helix configuration (Figure 2), but, phase I smoothing failed to discover the lowest energy state of polyalanine 58, the bent structure. Interestingly, in one case a smoothing scheme with a pole at distance zero ($\gamma = 0$), but with $P < 6$ fared better than not smoothing.

We also tried a variation of the smoothing scheme (3.4) where we
smoothed the Lennard-Jones term but kept the electrostatic term in the
unsMOOTHed form (i.e. \(C_{ij}/d_{ij}\)). A drawback of this smoothing scheme is
that attractive pairs have a negative pole at zero due to the electrostatic
term. This can yield local minimizers where two atoms coincide. In phase
I, this smoothing scheme found the best straight helix more often than the
one just presented, and it had an even more pronounced tendency to reduce
the number the number of distinct local minimizers found. However, this
scheme did poorly in phase II. See [3] for further details.

5.2. Smoothing polyalanine in phase II. This phase was only
applied to polyalanine 58, since phase I was successful in finding the low
energy states of polyalanines up to 40 and one of the low energy states
of polyalanine 58 (the straight \(\alpha\)-helix configuration). So our goal in this
phase is to find the other low energy state of polyalanine 58 (the bent
\(\alpha\)-helix configuration, Figure 2).

All runs in phase II had the same input. For input to this phase, we
picked the best 10 configurations (in terms of unsmoothed energies) from
the phase I run with \(\hat{E}_{<0.25,5>}\). This set included the -1539.494 (best straight
\(\alpha\)-helix) configuration. The phase II parameters were: 4 balancing and 20

<table>
<thead>
<tr>
<th>(P / \gamma )</th>
<th>best found minimizer [unsmooth]</th>
</tr>
</thead>
<tbody>
<tr>
<td>run 1</td>
<td>run 2</td>
</tr>
</tbody>
</table>

Summary of results of phase II on polyalanine 58. The same starting configuration
was used for all runs. Runs shown are arranged on best, second best, . . . . The \(\bigcirc \) means
we could not improve over the minimum (best) input potential energy. Numbers inside
the parenthesis indicate multiple occurrence of that particular minimizer. Values marked
'**' means these are the best known global minimizer.

<table>
<thead>
<tr>
<th>(P / \gamma )</th>
<th>best found minimizer [unsmooth]</th>
</tr>
</thead>
<tbody>
<tr>
<td>run 1</td>
<td>run 2</td>
</tr>
</tbody>
</table>

Table 2
non-balancing iterations. Each phase II small-scale minimization generated
50 configurations, from which we selected the lowest 25 configurations for
full-dimensional local minimization. We used the smoothing parameter
values that seemed effective in Phase I, as well as some neighboring values.

Table 2 tabulates the results of phase II on polyalanine 58. To get
a better overall picture, we did five runs for each smoothing parameter
combination. These are labeled run 1, run 2, . . . in Table 2 and ordered
with best performance first. For polyalanine 58, the global minimizer is
$-1567.240$. There are two points worth noting. First, for most of the
smoothing parameters, we were able to reach the global at least 20% of the
time (a single run out of five). In some cases we have even a better chance,
for example $\gamma = 0.10, P = 4.75$ (a success ratio of 80%). In some cases we
have multiple occurrences of the global minimizer in a single run, e.g. runs
1-3 in $\gamma = 0.10, P = 4.5$. Second, if we oversmooth then the results are
inferior to no smoothing, e.g. $\gamma = 0.15, P = 4.5$.

To further understand smoothing, we did extensive testing using the
single smoothing parameter choice, $\gamma = 0.15, P = 5$. Again, all runs
had the same input configurations to start with and the same limit on
balancing/non-balancing iterations. In 30 runs out of a total of 38 runs
we saw improvement in the minimizer value at end of phase II. In fact 23
runs out of these 30 runs were successful in finding the global minimizer.
We kept track of the iteration number when a specific minimizer is first
found. In one instance, the $-1567.240$ was found after only 10 iterations
(4 balancing and 6 non-balancing iterations), on average it was generated
on the 15-16th iteration.

It is interesting to trace the chains of minimizers produced by the
algorithm. Since each minimizer produced by phase II is a ‘child’ of some
minimizer found earlier, the local minimizers form a set of trees rooted at
the minimizers input to phase II. Although such a tree is quite large, we
can display the most important part of it by showing only those minimizers
which generated good minimizers. Recall that at step 3c of Algorithm
4.1), we perform 25 full dimensional minimizations, and thus find up to
25 different minimizers, which can be considered children of the minimizer
used to begin that step. A new minimizer is added to the list if it is among
the top 200 found so far. We show in Figure 6 a trace of a single starting
conformation $-1556.95$ (a straight $\alpha$-helix). It shows those minimizers in
the final list which produced children good enough to be added to the list.
The full tree has 178 nodes (minimizers) and is six levels deep. We also
show, in figure 7 a trace from a run that resulted in finding the global
energy minimum configuration three times, interestingly from two different
parents and from different smoothed minimizers. These trees show the
non-monotone nature of the algorithm, and the occurrence of order flips in
smoothing.

As an estimate for the cost of running the algorithm, we report the
number of function evaluations of the parallel version of Algorithm 4.1
**Fig. 6.** A trimmed tree of the full trace of one of the minimizers throughout the life of phase II ($\gamma = 0.15, P = 5$) which yielded the global smoothing minimizers on path to the global, $-1567.240$, are boldfaced. The full tree has 178 nodes (minimizers) and is 6 levels deep. To reduce page cluttering, we only show the non-leaf nodes in the full tree. The value at top is the value of smooth minimizer, the number beneath (italicized) is that after desmoothing the minimizer. The labels at left and right are the ranking of the smooth minimizer within the set of minimizers produced by the small-dimensional optimization before and after the polishing (step 3c in Algorithm 4.1), respectively.

**Fig. 7.** Another trimmed tree of phase II ($\gamma = 0.16, P = 5$). This run found the global thrice (minimizers on the path to the global are boldfaced). The full tree has 149 nodes and is 7 levels deep.

for polyalanine 58. The parallel version of phase I is slightly different than its sequential counterpart, and so the results will vary. The same is true in phase II which is implemented as a three layered structure as opposed to two in phase I. For further details, see [3] and [26]. For phase I, the average number of function evaluations is 2349 per sample. In phase II each iteration requires 21617 function evaluations in addition to the gradient evaluations. In terms of time, phase II takes about 3–4 times as
long as phase I to finish\footnote{Based on running the algorithm on six DEC Alphas using PVM.}. The final desmoothing phase required about 492 function evaluations per smoothed minimizer.

5.3. **Smoothing metenkephalin in phase I.** We also tried Algorithm 4.1 on metenkephalin, a small protein with 75 atoms and five amino acids (Figure 8). Though it is a smaller protein than polyalanine, it is heterogeneous (different amino-acids), so it is a good test for our smoothing technique.

![The tertiary structure of metenkephalin. It has a total of 75 atoms.](image)

The input parameters to smoothing were as follow: generate 600 sample points, then output the best 80 smoothed configurations. At the end of this phase, we desmooth the smoothed minimizers by doing full scale local minimization. Afterward, we pick few of the best (usually 10), unsmoothed minimizers. These will be used as input to the phase II algorithm.

The results for phase I are shown in Table 3. The results for each $\gamma, P$ combination are based on five runs, that is 400 smoothed minimizers. Unlike phase I in polyalanine (Table 1), which was successful in finding the global minimizer for polyalanines up to 40, here it never did find the global minimizer. Considering the fact that metenkephalin has fewer atoms than polyalanine 10, this confirms that this protein’s heterogeneity of amino-acids leads to a more challenging problem. From Table 3 we observe two different patterns. In most of the instances, smoothing was more likely to find a better best minimum than no smoothing. This is clear from the second column in the table. The other pattern is that with smoothing we are more likely to generate bad minima. This is evident from columns 3 and 4 in the Table 3. In other words, with smoothing we generated a wider range of minima. The best ones are better than those generated without smoothing, and at the same time the worst ones are worse than those generated with no smoothing.
<table>
<thead>
<tr>
<th>smooth params</th>
<th>$P$</th>
<th>$\gamma$</th>
<th>overall best min</th>
<th>avg of best minima</th>
<th>worst minimum</th>
<th>average minimum</th>
</tr>
</thead>
<tbody>
<tr>
<td>no smoothing</td>
<td></td>
<td></td>
<td>-13.05571</td>
<td>-10.67496</td>
<td>14.17403</td>
<td>0.96144</td>
</tr>
<tr>
<td>5.50</td>
<td>0.00</td>
<td></td>
<td>-11.59565</td>
<td>-10.49555</td>
<td>21.73974</td>
<td>1.95112</td>
</tr>
<tr>
<td></td>
<td>0.10</td>
<td></td>
<td>-16.90684</td>
<td>-13.22026</td>
<td>52.96502</td>
<td>3.14996</td>
</tr>
<tr>
<td>5.25</td>
<td>0.00</td>
<td></td>
<td>-15.82020</td>
<td>-11.76161</td>
<td>23.03342</td>
<td>2.62739</td>
</tr>
<tr>
<td></td>
<td>0.05</td>
<td></td>
<td>-16.90684</td>
<td>-13.60065</td>
<td>49.39080</td>
<td>3.94882</td>
</tr>
<tr>
<td></td>
<td>0.10</td>
<td></td>
<td>-13.92418</td>
<td>-12.51320</td>
<td>47.20828</td>
<td>4.74853</td>
</tr>
<tr>
<td>5.00</td>
<td>0.00</td>
<td></td>
<td>-11.09422</td>
<td>-8.65001</td>
<td>36.35550</td>
<td>4.14277</td>
</tr>
<tr>
<td></td>
<td>0.05</td>
<td></td>
<td>-11.02481</td>
<td>-8.79110</td>
<td>36.98082</td>
<td>4.72494</td>
</tr>
<tr>
<td></td>
<td>0.10</td>
<td></td>
<td>-17.44337</td>
<td>-14.33585</td>
<td>58.78065</td>
<td>4.72065</td>
</tr>
<tr>
<td></td>
<td>0.15</td>
<td></td>
<td>-17.09287</td>
<td>-13.42397</td>
<td>40.46203</td>
<td>4.97514</td>
</tr>
<tr>
<td></td>
<td>0.20</td>
<td></td>
<td>-11.05957</td>
<td>-9.18396</td>
<td>48.71783</td>
<td>5.67317</td>
</tr>
<tr>
<td></td>
<td>0.25</td>
<td></td>
<td>-13.15117</td>
<td>-11.82889</td>
<td>62.58349</td>
<td>6.25258</td>
</tr>
<tr>
<td>4.75</td>
<td>0.00</td>
<td></td>
<td>-15.80765</td>
<td>-10.99920</td>
<td>37.74938</td>
<td>4.09213</td>
</tr>
<tr>
<td></td>
<td>0.05</td>
<td></td>
<td>-12.87537</td>
<td>-10.99422</td>
<td>46.34991</td>
<td>4.58853</td>
</tr>
<tr>
<td></td>
<td>0.10</td>
<td></td>
<td>-14.93210</td>
<td>-11.61584</td>
<td>44.97417</td>
<td>5.10596</td>
</tr>
<tr>
<td></td>
<td>0.15</td>
<td></td>
<td>-15.66696</td>
<td>-12.95194</td>
<td>48.13361</td>
<td>6.67790</td>
</tr>
<tr>
<td></td>
<td>0.20</td>
<td></td>
<td>-15.38735</td>
<td>-11.24722</td>
<td>50.50901</td>
<td>6.64184</td>
</tr>
<tr>
<td></td>
<td>0.25</td>
<td></td>
<td>-12.27669</td>
<td>-9.59700</td>
<td>44.89006</td>
<td>6.78987</td>
</tr>
<tr>
<td>4.50</td>
<td>0.00</td>
<td></td>
<td>-13.97437</td>
<td>-11.60363</td>
<td>47.81042</td>
<td>4.64018</td>
</tr>
<tr>
<td></td>
<td>0.05</td>
<td></td>
<td>-14.89564</td>
<td>-9.42210</td>
<td>38.91343</td>
<td>4.50926</td>
</tr>
<tr>
<td></td>
<td>0.10</td>
<td></td>
<td>-17.09297</td>
<td>-13.10101</td>
<td>48.14509</td>
<td>5.38676</td>
</tr>
<tr>
<td></td>
<td>0.15</td>
<td></td>
<td>-15.80756</td>
<td>-12.05443</td>
<td>51.10003</td>
<td>6.44918</td>
</tr>
<tr>
<td></td>
<td>0.20</td>
<td></td>
<td>-13.22904</td>
<td>-11.18331</td>
<td>52.99727</td>
<td>7.58794</td>
</tr>
<tr>
<td></td>
<td>0.25</td>
<td></td>
<td>-16.90807</td>
<td>-11.17747</td>
<td>66.32735</td>
<td>8.58450</td>
</tr>
</tbody>
</table>

Table 3

Summary of results of phase I on metenkephalin. The results are based on five runs. The column overall best min corresponds to the overall best minimum among the five runs, while average of best minima is the average of the best in each of the five runs.

5.4. Smoothing metenkephalin in phase II. To be consistent throughout, every run in phase II used the same initial set of minimizers. The input for this phase was 10 unsmoothed minimizers from phase I with smoothing parameters $\gamma = 0.1, P = 5$. The best input configuration had energy $-17.09286$. The run time parameters were: 4 balancing and 20 non-balancing iterations. This is the same as we used for phase II in polyalanine. Phase II kept a sorted list of the best 250 smoothed minimizers. When a good minimizer is found, it is added to the list, and a minimizer from the end of the list is discarded.

The results for phase II are tabulated in Table 4. We did six runs
for each smooth parameter combination, with the best run labeled run 1, etc. Since even the algorithm without smoothing was twice successful in reaching the global minimum (−18.8475), we include more information to judge between different smoothings. Besides listing the best unsmoothed minimizer in each run, we list the iteration in which it was first generated, and the number of times it was generated throughout phase II. These two values are enclosed in brackets in front of the best unsmoothed minimizer (Table 4).

Overall it is clear that most of the smoothing runs did a better job than the no smoothing run. For example: all six runs of $\gamma = 0.04, P = 4.5$ were successful in reaching the global more than once, but these runs usually found the global minimizer later than the no-smoothing run (when the no-smoothing run actually found it). However, there is clearly significant random variation. Some smoothing runs were inferior to the no smoothing runs, e.g $\gamma = 0.10, P = 5.5$ and $\gamma = 0.05, P = 5$, where none of the runs reached the global. However, using the slightly different values, $\gamma = 0.04, P = 5$, resulted in reaching the global in five out of six runs.

6. Conclusions and Future Research. The protein conformation problem belongs to the class of NP-hard problems. By taking advantage of the internal structure of the protein, the algorithm of [8] performs well. However, this algorithm, without smoothing, failed to find the minimum energy state of polyalanine larger than 20. In the first part of the paper we introduced a new family of smoothing functions to use in conjunction with this algorithm. In our experiments, this algorithm was able to find the apparent low energy state(s) of all sizes of polyalanines we tried. For polyalanine 58 we found two different low energy states, a straight and a bent $\alpha$-helical structure. The algorithm also performed well on a smaller, but more complex protein, metenkephalin.

There are some points worth future consideration. The algorithm we present uses smoothing to simplify the energy function, but minimizes the harmful effect of order flips by using more computational effort to track more local minimizers. An important issue is how this extra effort will scale when we tackle larger proteins. It is also desirable to achieve a better understanding of the mathematical behavior of the smoothing function as smoothing increases, especially in comparison with spatial averaging techniques. The smoothing parameter values that worked best were somewhat different for the two proteins we studied; how to estimate the best values in advance is another important topic for future work.

REFERENCES

New Smoothing Techniques for Global Optimization in Solving for Protein ...


<table>
<thead>
<tr>
<th>Formula</th>
<th>Value</th>
<th>Value</th>
<th>Value</th>
<th>Value</th>
<th>Value</th>
<th>Value</th>
<th>Value</th>
<th>Value</th>
<th>Value</th>
<th>Value</th>
<th>Value</th>
<th>Value</th>
<th>Value</th>
<th>Value</th>
<th>Value</th>
<th>Value</th>
<th>Value</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>(1/6) 1908</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
</tr>
<tr>
<td>(1/7) 1908</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
</tr>
<tr>
<td>(1/8) 1908</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
</tr>
</tbody>
</table>

Summary of results of phase II on methionine. The same trend was used for all runs. The "p" refers to the global minimum. Runs are notched by best, second best, and best.