Optimization Methods and Software
Vol. 00, No. 00, Month 200x, 1–12

GLOBAL OPTIMIZATION IN PROTEIN DOCKING USING CLUSTERING, UNDERESTIMATION AND SEMIDEFINITE PROGRAMMING

ROUMMEL F. MARCIA[a,*], JULIE C. MITCHELL[a,†], and STEPHEN J. WRIGHT[b,‡]

[a]Departments of Biochemistry and Mathematics, UW-Madison, Madison, WI 53706
[b]Department of Computer Sciences, UW-Madison, Madison, WI 53706

(Received December 2005)

The underestimation of data points by a convex quadratic function is a useful tool for approximating the location of the global minima of potential energy functions that arise in protein-ligand docking problems. Determining the parameters that define the underestimator can be formulated as a convex quadratically constrained quadratic program and solved efficiently using algorithms for semidefinite programming (SDP). In this paper, we formulate and solve the underestimation problem using SDP and present numerical results for active site prediction in protein docking.

AMS Subject Classification: 65K05, 65K10, 90C22, 90C30, 90C90

Keywords: Protein docking, Global optimization, Convex underestimation, Semidefinite programming

1 Introduction

Protein-ligand docking problems in computational biology can be formulated as global minimization problems in which the docked configuration of the two molecules corresponds to the global minimum of a potential energy function describing the molecular interaction. Typically, the energy landscape is funnel-shaped and highly nonlinear with many local minima, making the docking

* Address: Mitchell Lab, 433 Babcock Drive, Madison, WI 53706, U.S. Tel.: (608) 262-6844. Fax: (608) 262-3453. E-mail: marcia@math.wisc.edu
† Address: 433 Babcock Drive, Madison, WI 53706, U.S. Tel.: (608) 263-6819. Fax: (608) 262-3453. E-mail: mitchell@math.wisc.edu
‡ Address: 120 West Dayton Street, Madison, WI 53706, U.S. Tel.: (608) 262-4838. Fax: (608) 262-9777. E-mail: swright@cs.wisc.edu

problem very difficult to solve. Convex global underestimators (CGU) [1–5] were developed to determine the location of the global minimum of such functions by iteratively underfitting a set of data points over a contracting domain by a sequence of strictly convex quadratic functions. The (unique) minimum of each convex quadratic underestimator is an approximation to the global minimum and is used to define the domain in the next iteration. Underestimation methods have been successfully implemented in protein structure determination [1–4] as well as in protein docking [6–10].

Determining the parameters p that define the underestimator $q(p;x)$ involves solving a nonlinear program that minimizes L^1 distance between the energy function $f(x^{(k)})$ and $q(p;x^{(k)})$, $k=1,2,\ldots,m$, subject to $q(p;x^{(k)}) \leq f(x^{(k)})$ for each k; see (1). The L^1 distance is used primarily because it is robust with respect to outliers [11–13]. The quadratic function $q(p;x)$ is required to underestimate $f(x)$ so that its minimizer is a plausible predictor of the global minimum of $f(x)$. To guarantee that $q(p;x)$ is convex, its Hessian matrix H with respect to x must be positive definite. In the original formulation of CGU [1], H was chosen to be a diagonal matrix with positive diagonal entries, which simplifies the underestimation problem to a linear program. The choice for the Hessian was generalized to dense positive definite matrices in [5], where H was expressed in terms of its Cholesky factorization $H=LL^T$, with L a lower triangular matrix with positive diagonal entries. In this case, the nonlinear program becomes a quadratically constrained quadratic program whose global solution is not necessarily easy to find. A two-phase approach, proposed in [5], finds an underestimator with a diagonally dominant Hessian H in the first phase, using H as the initial guess in the second phase, which solves for the Cholesky factorization LL^T. The globally optimal positive definite Hessian is almost always found by this approach. In this paper, we solve the underestimation problem using a single-phase approach, formulating it as a semidefinite programming problem that can be solved efficiently.

Semidefinite programming (SDP) can be viewed as an extension of linear programming in which the unknowns include symmetric matrices as well as vectors and scalars, and nonnegativity constraints on the variables become positive semidefiniteness requirements on the symmetric matrix variables. More information on SDP can be found in [14] and [15]. It is natural to formulate (1) as an SDP in which the Hessian of the quadratic underestimator q appears as one of the unknowns. Paschalidis et al. [16], concurrently with the submission of this paper, described an underestimation procedure based on an SDP formulation, using a biased sampling procedure to select new starting points at each major iteration. The authors discuss the relationship of their algorithm to earlier versions of CGU [4], and obtain test results on three problems, including those used in Examples 4 and 5 below. However, they fixed the three orientation variables for the ligand at their optimal values, and applied their

procedure only to the three translational variables, and from a nearby starting point. In contrast, our results above did not assume any such knowledge of the global minimizer.

A second innovation presented in this paper is the use of clustering to determine separate regions of space to be searched for the global minimizer. Frequently, an initial scan of the parameter space reveals widely separated regions with low function values, which can be identified by applying clustering procedures to the local minima found during the scan. A process of repeated convex underestimation, random point generation, and local optimization can be applied to each cluster separately, to find the "global" minimizer for each cluster. The best such solution becomes our estimate of the overall global solution. (We note that clustering was not used in [16].)

The paper is organized as follows. We describe the convex quadratic underestimator and the SDP formulation in Section 2. Section 3 gives details on DoME, the algorithm for global minimization of energy functions arising in docking applications, and Section 4 describes computational tests on five problems from the Protein Data Bank [17]. We summarize our conclusions in Section 5.

2 SDP Formulation

Given m data points $(x^{(1)}, f(x^{(1)}))$, $(x^{(2)}, f(x^{(2)})), \cdots, (x^{(m)}, f(x^{(m)}))$, where $x^{(k)} \in \Re^n$, $k = 1, 2, \cdots, m$, and $f\colon \Re^n \to \Re$, we compute a convex quadratic function that underestimates these data points by defining a quadratic function $q(c_0, c, H; x) = c_0 + c^T x + \frac{1}{2} x^T H x$, where $c_0 \in \Re$, $c \in \Re^n$, and $H \in \Re^{n \times n}$, and solving the minimization problem

$$\begin{array}{lll} \underset{c_0,c,H}{\text{minimize}} & \displaystyle\sum_{k=1}^{m} s_k & \\ \text{subject to} & q(c_0, c, H, x^{(k)}) + s_k = f(x^{(k)}), & k = 1, 2, \cdots, m \\ & s_k \geq 0, & k = 1, 2, \cdots, m \\ & H \text{ symmetric positive definite,} & \\ & |c_0|, \|c\|, \text{ and } \|H\| \text{ bounded,} & \end{array} \tag{1}$$

where $s_k \in \Re$ for $k = 1, 2, \cdots, m$. The constraint that H is positive definite ensures that $q(c_0, c, H; x)$ is convex with respect to x. We impose the following explicit bounds on c_0 and c:

$$|c_0| \leq \beta_0, \quad \text{and} \quad \|c\|_\infty \leq \beta_c, \tag{2}$$

and discuss the bounding of $\|H\|$ further below.

We now formulate (1) as a semidefinite program. First, by introducing the slack variables $v_0^+, v_0^- \in \Re$ and $v^+, v^- \in \Re^n$, we can express (2) as

$$\begin{aligned} v_0^+ &= \beta_0 - c_0 \geq 0 \\ v_0^- &= c_0 + \beta_0 \geq 0 \\ v^+ &= \beta_c e - c \geq 0 \\ v^- &= c + \beta_c e \geq 0, \end{aligned}$$

where $e \in \Re^n$ is the vector of ones. Note that

$$c_0 = \frac{1}{2}(v_0^- - v_0^+), \quad \text{and} \quad c = \frac{1}{2}(v^- - v^+).$$

Using these formulae, we can eliminate the parameters c_0 and c, and the constraints (2) can be written as follows:

$$\begin{aligned} v_0^+ + v_0^- &= 2\beta_0 \\ v^+ + v^- &= 2\beta_c e \\ v_0^+,\ v_0^-,\ v^+,\ v^- &\geq 0. \end{aligned}$$

Next, we note that $x^T H x = X \bullet H$, where $X = xx^T$ and $A \bullet B = \sum_{i,j} A_{i,j} B_{i,j}$ is the standard inner product of symmetric matrices. To ensure that q has a (strictly) positive definite Hessian, we replace H by $H + \varepsilon I_n$, where $\varepsilon > 0$ is some parameter that is a lower bound to the smallest eigenvalue of the desired Hessian. The corresponding equality constraint becomes

$$\begin{gathered} c_0 + c^T x^{(k)} + \frac{1}{2}(X^{(k)} \bullet H) + s_k = f(x^{(k)}) - \frac{\varepsilon}{2}\|x^{(k)}\|_2^2, \\ s_k \geq 0. \end{gathered}$$

The size of H can be controlled by adding a regularization term $\lambda \cdot \mathrm{trace}(H) = \lambda \cdot (I_n \bullet H)$ to the objective, where λ is a "tuning" parameter that can be successively increased to force $\|H\|$ to drop below a prescribed bound. In fact, λ can be interpreted as a Lagrange multiplier for a constraint $\mathrm{trace}(H) \leq C$, for some bound C. (In our computational experience, $\|H\|$ does not grow too large even for $\lambda = 0$, but we include it in our formulation for completeness.)

In summary, our SDP formulation of (1) is given by

$$\begin{aligned}
\text{minimize} \quad & \sum_{k=1}^{m} s_k \;+\; \lambda(I_n \bullet H) && \text{(SDP)}\\
\text{subject to} \quad & \frac{1}{2}(v_0^- - v_0^+) + \frac{1}{2}(v^- - v^+)^T x^{(k)} + \frac{1}{2}(X^{(k)} \bullet H) + s_k \\
& \qquad = f(x^{(k)}) - \frac{\varepsilon}{2}\|x^{(k)}\|_2^2 \;\text{ for all } k \\
& v_0^+ + v_0^- = 2\beta_0 \\
& v^+ + v^- = 2\beta_c e \\
& v_0^+,\; v_0^-,\; v^+,\; v^- \geq 0 \\
& s_k \geq 0, \quad k = 1, 2, \cdots, m, \\
& H \succeq 0.
\end{aligned}$$

Although we can apply transformations to the vector and scalar variables to obtain a problem with symmetric matrix variables only, such transformations are computationally inefficient and in any case unnecessary, as current SDP software can handle such variables explicitly.

3 The DoME Algorithm

The Docking Mesh Evaluator (DoME) is software for predicting the active site of proteins upon binding with ions, DNA, ligands, and other macromolecules. The proteins are treated as rigid bodies, with one (receptor) fixed in space while the other (ligand) is allowed to move and rotate, leading to six degrees of freedom: three translational and three rotational. Previously, DoME described the molecular interaction by defining an energy model based only on solvent effects and van der Waals forces [7]. Specifically, the electrostatic interactions in DoME were described by a finite-element solution to the Poisson-Boltzmann equations (PBE) while the dipole moments and steric repulsion were computed using a standard Lennard-Jones 6-12 formula. Because the computed solution to the PBE was piecewise-linear, the potential energy function was not differentiable, and therefore, non-gradient-based optimization had to be used for local minimization.

In the new version of DoME, the potential energy function has been made differentiable by utilizing the analytical solution to the linearized PBE, known as the Yukawa potential. While this model does not address dialectric effects to the same level of detail as the Poisson-Boltzmann model, it nonetheless in-

corporates the Debye-Huckel screening of charges in solvents. Two additional terms, representing hydrogen bonding and desolvation, have also been included in this version of DoME. Solvation plays an important role in protease-inhibitor complexes. The hydrogen bond energy term has a hydrogen-acceptor cutoff distance of 4.5Å and a donor-hydrogen-acceptor (D-H-A) cutoff angle of 90°. It includes a dependence on the D-H-A angle since bond strength favors a linear alignment. The free energy associated with removing solvents at the active site is approximated by computing the change in the solvent accessible surface area [18]. (For further details on each energy term, see [19].)

DoME searches for the global minimum by performing an exhaustive scanning preprocessing phase, followed by a sequence of major iterations, each consisting of scanning and underestimation. In the preprocessing phase, a scan is performed over the six degrees of freedom by holding the receptor in a fixed position and orientation, then placing the ligand at various distances from it (fixing three degrees of freedom) and oriented in various ways (fixing the other three). Typically, energies are evaluated at about 2 million sampling points during this initial phase, and the lowest 900 are used as starting points for local optimizations. The local minima thus found are underfitted with the convex quadratic function constructed in the manner described in Section 2. The global minimizer of the underestimator is used to initialize another local search and thus obtain another local minimizer, which we call x^*_{pred}. On each subsequent major iterations, we construct a new, smaller search domain that encompasses x^*_{pred} along with the lowest k_b local minima obtained from the other local searches. (A typical value for k_b is three.) Random points are generated within this smaller domain, followed by local minimization from each of these points, underestimation, and another local search form the minimizer of the underestimator. The process is repeated until the the predicted global minima is close to the local minima with the lowest known function value and the domain size is sufficiently small. This algorithm is described more fully in [7].

It has been shown that the coupled use of scanning and optimizing is more effective in determining points of low energy values than by scanning or optimization alone [6]. However, we find that the initial exhaustive scan sometimes produces points with low energy values in distinct regions in space. In this work, we enhanced DoME by clustering the low-energy points found during the scan in the preprocessing phase on the basis of their (x, y, z) coordinates. We then form a separate quadratic underestimator for each cluster and apply the subsequent major iterations to each cluster separately. At the end of this process, we obtain an approximate global minimizer for each cluster. For clustering, we used the k-Median algorithm of Bradley et al. [20], which minimizes the total L^1 distance between points in a chosen cluster. This algorithm is robust and is guaranteed to converge to a solution satisfying the mini-

mum principle necessary optimality condition for the problem [21]. No more than four clusters were used, ensuring that each cluster contained sufficiently many points for underestimation. This approach differs from the clustering algorithm for protein complexes called ClusPro [22], which computes the root-mean-square deviation (RMSD) between ligand residues at the interface of each candidate conformations and groups those structures within a (default) cluster radius of 9Å RMSD. The clusters are then ranked based on the number of structures they contain. It is possible that such an approach would yield many clusters (as many as 30) and would not guarantee the number of data points necessary to determine the convex quadratic underestimator, so we elected not to use this approach.

4 Computational Examples in Protein Docking

We applied the enhanced DoME code described in the previous section to five examples from the Protein Data Bank [17]. For each of these examples, the bound configuration is known, allowing us to compare the DoME results with the known global minimizer.

Example 1: *CheY-binding domain of CheA in complex with CheY (1A0O).* The 1A0O complex consists of response regulator of bacterial chemotaxis, CheY, bound to the recognition domain from its cognate histidine kinase, CheA [27]. This specific recognition domain minimizes the cross-talk in signal transmission mediated by Mg^{2+}-dependent phospho-relay reactions between histidine auto-kinases and phosphoaccepting receiver domains.

Example 2: *V-1 Nef protein in complex with wild type Fyn SH3 domain (1AVZ).* The antibody-antigen complex 1AVZ consists of a viral protein, HIV-1 Nef, and the host cell signal transduction protein, Fyn tyrosine kinase SH3 domain [28]. The interaction between these two proteins provides for long term survival of infected T cells and for destruction of non-infected T cells by inducing apoptosis.

Example 3: *Trypsin complex with Bowman-Birk inhibitor (1TAB).* The 1TAB complex consists of the enzyme trypsin and BBI, the Bowman-Birk trypsin-inhibitor, which is a polypeptide chain of 71 amino acids highly cross-linked by seven disulfide bridges [29]. Elevated levels of trypsin have been found in pancreatic tumors, and BBI, commonly found in soybeans, has been shown to suppress this type of tumor in various animals.

Example 4: *Barnase-barstar complex (1BRS)* Barnase is an extracellular ribonuclease found in bacillus amyloliquefaciens. The intracellular polypeptide

Cluster	Iter	$E(x^0_{\min})$	$E(x^f_{\min})$	$\|\cdot\|_2$	$\|\cdot\|_\theta$
1	4	-29.652	-41.779	32.516	1.314
2	3	-58.535	-58.804	0.138	0.022
3	4	-42.855	-46.526	55.774	3.093
4	5	-41.645	-44.416	55.690	2.954
-	4	-58.535	-58.788	0.122	0.025

Table 1: Results for 1A0O.

inhibitor barstar disrupts its potentially lethal functions by sterically blocking its active site with a helix and adjacent loop segment [30].

Example 5: *Trypsin-pancreatic enzyme inhibitor complex (2PTC).* The 2PTC complex involves trypsin interacting with a bovine pancreatic enzyme inhibitor [31].

We solved (SDP) using the MATLAB software package SDPT3 (version 3.0) of Tütüncü et al. [23], which is an interior-point algorithm that uses a predictor-corrector primal-dual path-following method. In our numerical testing, the default HKM [24–26] direction was used. All runs were made on a single 2.20 GHz Pentium 4 processor Linux workstation with 896 MB of RAM from Dell Computers and 2.0 GHz Apple Power Mac G5 Cluster. The computational times for each cluster in each example ranged between 7.5 and 12.5 hours on four processors, with 80% of the time spent on local optimization and 20% on random point generation and function evaluations. Constructing the underestimator at each iteration took no more than two seconds.

Results for the five examples appear in Tables 1–5. We denote the local minima with the lowest energy value in the initial scanning and in the final iteration by $x^0_{\min}$ and $x^f_{\min}$, respectively. For each of the four clusters, we list the number of major DoME iterations (`Iter`), the energy values ($E(x^0_{\min})$ and $E(x^f_{\min})$) in kcal/mol, and the 2-norm $\|\cdot\|_2$ (Å) and angular distances $\|\cdot\|_\theta$ (radians) of the local minima $x^f_{\min}$ to the known docked configuration. As a point of comparison, we list in the last row of each table, indicated by "–" in the Cluster column, the results obtained by scanning and underfitting without clustering the data points in the initial iteration.

Analysis of Results. In the five examples presented, DoME approximated the locations of the global minima by clustering the initial points followed by iterative underestimation. In the 1A0O complex (Example 1), a point near the global minimum was detected in DoME's initial scanning, in cluster 2. Naturally, DoME identified this point correctly as the global minimum, after several iterations. It is interesting to note that DoME also finds the global solutions if clustering is not used. Due to the low energy value of the minimum

Cluster	Iter	$E(x^0_{\min})$	$E(x^f_{\min})$	$\|\cdot\|_2$	$\|\cdot\|_\theta$
1	3	-31.071	-45.214	24.565	2.075
2	4	-42.132	-45.280	24.855	1.836
3	4	-45.515	-53.381	6.353	2.987
4	4	-47.406	-76.947	0.276	0.079
-	5	-47.406	-60.195	5.556	2.893

Table 2: Results for 1AVZ.

Cluster	Iter	$E(x^0_{\min})$	$E(x^f_{\min})$	$\|\cdot\|_2$	$\|\cdot\|_\theta$
1	5	-41.503	-44.450	36.134	2.767
2	3	-50.438	-59.761	0.092	0.081
3	3	-47.359	-47.587	30.139	0.567
4	3	-52.275	-53.503	36.360	1.104
-	5	-52.275	-53.524	36.392	1.108

Table 3: Results for 1TAB.

Cluster	Iter	$E(x^0_{\min})$	$E(x^f_{\min})$	$\|\cdot\|_2$	$\|\cdot\|_\theta$
1	5	-40.996	-47.611	44.068	1.193
2	4	-49.732	-71.465	0.001	0.110
3	6	-52.534	-52.550	43.042	2.666
4	4	-41.650	-42.118	33.400	3.011
-	4	-52.534	-52.553	43.040	2.666

Table 4: Results for 1BRS.

Cluster	Iter	$E(x^0_{\min})$	$E(x^f_{\min})$	$\|\cdot\|_2$	$\|\cdot\|_\theta$
1	4	-41.383	-61.791	0.008	0.133
2	5	-45.805	-53.242	39.491	2.292
3	4	-47.584	-48.696	46.499	2.632
4	4	-49.924	-53.255	30.916	1.243
-	6	-49.924	-55.015	37.828	0.624

Table 5: Results for 2PTC.

($E(x^0_{\min}) = -58.525$ kcal/mol) in relation to the other local minima values found, the shape of the underestimator is such that its minimum nearly coincides with the global minimum. Thus there was a large decrease in the size of the search domain after the initial scanning phase, and subsequent iterations were able to obtain nearby local minima with even lower energy values.

In Example 2 (1AVZ), the global minimum was again found in the initial scanning but was removed from the set of local minimizers before performing the underestimation, to see if subsequent iterations could recover this point. Again, clustering and iterative underestimation was able to locate a very good

approximation to the global minimum, even though the initial point $x^0_{\min}$ in cluster 4 had a much higher energy value than the global minimizer, which lies in this cluster. When clustering was omitted in this example, DoME found a minimizer that that was located relatively near the global minimum (about 5.5 Å away) but was oriented incorrectly. It is the steric repulsion induced by this incorrect orientation that prevented the ligand molecule from achieving the correct translational coordinates for binding.

In Examples 3, 4, and 5 (the 1TAB, 1BRS, and 2PTC complexes), widely separated regions of low energy are identified during the initial scan, and the use of clustering enabled DoME to identify the global minimizer correctly in each case. When clustering was not used, the single quadratic underfitting function had its minimizer between these low-energy regions, in a region that was neither low in energy value nor near any of the minima. In each of these cases, when a local search was performed from the minimizer of the convex underestimator, the best of the four initial local minimizers from each of the clusters was identified as x^*_{pred}. Subsequent DoME iterations in the single-cluster case failed to identify the known global minimizers.

It is interesting to note that our "divide and conquer" approach to optimization always produced a near-native (globally minimizing) structure as the predicted global minimum of one of the initial clusters. In two of five cases, the near-native configuration corresponded was found in the cluster whose local minimizer had the lowest function value after the initial scanning phase. However, in the other three cases (1TAB, 1BRS, 2PTC) the near-native structure was obtained in a different cluster, one for which the local minimizer obtained after the scanning phase had a higher function value. This fact demonstrates the value of the multiple cluster CGU approach, because it allows for several domains to be searched, thereby increasing the chance of producing a near-native solution.

5 Conclusion

We presented a method for computing a convex quadratic function that underestimates a set of points for determining the global minimum of a function. We formulated the problem as a semidefinite program, which generally can be solved efficiently in theory and practice. We applied this approach in the context of protein docking and showed that a combination of clustering and iterative underestimation effectively predicted near-native docking configurations for several test cases. A more comprehensive survey of docking problems will determine whether near-native conformations are always found as cluster global minima.

Acknowledgments

The authors would like to thank Susan Lindsey and Erick Butzlaff for their contributions to DoME. This work was supported by the Department of Energy Grant DE-FG03-01ER25497 and the National Library of Medicine Training Grant 5T15LM007359-03.

References

[1] A. Phillips, J. Rosen, and V. Walke (1996). Molecular structure determination by convex global underestimation of local energy minima. volume 23 of *DIMACS Series in Discrete Mathematics and Theoretical Computer Science*, pages 181–198. Amer. Math. Soc.

[2] K. A. Dill, A. T. Phillips, and J. B. Rosen (1997). CGU: an algorithm for molecular structure prediction. In *Large-scale optimization with applications, Part III (Minneapolis, MN, 1995)*, volume 94 of *IMA Vol. Math. Appl.*, pages 1–21. Springer, New York.

[3] J. B. Rosen, K. A. Dill, and A. T. Phillips (1997). Protein structure and energy landscape dependence on sequence using a continuous energy function. *J. Comp. Biol.*, **4**, 227–239.

[4] A. T. Phillips, J. B. Rosen, K. .A. Dill (2001). Convex global underestimation for molecular structure prediction. In: P. M. Pardalos et al (Ed.) *From Local to Global Optimization* (Dordrecht, Netherlands: Kluwer Academic Publishers), 1–18.

[5] J. B. Rosen and R. F. Marcia (2004). Convex quadratic approximation. *Comput. Optim. Appl.*, **28**(2), 173–184.

[6] J. C. Mitchell, J. B. Rosen, A. T. Phillips, and L. F. Ten Eyck (1999). Coupled optimization in protein docking. In *Proceedings of the Third Annual International Conference on Computational Molecular Biology*, pages 280–284. ACM Press.

[7] R. F. Marcia, J. C. Mitchell, and J. B. Rosen (2005). Iterative convex quadratic approximation for global optimization in protein docking. *Comput. Optim. Appl.*, **32**(3), 285–297.

[8] O. L. Mangasarian, J. B. Rosen, and M. E. Thompson (2005). Global minimization via piecewise-linear underestimation. *J. Global Optim.*, **32**(1), 1–9.

[9] O. L. Mangasarian, J. B. Rosen, and M. E. Thompson (2006). Convex kernel estimation of functions with multiple local minima. *Comput. Optim. Appl.*, **34**(1), 35–45.

[10] O. L. Mangasarian, J. B. Rosen, and M. E. Thompson (2006). Nonconvex piecewise-quadratic underestimation for global minimization. *J. Global Optim.*, **34**(4), 475–488.

[11] I. Barrodale (1968). L_1 approximation and the analysis of data. *Appl. Stat.*, **17**, 51–57.

[12] I. Barrodale and F. D. K. Roberts (1973). An improved algorithm for discrete L_1 linear approximation. *SIAM J. Numer. Anal.*, **10**, 839–848.

[13] J. B. Rosen, H. Park, J. Glick, and L. Zhang (2000). Accurate solution to overdetermined linear equations with errors using L_1 norm minimization. *Comput. Optim. Appl.*, **17**(2-3), 329–341.

[14] Y. Nesterov and A. Nemirovskii (1994). *Interior-point polynomial algorithms in convex programming*, volume 13 of *SIAM Studies in Applied Mathematics*. Society for Industrial and Applied Mathematics (SIAM), Philadelphia, PA.

[15] L. Vandenberghe and S. Boyd (1996). Semidefinite programming. *SIAM Rev.*, **38**(1), 49–95.

[16] I. Ch. Paschalidis, Y. Shen, S. Vajda, P. Vakili (2004). A semi-definite programming-based underestimation method for global optimization in molecular docking. *Proceedings of the 44th IEEE Conference on Decision and Control, and the European Control Conference 2005*, 3675–3680.

[17] F.C. Bernstein, T.F. Koetzle, G.J.B. Williams, E.F. Jr. Meyer, M.D. Brice, J.R. Rogers, O. Kennard, T. Shimanouchi, and M. Tasumi (1977). The Protein Data Bank: a computer-based archival file for macromolecular structures. *J. Mol.Biol.*, **112**, 535–542.

[18] W. Hasel, T. .F. Hendrikson, and W. C. Still (1988) A rapid approximation to the solvent accessible surface areas of atoms. *Tetrahed. Comp. Method*, **1**, 103–116.

[19] T. Schlick (2002). *Molecular Modeling and Simulation*, (New York: Springer-Verlag).

[20] P. S. Bradley, O. L. Mangasarian, and W. N. Street (1997). Clustering via concave minimization. In Michael C. Mozer, Michael I. Jordan, and Thomas Petsche, editors, *Advances in Neural Information Processing Systems*, volume 9, page 368. The MIT Press.

[21] K. P. Bennett and O. L. Mangasarian (1993). Bilinear separation of two sets in n-space. *Comput. Optim. Appl.*, **2**(3), 207–227.

[22] S. R. Comeau, D. W. Gatchell, S. Vajda, and C. J. Camacho (2004). ClusPro: an automated docking and discrimination method for the prediction of protein complexes. *Bioinformatics*, **20**(1), 45–50.
[23] R. H. Tütüncü, K. C. Toh, and M. J. Todd (2003). Solving semidefinite-quadratic-linear programs using SDPT3. *Math. Program.*, 95(2, Ser. B):189–217. Computational semidefinite and second order cone programming: the state of the art.
[24] C. Helmberg, F. Rendl, R. J. Vanderbei, and H. Wolkowicz (1996). An interior-point method for semidefinite programming. *SIAM J. Optim.*, **6**(2), 342–361.
[25] M. Kojima, S. Shindoh, and S. Hara (1997). Interior-point methods for the monotone semidefinite linear complementarity problem in symmetric matrices. *SIAM J. Optim.*, **7**(1), 86–125.
[26] R. D. C. Monteiro (1997). Primal-dual path-following algorithms for semidefinite programming. *SIAM J. Optim.*, **7**(3), 663–678.
[27] M. Welch, N. Chinardet, L. Mourey, C. Birck, and J.P. Samama (1998). Structure of the CheY-binding domain of histidine kinase CheA in complex with CheY. *Nat. Struct. Biol.*, **5**, 25–29.
[28] S. Arold, P. Franken, M.P. Strub, F. Hoh, S. Benichou, R. Benarous, and C. Dumas (1997). The crystal structure of HIV-1 Nef protein bound to the Fyn kinase SH3 domain suggests a role for this complex in altered T cell receptor signaling. *Structure*, **5**, 1361–1372.
[29] Y. Tsunogae, I. Tanaka, T. Yamane, J. Kikkawa, T. Ashida, C. Ishikawa, K. Watanabe, S. Nakamura, and K. Takahashi (1986). Structure of the trypsin-binding domain of Bowman-Birk type protease inhibitor and its interaction with trypsin. *J. Biochem (Tokyo)*, **100**, 1637–1646.
[30] A. M. Buckle, G. Schreiber, A. .R. Fersht (1994). Protein-protein recognition: crystal structural analysis of a barnase-barstar complex at 2.0-Å resolution. *Biochemistry*, **33**, 8878–8889.
[31] M. Marquart, J. Walter, J. Deisenhofer, W. Bode, R. Huber (1983). The geometry of the reactive site and the peptide groups in trypsin, trypsinogen, and its complexes with inhibitors. *Acta Crystallogr., Sec. B*, **39**, 480–490.