Parsimonious Regularization using Genetic Algorithms Applied to the Analysis of Analytical Ultracentrifugation Experiments

Emre H Brookes
Department of Computer Science
University of Texas at San Antonio
ebrookes@cs.utsa.edu

Borries Demeler
Department of Biochemistry
University of Texas Health Science Center at San Antonio
demeler@biochem.uthscsa.edu

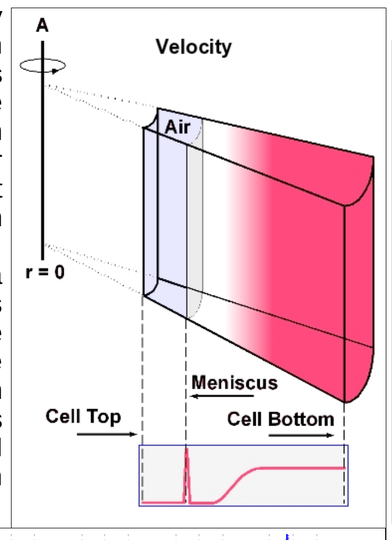
ABSTRACT

Analytical Ultracentrifugation (AUC) is an experimental technique used to determine shape and molecular weight of biological macromolecules and synthetic polymers in solution. Finding the best fit model for AUC experimental data is a difficult inverse problem complicated by presence of noise. Eliminating the effects of noise traditionally involves the use of Tikhonov or Maximum-Entropy regularization. These methods introduce a bias which smoothes the solution and thus falsely increases number of molecules in the model. We apply Genetic Algorithms to determine a parsimonious model with a goodness-of-fit approximating the level of noise present in the data.

ANALYTICAL ULTRACENTRIFUGATION

Analytical Ultracentrifugation (AUC) is a powerful technique for studying macromolecular systems in solution [1,2,3]. This method can be used to follow assembly processes of multi-enzyme complexes, characterize recombinant proteins, asses composition and characterize macromolecular mixtures that are heterogeneous in mass and shape. The techniques addressed in the poster are currently being used in studies focusing on macromolecular properties of biological systems involved in disease, cancer and aging, and on synthetic polymers, colloids and crystals of interest to material science and physics.

In AUC sedimentation velocity experiments a sample in solution contained in a sector shaped cell is placed in the ultracentrifuge. The ultracentrifuge runs at speeds from 2,000 to 60,000 RPM. At regular time intervals, the instrument records a radial concentration profile of the cell as shown in *Figure 1*. The experiment starts with a uniformly distributed sample. As time progresses, the sample sediments towards the bottom of the cell. The data obtained from such an experiment is typically shown as a superposition of the radial concentration profiles as shown in *Figure 2*.



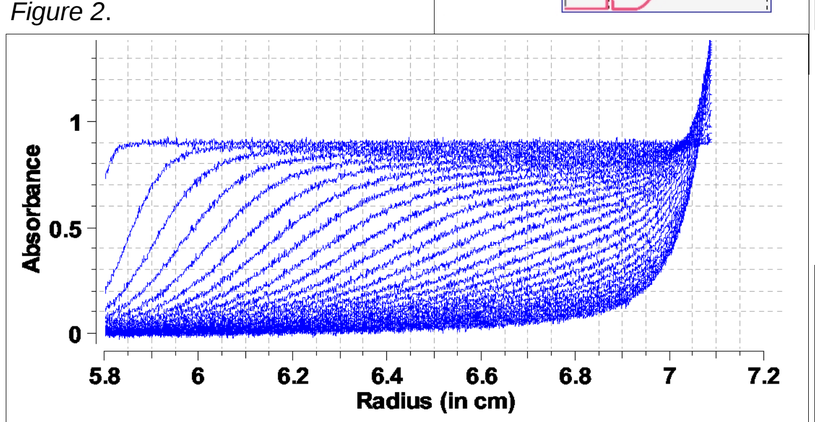


Figure 2 - Experimental data

The sample may contain several solutes, each a different type of molecule present at some concentration. Each solute's behaviour in the ultracentrifuge is described by a PDE known as the Lamm equation [4]. The Lamm equation is parameterized by two values: *s* - the sedimentation coefficient and *k* - the frictional ratio. *s* describes the rate of a solute's sedimentation and *k* is a measure of its shape as shown in *Figure 3*.

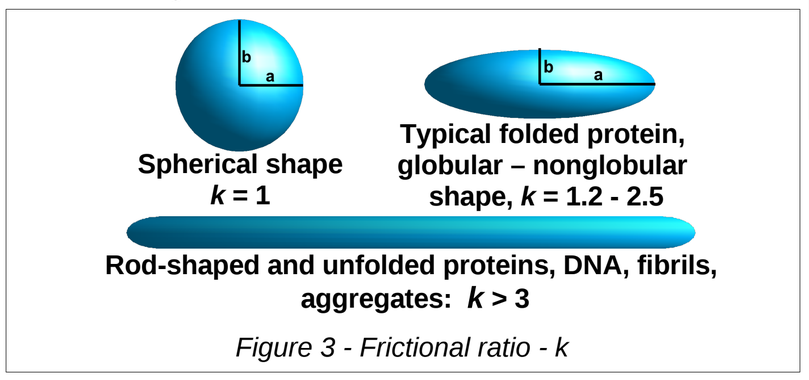


Figure 3 - Frictional ratio - k

Superposition of Lamm equation solutions holds for multiple non-interacting solutes. Given *s, k,* speed, solvent viscosity and density, solute specific volume and temperature the molecular weight (MW) can be computed. The simulated noisy experimental data in *Figure 2* contains three solutes with values from *Table 1.*

MW	s coefficient	frictional ratio *k*	concentration
1e4	1.3269e-13	1.3139	0.3
2e4	2.7675e-13	1	0.2
4e4	1.9214e-13	2.2865	0.4

Table 1 - Target values for a 3 solute system

It is straightforward to produce simulated experimental data from the table above, but it is much more difficult to determine the table values from the experimental data. It is very difficult to determine even the number of different types of solutes present. Knowing the number of solutes, their molecular weights, concentrations, and shapes is of primary importance to the researcher. Our techniques address these problems.

INVERSE PROBLEM

To determine the number of solutes and their *s* and *k* values present in experimental data in vector **b** consists of the following steps:

1. Build a set *S* of likely solute parameter *(s,k)* pairs.
2. Solve the Lamm equation for each element of *S* and place these solutions into the columns of a matrix **A**.
3. Use a nonnegatively constrained least-squares (NNLS [5]) method to find the best fit combination of columns of **A** to the experimental data vector which

This basic procedure results in a vector **x** which contains zero for the solute parameters which do not contribute to the best-fit solution and a positive value for contributing solutes. The positive elements of **x** contain the concentration of the respective solute. We denote the number of positive elements of **x** by *nz*(**x**). The goodness-of-fit of **Ax** to **b** is measured by root mean square deviation (RMSD). Step 3 can be modified to support Tikhonov or Maximum-Entropy regularization [6], which generally increases *nz*(**x**).

The above procedure to determine the best fit model for a given *S* is typical of two methods that have been used to solve this problem. The key difference is determining the set *S* of step 1. In the *C(s)* [7] method, *S* consists of a one dimensional grid of *s* values with a fixed *k* and then a one dimensional line search is performed over *k* minimizing RMSD. This method suffers from the inability to find systems in which solutes exhibit heterogeneity in *k*.

In another method known as the two dimensional spectrum analysis (2DSA) [8], *S* contains pairs from a two dimensional grid placed on the *(s,k)* plane. A 2DSA of our experimental data produced the results in *Figure 4* below where the blue dots represent solutes present in the best fit solution and the Xs represent the target solutes from *Table 1*.

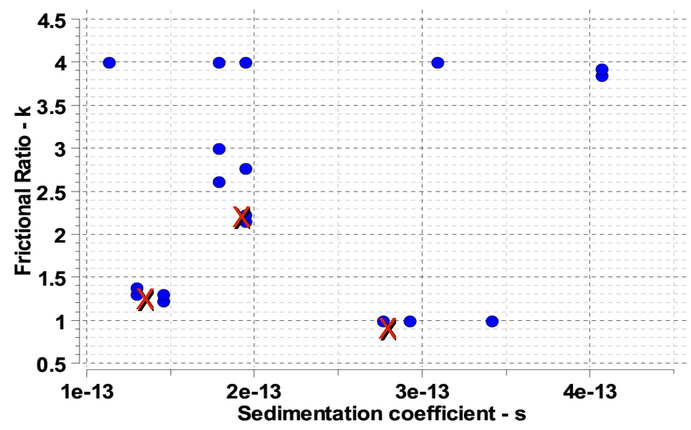


Figure 4 - 2DSA results

It can be seen from *Figure 4* that along with the correct solutes, there are many false positives. *nz*(**x**)=18 >> 3 for this analysis. 2DSA does a reasonable job of determining the correct molecular weights as is shown in *Figure 5* below where Xs again represent the target values from *Table 1*. 2DSA did not find the correct solute concentrations.

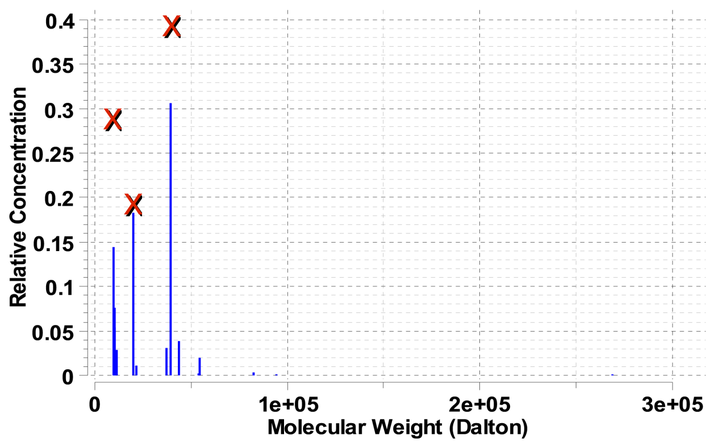


Figure 5 - 2DSA results - MW

GENETIC ALGORITHM METHOD

For our genetic algorithm method, each individual of our population is a set *S*. To compute the fitness of each individual, we compute the RMSD using same basic procedure as used in *C(s)* and 2DSA. In addition, we use *nz*(**x**) as a penalty factor to impact the fitness of individuals which have larger numbers of solutes in their best fit model.

Population initialization is critical to good performance. 2DSA is used to constrain population initialization and mutation. For each solute identified by 2DSA we place buckets which constrain the mutation range. This is shown in *Figure 6* below where the blue dots represent 2DSA solutions, the Xs represent target solutions (not representative of *Table 1* for this figure only), and the green boxes represent the bucket constraints for associated solutes in the individual.

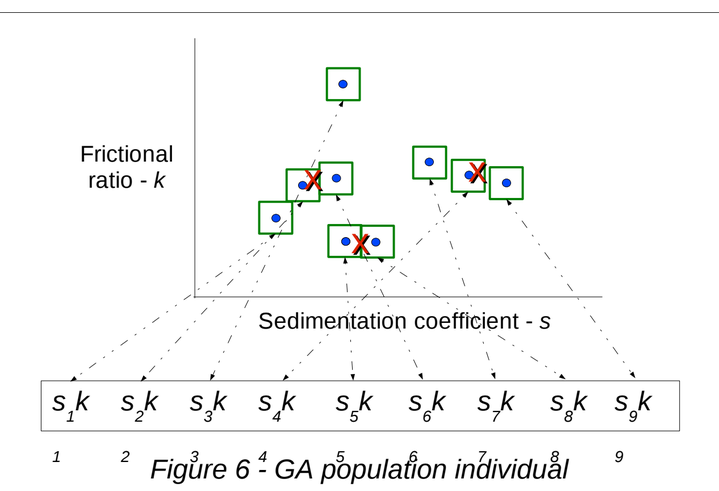


Figure 6 - GA population individual

Starting from the 2DSA results of *Figure 4* we produce buckets as shown in green in *Figure 7* below.

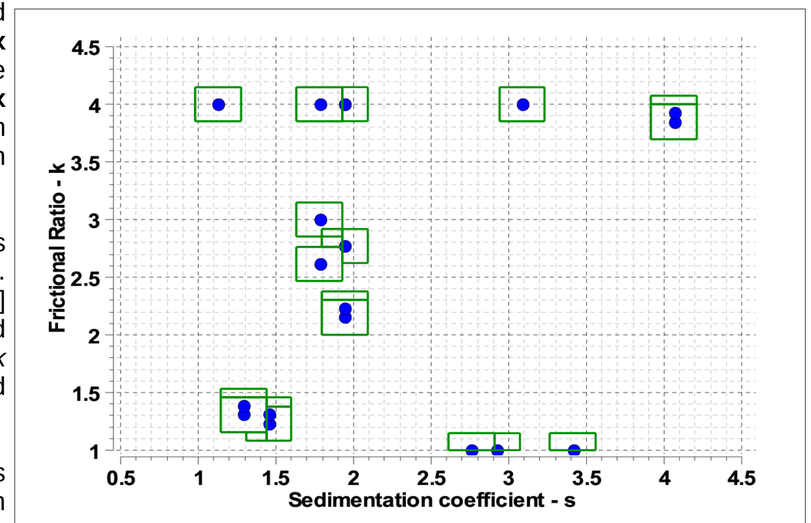


Figure 7 - Buckets for GA

Then we ran 100 Monte Carlo iterations of the GA using parsimonious regularization with produced the results shown in *Figures 8 and 9*.

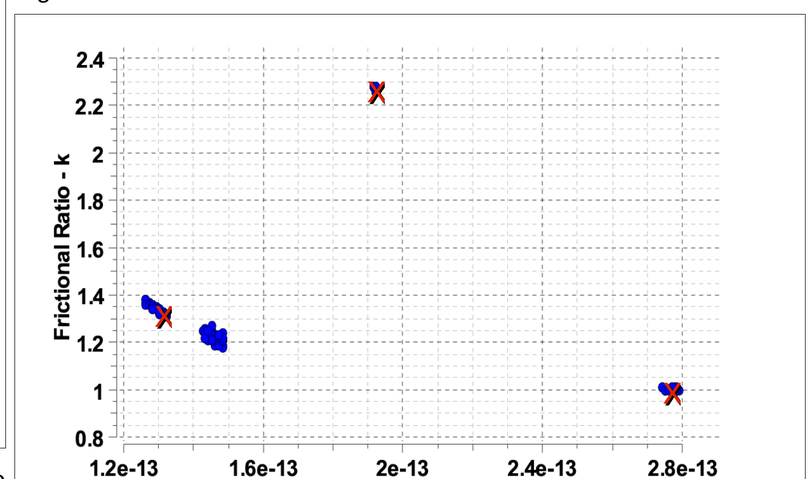


Figure 8 - GA results

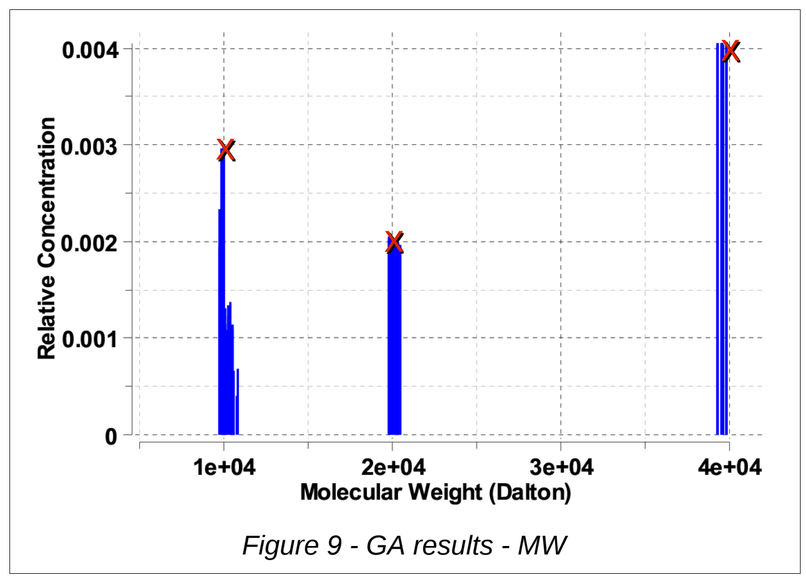


Figure 9 - GA results - MW

Our method correctly identified the concentrations and molecular weights providing a solution superior to 2DSA alone. The difference is summarized in the following table:

Method	No. of Parameters	RMSD
2DSA	18	4.513e-3
GA	3.62	4.542e-3

We have presented a new method using a GA with regularization for parsimonious solutions in AUC. The method correctly indicates the number of solute parameters present in the data with identical fitness to much less parsimonious solutions. False positives are removed and the information in the final result closely matches the original model used to produce the experimental data. This is of major importance to the researcher, as this is the first known method to do so and can subsequently give the most accurate molecular weight and shape determinations.

ACKNOWLEDGEMENTS

We would like to thank Jeremy Mann for assistance with the BCF Linux cluster.

We gratefully acknowledge the support by the Kleberg Foundation. This research has been supported by NSF Grant DBI-9974819, NIH-RRR022200 and the San Antonio Life Science Institute with Grant #10001642, all to B.D.

REFERENCES

[1] J. L. Cole and J. C. Hansen. Analytical ultracentrifugation as a contemporary biomolecular research tool. In *J. Biomolecular Techniques*, volume 10, pages 163-174, 1999.

[2] B. Demeler. Hydrodynamic Methods. In *Bioinformatics Basics: Applications in Biological Science and Medicine*. 2nd Edition, pages 226-255. CRC Press LLC, 2005.

[3] B. Demeler and K. E. van Holde. Sedimentation velocity analysis of highly heterogeneous systems. In *Anal. Biochem.*, volume 335, pages 279-288, 2004.

[4] O. Lamm. Die differentalgleichung der ultrazentrifugierung. In *Ark. Mat. Astrol. Fys*., volume 21B, pages 1-4, 1929.

[5] C. L. Lawson and R. J. Hanson. *Solving Least Squares Problems*. Prentice Hall, New Jersey, 1974.

[6] R. C. Aster, B. Borchers, and C. H. Thurber. *Parameter Estimation and Inverse Problems*. Elsevier Academic Press, London, 2005.

[7] P. Schuck. Size-distribution anal. of macromolecules by sedimentation velocity ultracentrifugation and lamm equation modeling. In *Biophys. J.*, volume 78, pages 1609-1619, 2000.

[8] E. H. Brookes, R. V. Boppana, and B. Demeler. Computing large sparse multivariate optimization problems with an application in biophysics. In *SuperComputing 2006 Conference Proceedings*. ACM, IEEE, November 2006.