Modeling of Data

A *first principles* approach allows us to model the experimental data by fitting it to a mathematical model. The model represents the physics of the experiment, and contains parameters of interest to the experimentalist. We need to find the values of these parameters by adjusting the model so it matches the data. This is a hard problem called an "*inverse problem*" that requires optimization (fitting) algorithms which aid us in adjusting the parameters so the model fits the data.

In UltraScan this is accomplished by a least squares fitting approach that compares each data point from the model with the corresponding point in the experimental data:

$$Minimize \sum_{i=1}^{N} \left(Data_i - Model_i \right)^2 \quad (i \text{ over radius and time})$$

Optimally, the difference is zero, but because of experimental noise this never happens, since the model is noise free.

Summary:

Time- and radially- invariant noise can be fitted by UltraScan and removed from the data to improve the results.

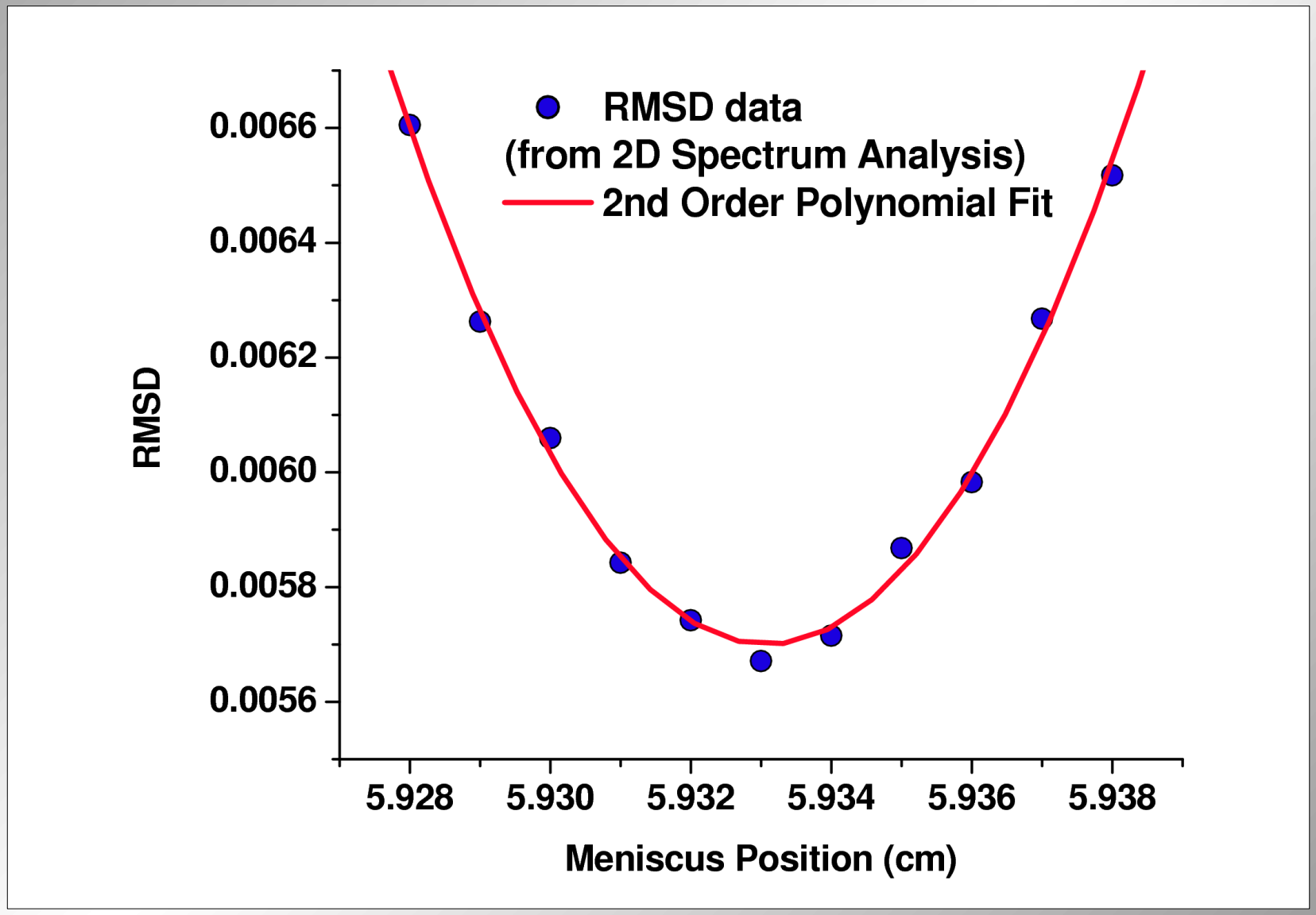
Stochastic noise cannot be removed and should be minimized by maintaining a well calibrated instrument and performing a well-designed experiment!

Data subtraction in absorbance mode convolutes two stochastic vectors and leads to an increase in stochastic noise by $\sim\sqrt{2}$

Remember:

you cannot get reliable answers if you start with low quality input data!

Factors that affect Accuracy - Meniscus



Modeling Flow with the Lamm Equation

$$\left(\frac{\partial C}{\partial t}\right)_r = \frac{-1}{r}\frac{\partial}{\partial r}\left[s\omega^2 r^2 C - D r \frac{\partial C}{\partial r}\right]_t$$

Concentration **Sedimentation** **Diffusion**

The Lamm Equation describes the flow of a single solute in the sector- shaped analytical ultracentrifugation cell over time and radius. This allows us to simulate an entire experiment from start to finish.

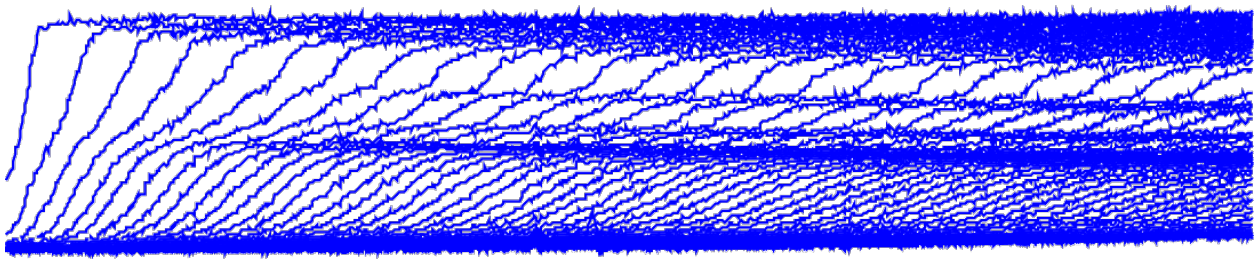
To solve this equation we use the finite element method. This method discretizes the two independent variables, the radius and the time.

This way we can calculate the concentration of the solute during the experiment for each radial point at each time point (scan).

Multiple non-interacting solutes are modeled by summing the results from two independent simulations.

Cao W., Demeler B. Modeling analytical ultracentrifugation experiments with an adaptive space-time finite element solution of the Lamm equation. (2005) Biophys J. 89(3):1589-602.

Lamm Equation for Non-interacting Systems:



Lamm equation *L(s, D, C)* for a single ideal solute:

$$\left(\frac{\partial C}{\partial t}\right)_r = \frac{-1}{r}\frac{\partial}{\partial r}\left[s\omega^2 r^2 C - D r \frac{\partial C}{\partial r}\right]_t$$

Concentration **Sedimentation** **Diffusion**

Lamm equation for a mixture of non-interacting solutes:

$$C = \sum_{i=1}^{n} c_i L(s_i, D_i)$$

Lamm Equation for Interacting Systems

Lamm equation *L(s, D, C)* for a single ideal solute:

$$\left(\frac{\partial C}{\partial t}\right)_r = \frac{-1}{r}\frac{\partial}{\partial r}\left[s\omega^2 r^2 C - D r \frac{\partial C}{\partial r}\right]_t$$

Concentration **Sedimentation** **Diffusion**

Lamm equation for an interacting system (e.g., monomer-dimer, mass action applies):

$$M + M = D \qquad K_a = \frac{[D]}{[M]^2}$$

$$C = \left[L\left(\bar{s}, \bar{D}\right)\right]_{r,t} \qquad \bar{s} = \frac{\sum_{j=1}^{m} s_j C_j}{C_T} \qquad \bar{D} = \frac{\sum_{j=1}^{m} D_j\left(\partial C_j/\partial r\right)}{\sum_{j=1}^{m}\left(\partial C_j/\partial r\right)}$$

Optimization and Analysis Methods for Sedimentation Velocity

2-dimensional Spectrum Analysis (2DSA): High-resolution, general and model-independent solution for size and anisotropy distributions of non-interacting systems

Parametrically Constrained Spectrum Analysis (PCSA): Identifies size/anisotropy relationships for polymerizing systems and provides a constrained fit over the 2-dimensional sedimentation/diffusion space.

Custom Grid Analysis (CG): Takes advantage of prior knowledge to parameterize the 2DSA grid in terms of alternate hydrodynamic variables.

(Discrete Model) Genetic Algorithms (GA): Robust non-linear least squares optimization method that provides parsimonious regularization of 2DSA spectra. Also used for fitting of discrete, non-linear models (reversible association, non-ideality, co-sedimenting solvents).

Monte Carlo Analysis (MC): Used to measure the effect of noise on the fitted parameters, yields parameter distribution statistics

van Holde – Weischet Method (vHW): Used to generate diffusion-corrected sedimentation profiles which provide finely detailed comparisons between multiple samples.

C(s), C(s, f/f0), C(s, M): Low resolution methods - not used in UltraScan.

Nonlinear Least Squares Finite Element Fitting

Direct Boundary fitting uses a nonlinear least squares minimization approach to fit a model function (a sum of Lamm equations) Y* to an experimental dataset Y:

Our Model:

$$Y^* = \sum_{k=1}^{n} c_k L\left(s_k, D_k\right) + b$$

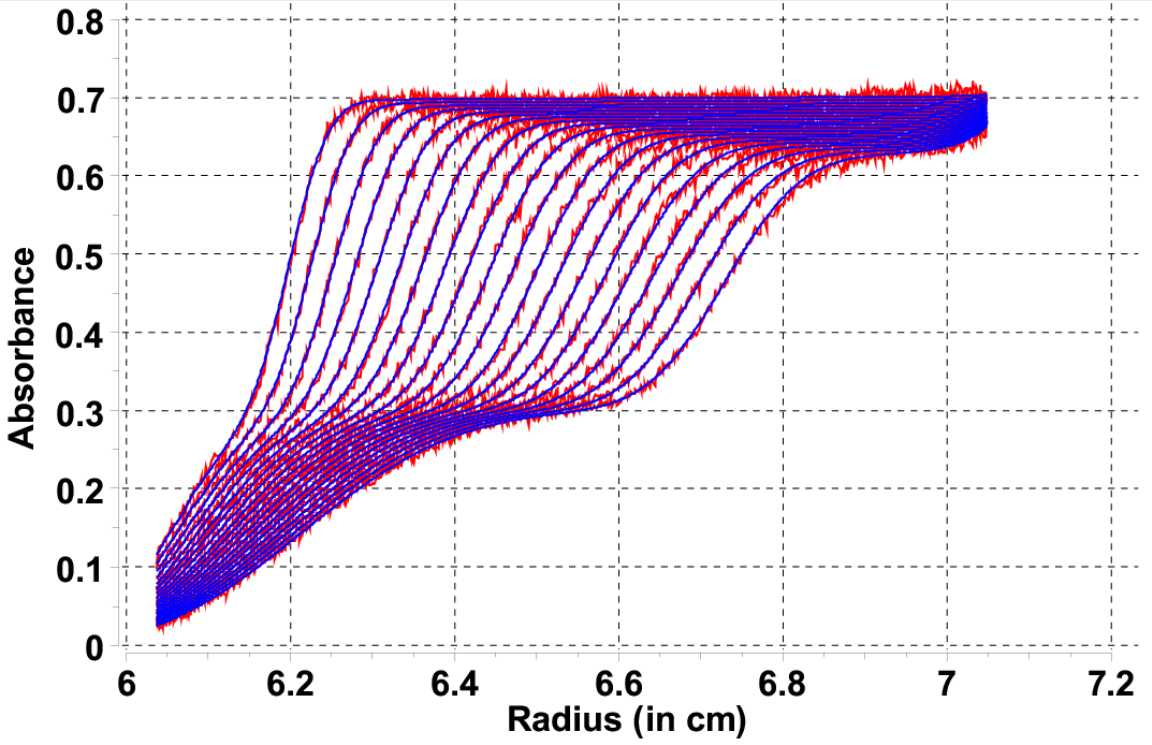
The model is compared to the experimental data in the least squares sense for each data point in the experiment (over time and radius)

$$Min \sum_{i=1}^{r} \sum_{j=1}^{t} \left[Y_{ij}^* - Y_{ij} \right]^2$$

here, c, b, s and D are nonlinear parameters, and are adjusted independently in an iterative fit (Svedberg, SedAnal, Lamm) .

Nonlinear Least Squares Finite Element Fitting

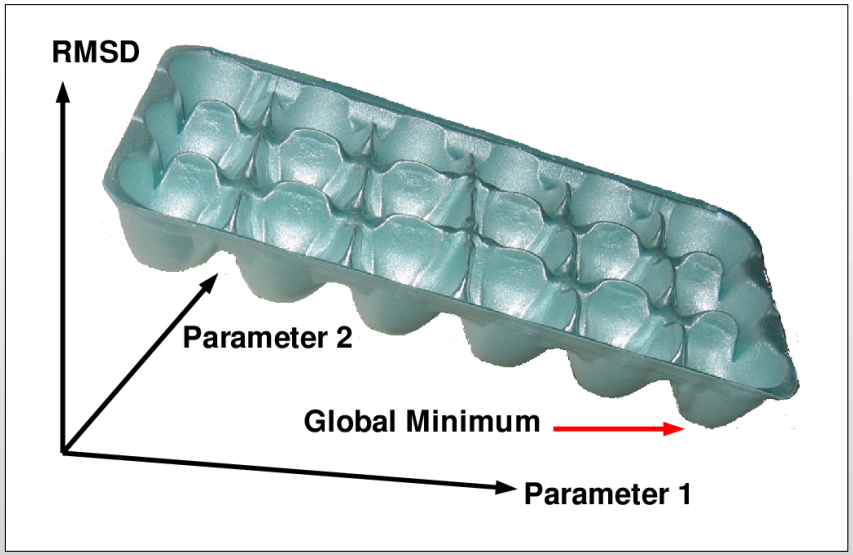
Finite Element - Nonlinear Least Squares (RMSD: 4.61 x 10^{-3})
Monte Carlo is needed to define statistical confidence of fitted parameters.



M_1: 128.8 kD (135.7 kD)
f/f_0: 3.10
s_1: 5.43 x 10^{-13}
D_1: 2.28 x 10^{-7}

M_2: 14.6 kD (14.3 kD)
f/f_0: 1.29
s_2: 1.71 x 10^{-13}
D_2: 1.02 x 10^{-7}

The Optimization Challenge:



Problem with nonlinear least squares optimization:

For multi-component systems, the nonlinear least squares fitting algorithm gets easily stuck in local minima and the solution depends on the starting points. Problem gets worse with more parameters (i.e., multiple components).

The Optimization Challenge:

1. For complicated problems, nonlinear optimization will fail and the fitting algorithm will not converge to the global optimum.

2. In addition, due to noise the solution will not be unique and there will be an infinite number of equally likely solutions with the same χ^2

How do we get around these problems?

Problem 1 can be alleviated by *linearizing* the problem

Problem 2 is intractable. The best we can do is to perform a statistical error analysis and use Monte Carlo methods.

C(s)/C(M) Method (P. Schuck)

Linearization Approach 1 – keeping a constant f/f_0 value:

Decomposition of the concentration function into a linear combination of orthogonal basis functions (Lamm equations) distributed over a partitioned s-value range and a constant frictional ratio $\Phi = f/f_0$:

$$C = \underbrace{c_1 L\left(s_1, D\left(s_1, \Phi\right)\right)}_{\text{Component 1}} + \underbrace{c_2 L\left(s_2, D\left(s_2, \Phi\right)\right)}_{\text{Component 2}} + \ldots$$

Fit only the amplitudes (c_j) of those components that make a non-zero contribution by doing a non-negatively constrained *linear* least squares fit over all components.

C(s)/C(MW) Method (P. Schuck)

Parameterization Approach:

Instead of using nonlinear fitting parameters s and D (which are required for the solution of the Lamm equation), we treat these parameters as constants. The s-value is partitioned over a range from $s_{min} < s < s_{max}$ in equi-distant intervals. Using the Stokes-Einstein relationship, the diffusion coefficient can be expressed as a function of the sedimentation coefficient and a constant frictional ratio $\Phi = f/f_0$

$$D = \frac{RT}{18\pi N(\Phi\eta)^{2/3}}\sqrt{\frac{2(1-\bar{\nu}\rho)}{s\bar{\nu}}}$$

This way, given an s-value and a fixed shape, a corresponding diffusion coefficient can be calculated for each s-value and the Lamm equation term for each species can be calculated. Then the only question remaining is the amplitude of each term, which is a linear fit, and the best match for k. The frictional ratio can be adjusted for a best fit average using a line search.

Schuck P. Size-distribution analysis of macromolecules by sedimentation velocity ultracentrifugation and Lamm equation modeling. Biophys. J. 78(3):1606-19, 2000

C(s)/C(MW) Method (P. Schuck)

Perform a *linear* fit using the NNLS method* and only fit the amplitudes c_j subject to the constraint $c_j \geq 0$

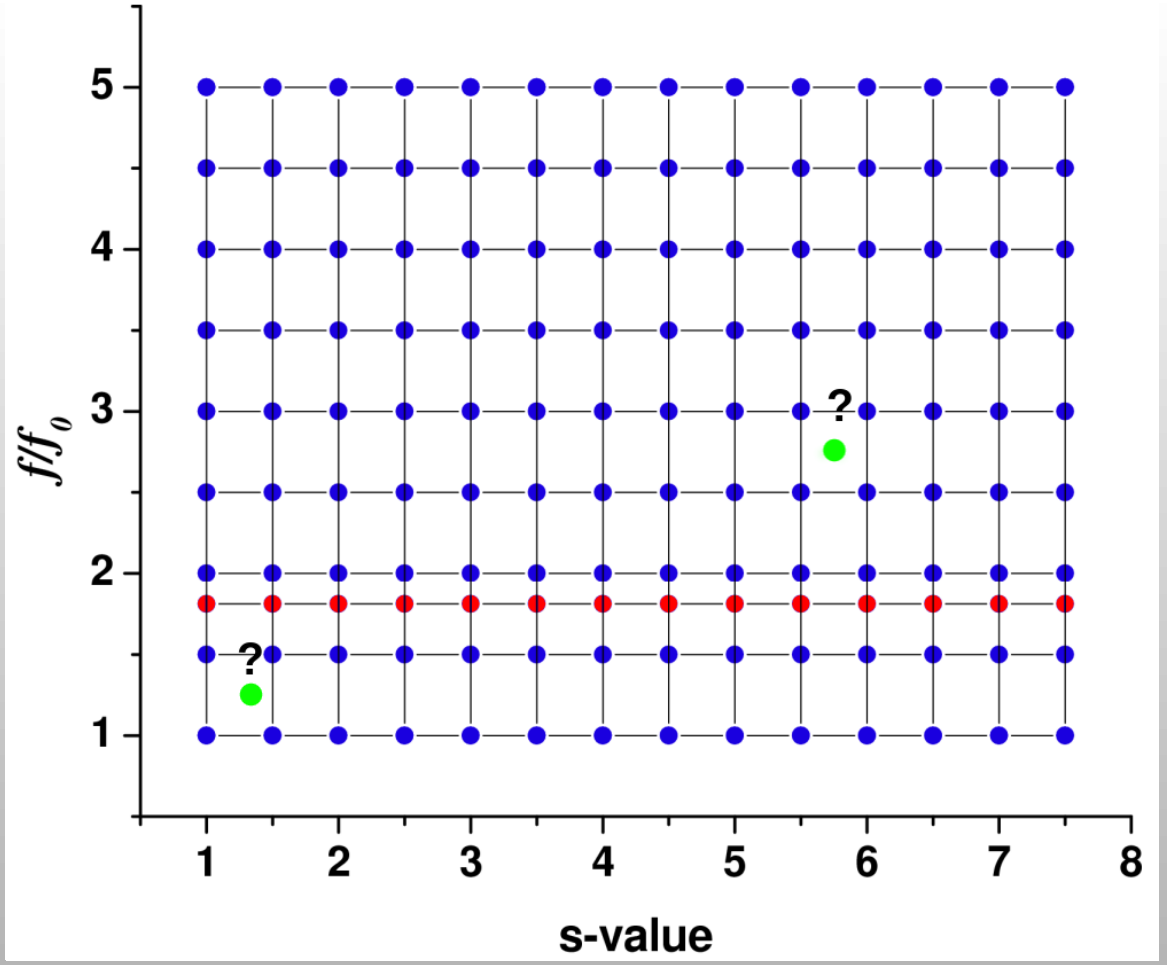
$$Min \sum_{i=1}^{m} \sum_{j=1}^{n} \left[c_j L\left(s_j, D\left(s_j\right)\right) - \boldsymbol{Y_i} \right]^2$$

Note: This will generate Lamm equations that have a fixed frictional ratio and a diffusion coefficient that is linked to the sedimentation coefficient.

ALL PARAMETERS EXCEPT THE AMPLITUDES ARE CONSTANT!

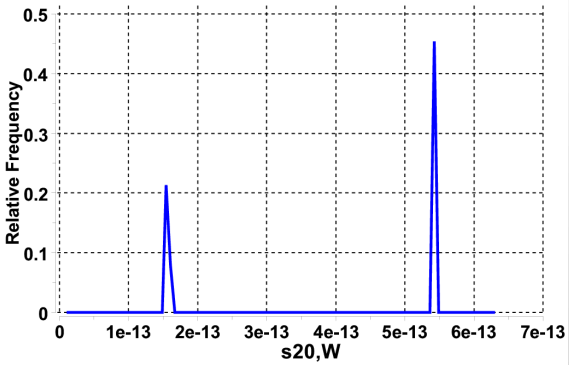
Lawson, C. L. and Hanson, R. J. 1974. Solving Least Squares Problems. Prentice-Hall, Inc. Englewood Cliffs, New Jersey

C(s)/C(MW) Method (P. Schuck)

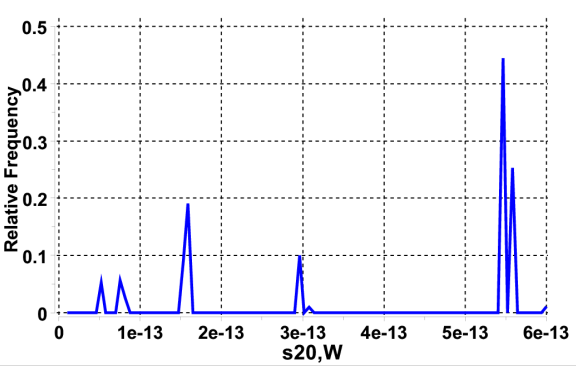


C(s)/C(MW) Method (P. Schuck)

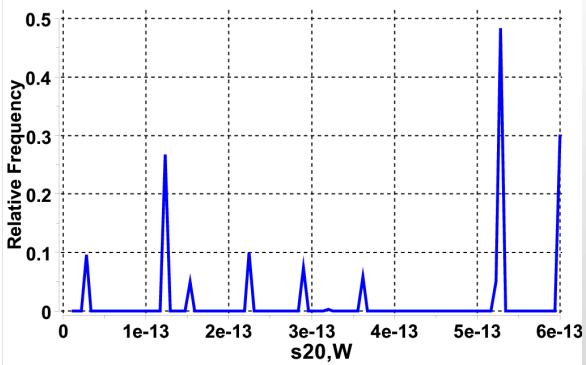
C(s) method (lowest f/f0): s_1: 5.43 x 10^{-13} (61 %), s_2: 1.56 x 10^{-13} (39 %)



f/f0 = 1.29 (Lysozyme)



f/f0 = 2.297 (fitted)



f/f0 = 3.10 (DNA)

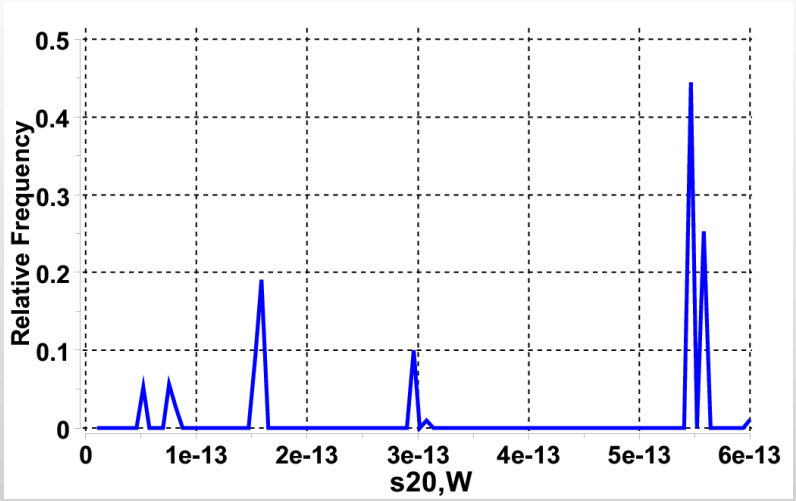
RMSD for C(s) fit: 6.0e-3, RMSD for FE fit: 4.61e-3

With increasing f/f0, the number of artifactual peaks increases
(regularization hides this problem)

Fitted f/f0 values provide an *average* of all components

C(s)/C(MW) Method (P. Schuck)

C(s) method (lowest f/f0): s_1: 5.43 x 10^{-13} (61 %), s_2: 1.56 x 10^{-13} (39 %)



f/f0 = 2.297 (fitted)

Lysozyme:

Molecular Weight = 30.1 kD
too high

DNA:

Molecular Weight = 120.6 kD
too low

Motivation: Wish List for an Optimal Method:

We need a method that satisfies the following criteria:

Generality – works accurately and reliably for ***any*** system

High resolution/high information content (s, D, partial conc., Kds)

Model independent – it needs to be able to find it's own model

Suitable for global fitting – can integrate other experiments

Always converges to the global minimum (overcomes the egg carton problem!)

Computationally efficient

2-Dimensional Spectrum Analysis

Solution: Allow for variation in f/f_0 as well.

This is now a very large problem, but one that can fortunately be calculated in a single iteration, with one Lamm equation for each coordinate point in the grid:

$$Y^* = \sum_{s=s_{min}}^{s_{max}} \sum_{k=1}^{k_{max}} c_{s,k} L\left[s, D(s,k)\right] + b \qquad Min \sum_{i=1}^{r} \sum_{j=1}^{t} \left[Y_{ij}^* - Y_{ij} \right]^2$$

$$Ax = b \qquad Lc = Y$$

Using **NNLS** for this problem guarantees $c_{s,k} > 0$

m = # of radial points * # of time points = 1000 * 100 = 100,000

n = # of sedimentation value grid points (~30 - 50)

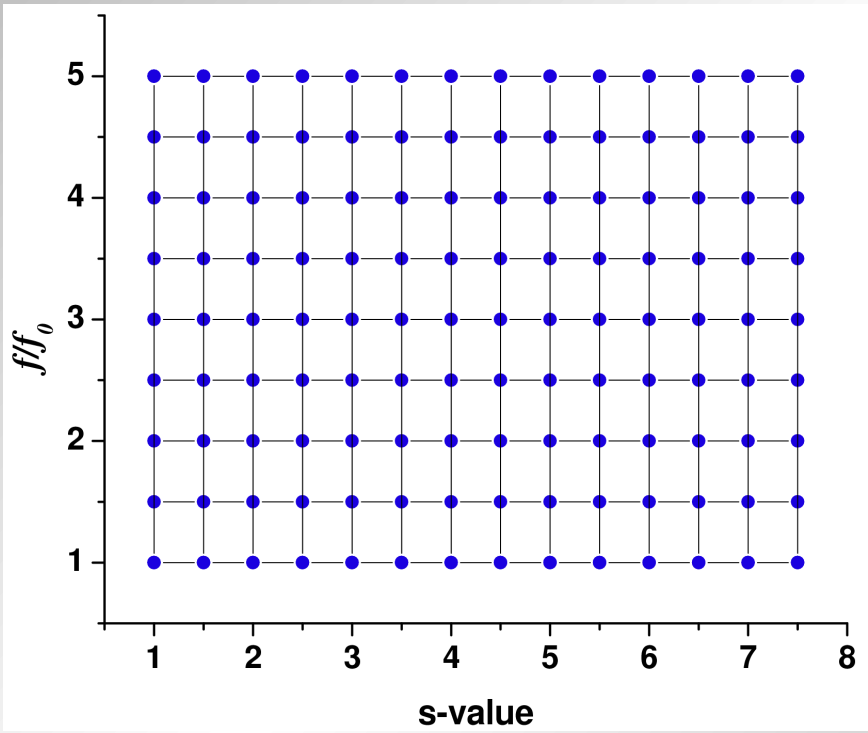
f = # of f/f0 value grid points (~30-50)

Total size: 250 million * 4 bytes/value + workspace, altogether > 1 GB

Brookes, E, Cao, W, Demeler, B. A two-dimensional spectrum analysis for sedimentation velocity experiments of mixtures with heterogeneity in molecular weight and shape. Eur Biophys J. 2010 39(3):405-14.

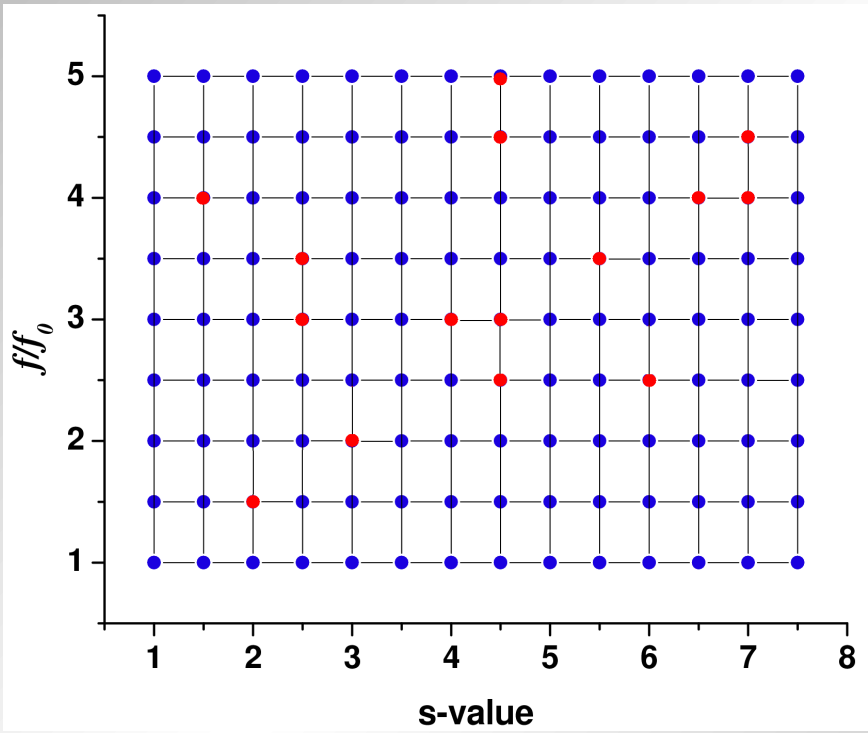
2-D Spectrum Analysis - Refinement:

Step 1: Start with original grid definition:



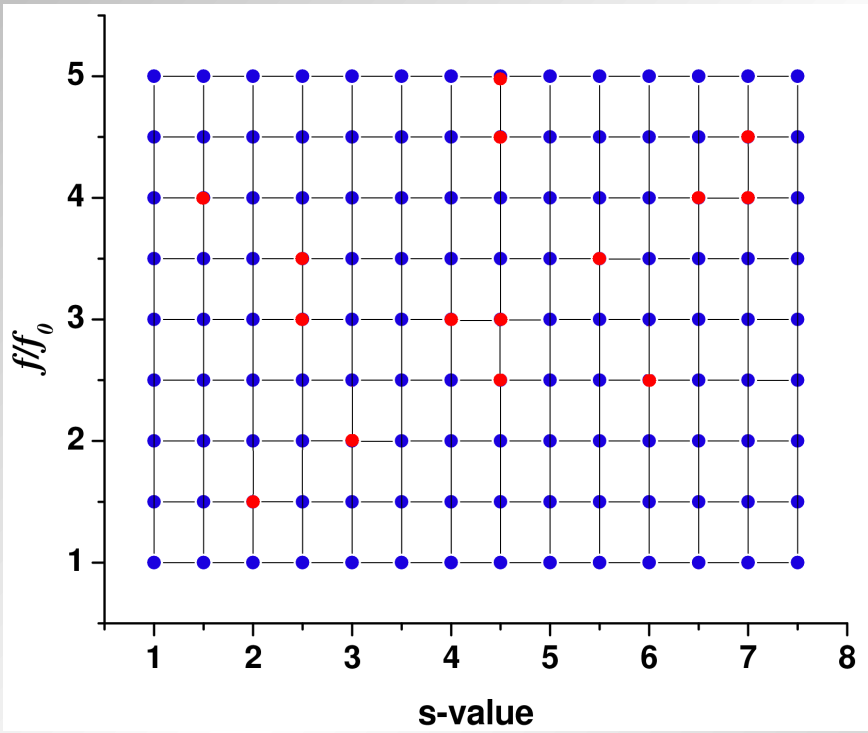
2-D Spectrum Analysis - Refinement:

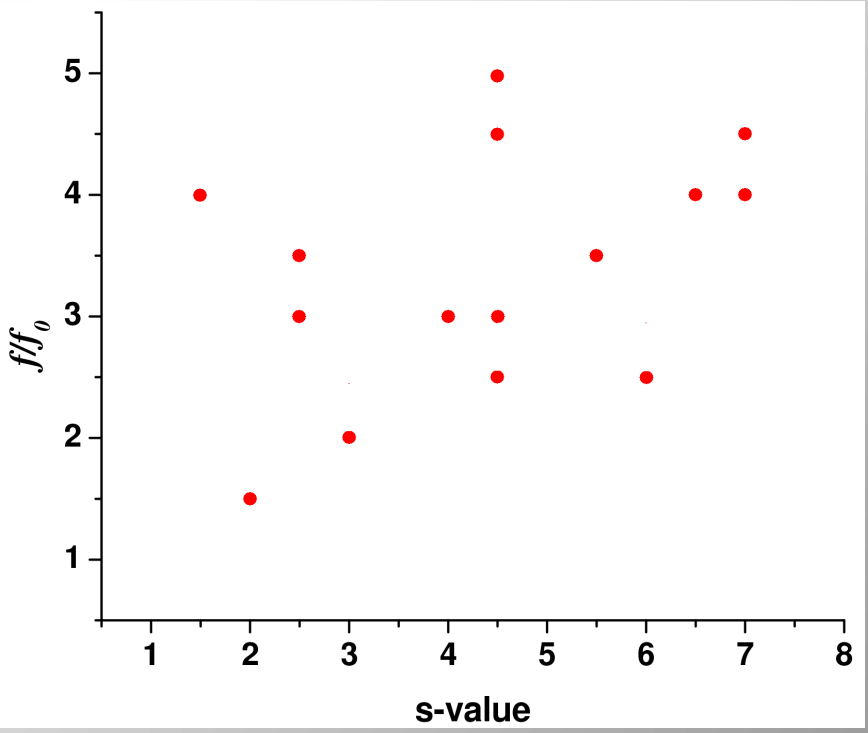
Step 2: Perform NNLS



2-D Spectrum Analysis - Refinement:

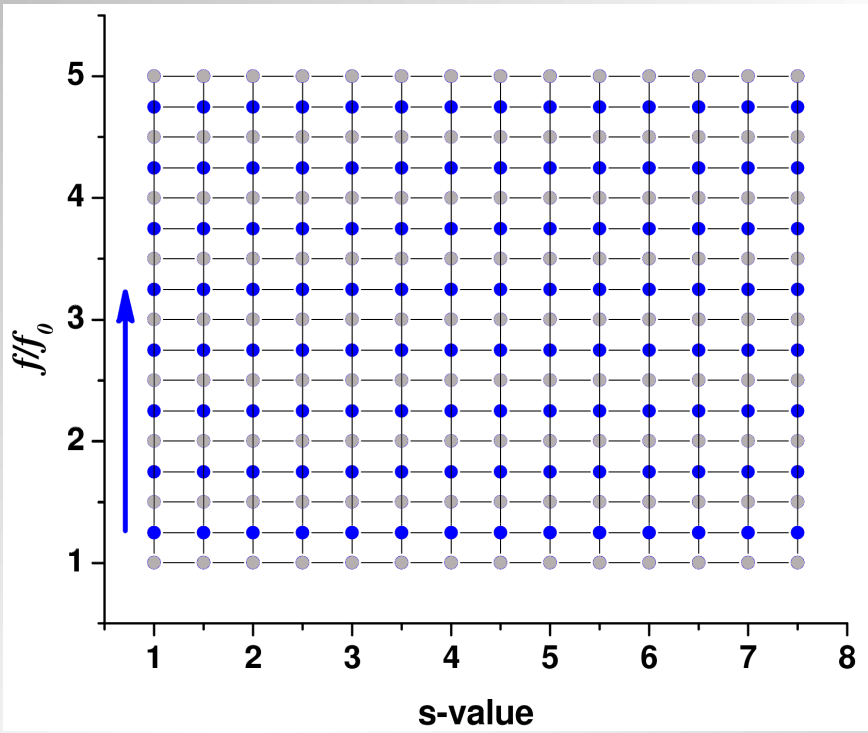
Step 3: Save non-zero elements into a separate array

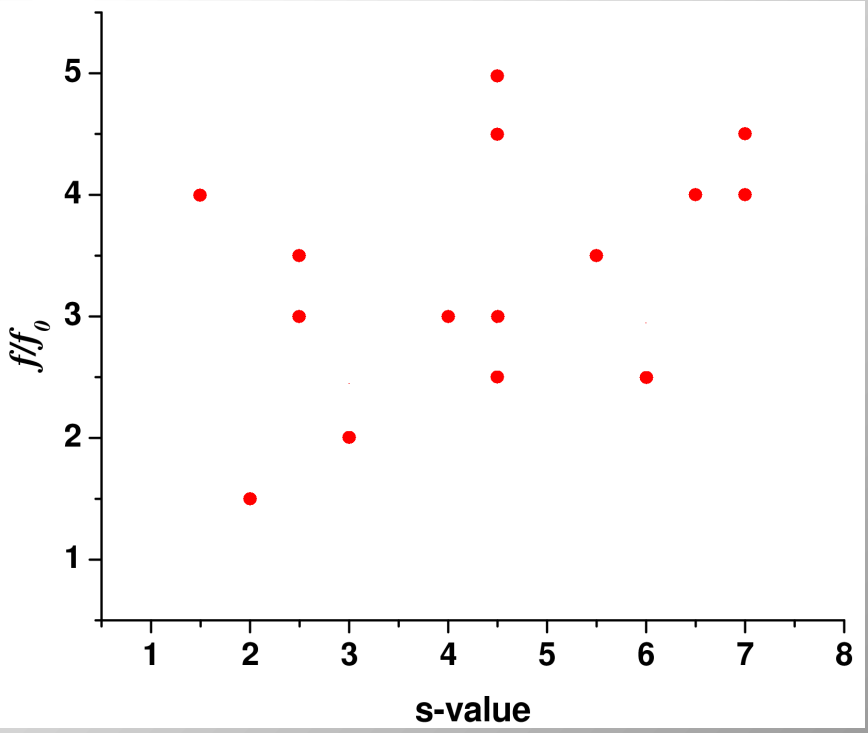




2-D Spectrum Analysis - Refinement:

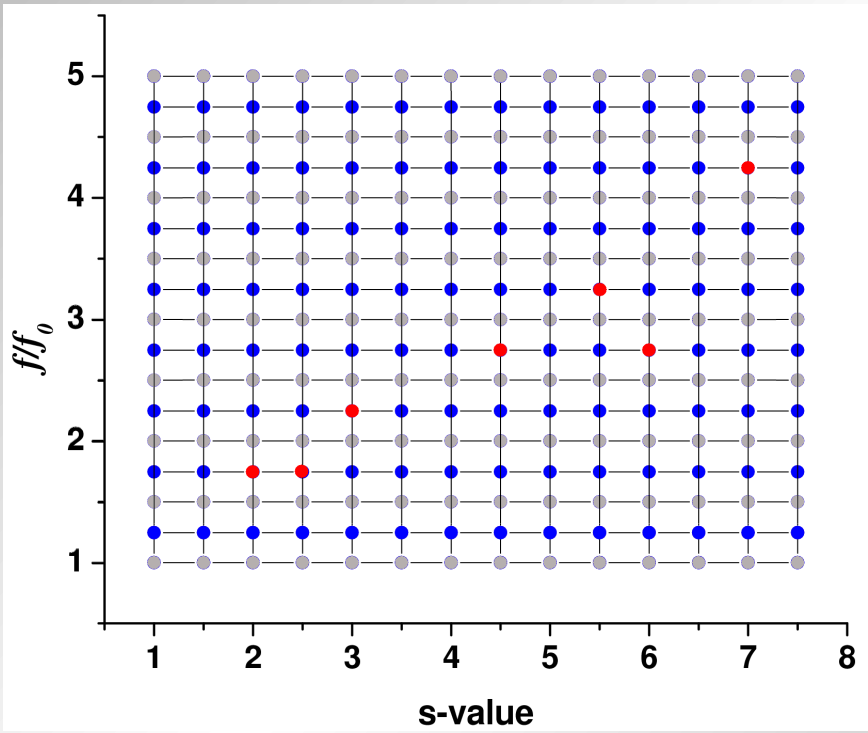
Step 4: Shift grid into Y-direction

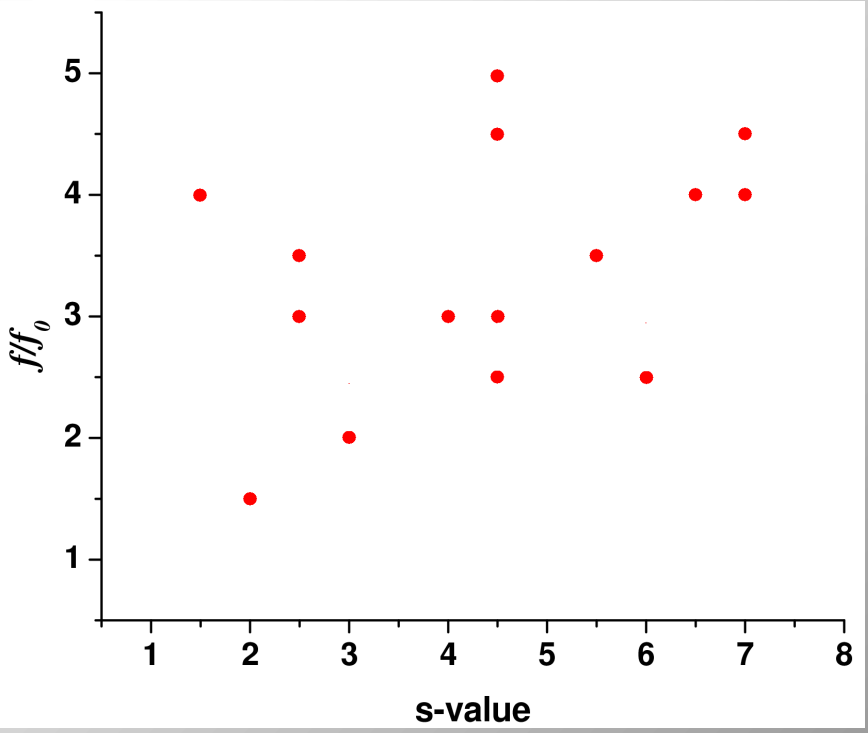




2-D Spectrum Analysis - Refinement:

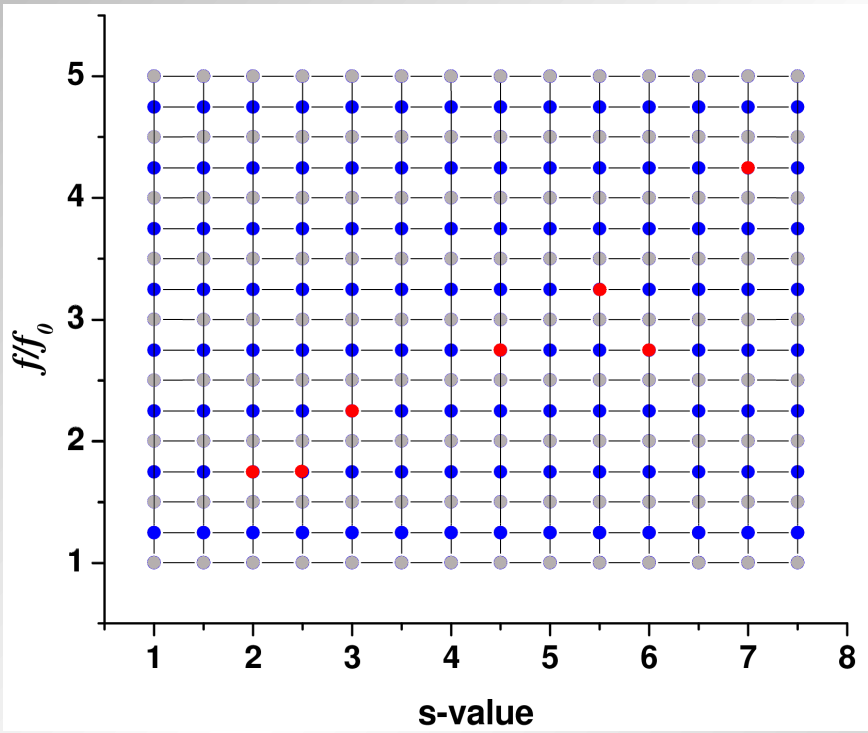
Step 5: Perform NNLS again, but only on the shifted grid (blue)

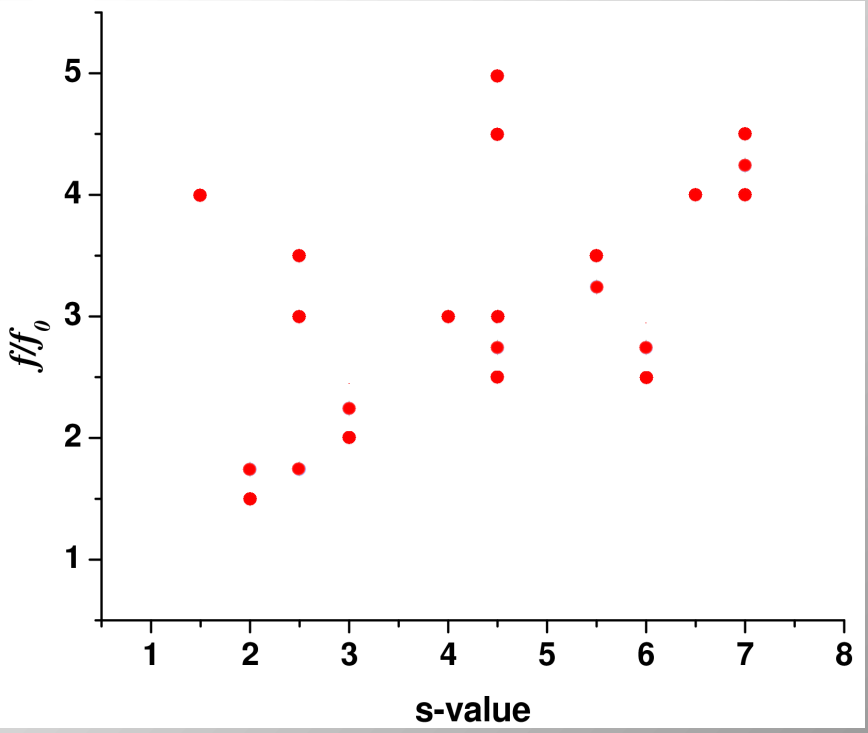




2-D Spectrum Analysis - Refinement:

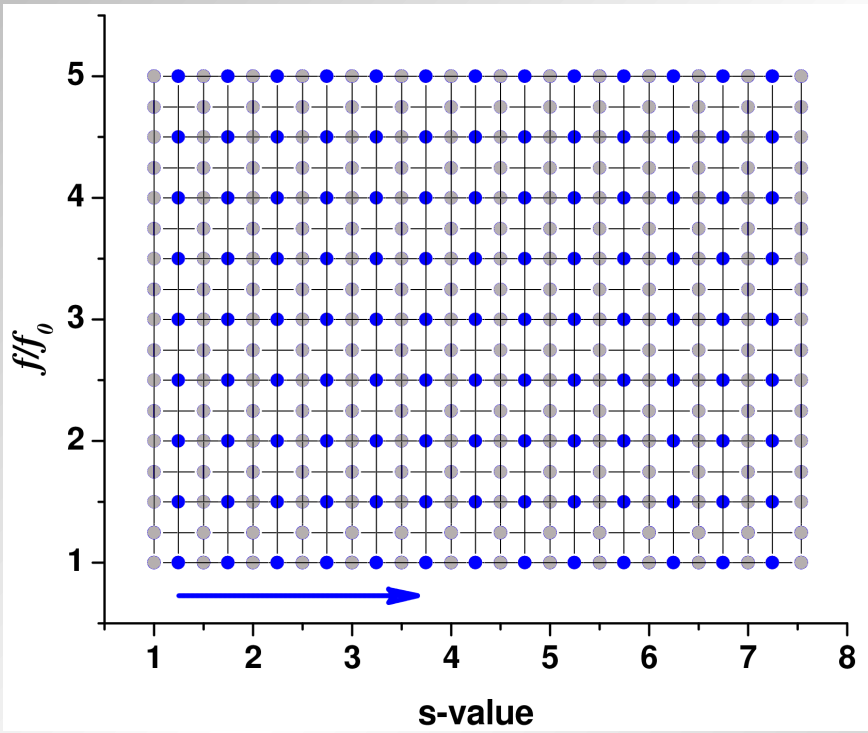
Step 6: Add the newly found non-zero elements to the stored array

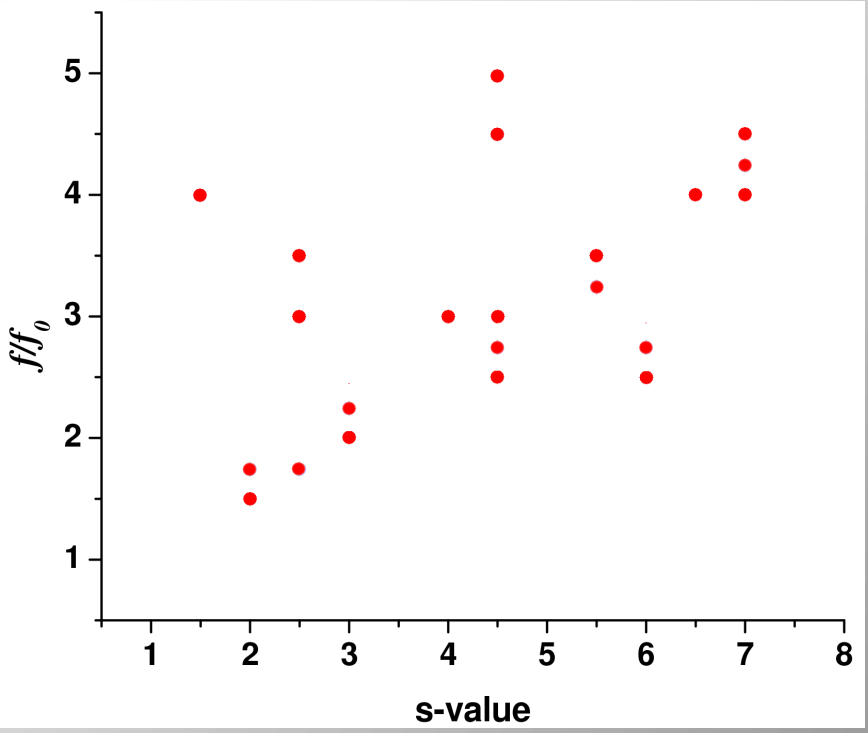




2-D Spectrum Analysis - Refinement:

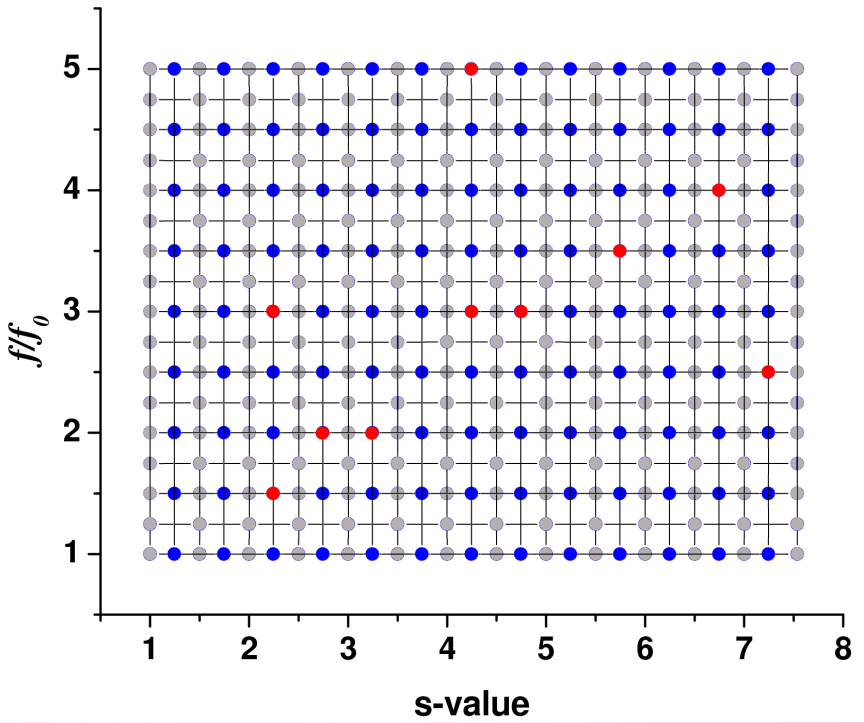
Step 7: Now shift the grid into the X-direction

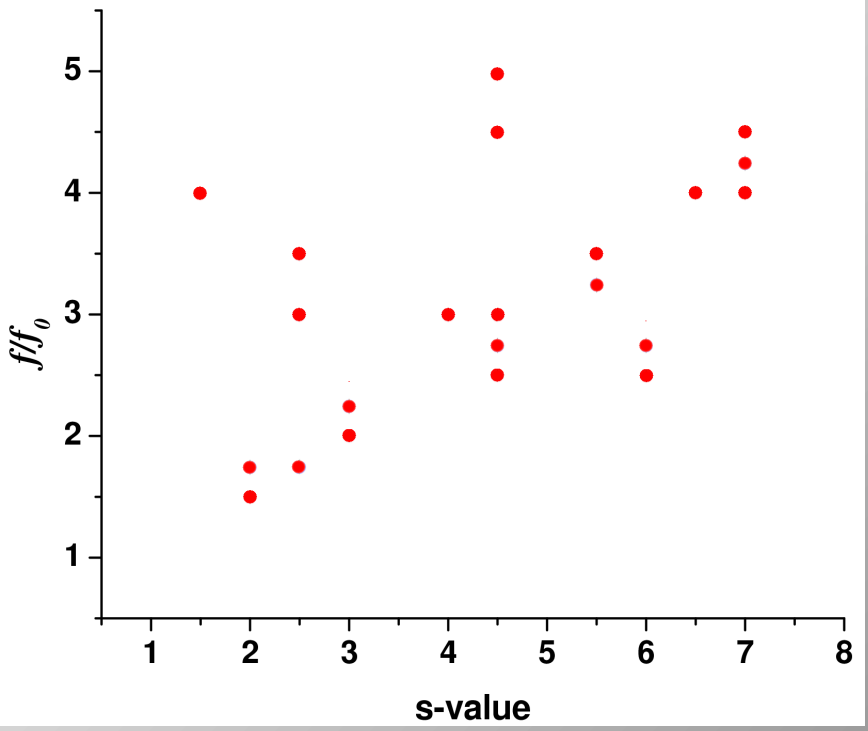




2-D Spectrum Analysis - Refinement:

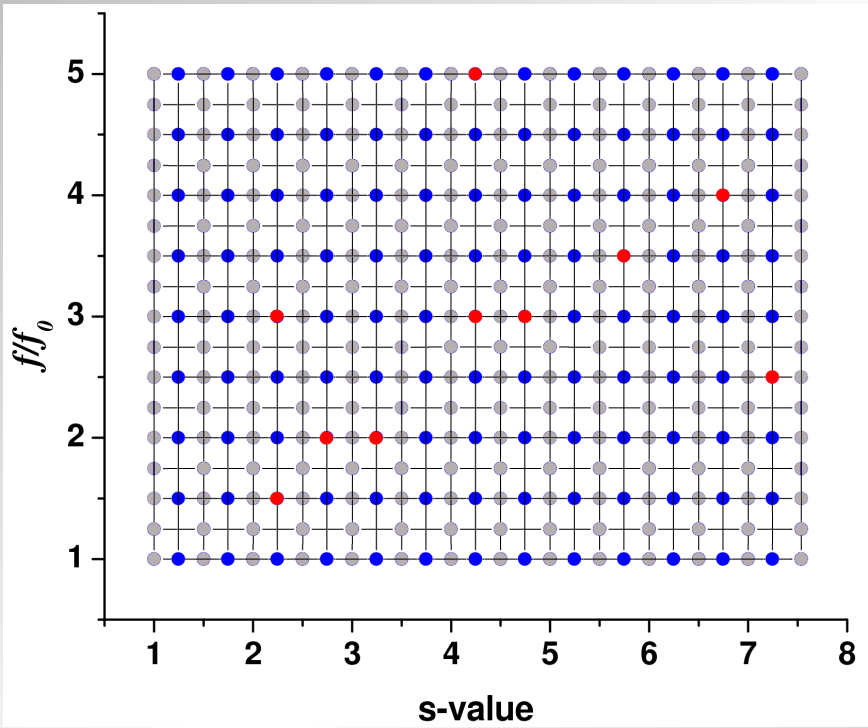
Step 8: Perform NNLS on the shifted grid again

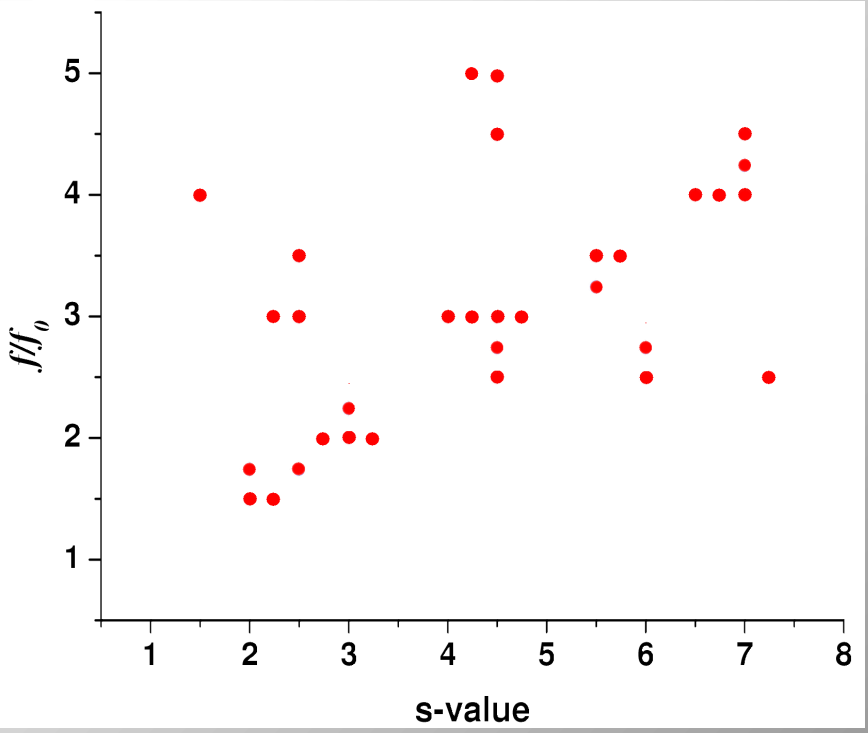




2-D Spectrum Analysis - Refinement:

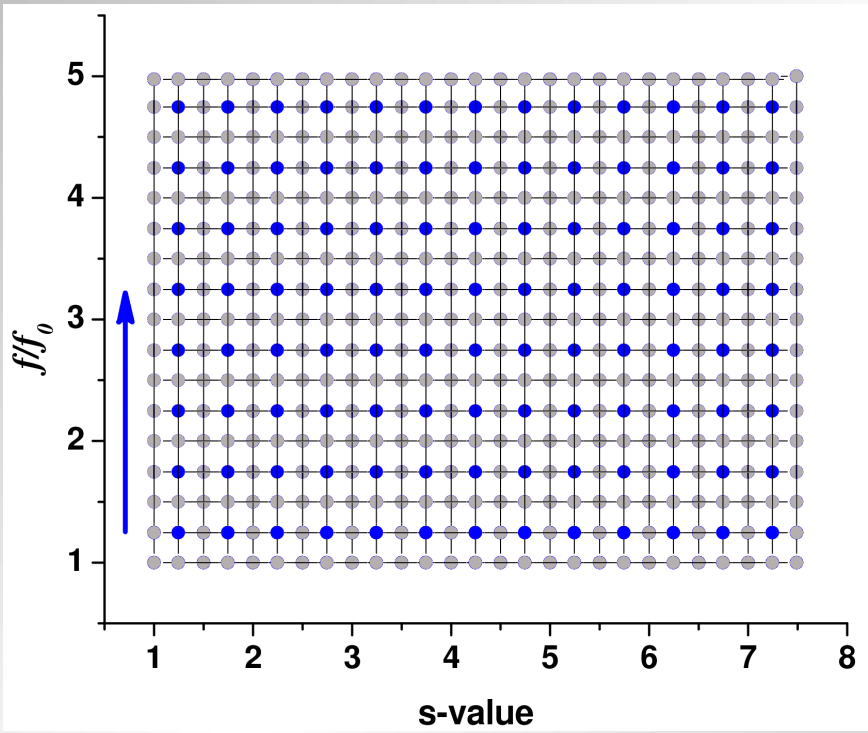
Step 9: Add the new non-zero elements to the stored array

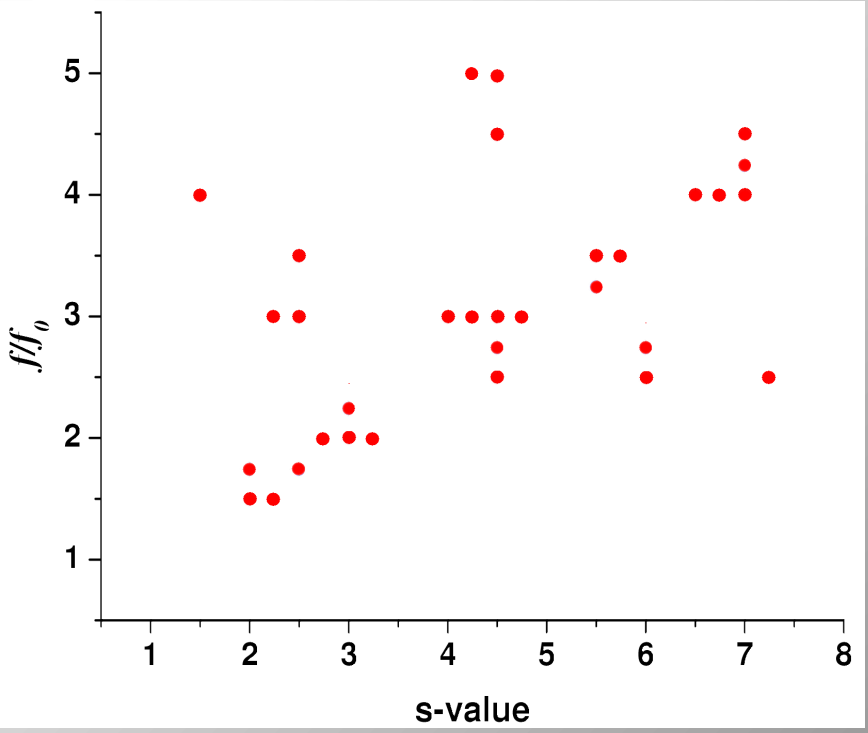




2-D Spectrum Analysis - Refinement:

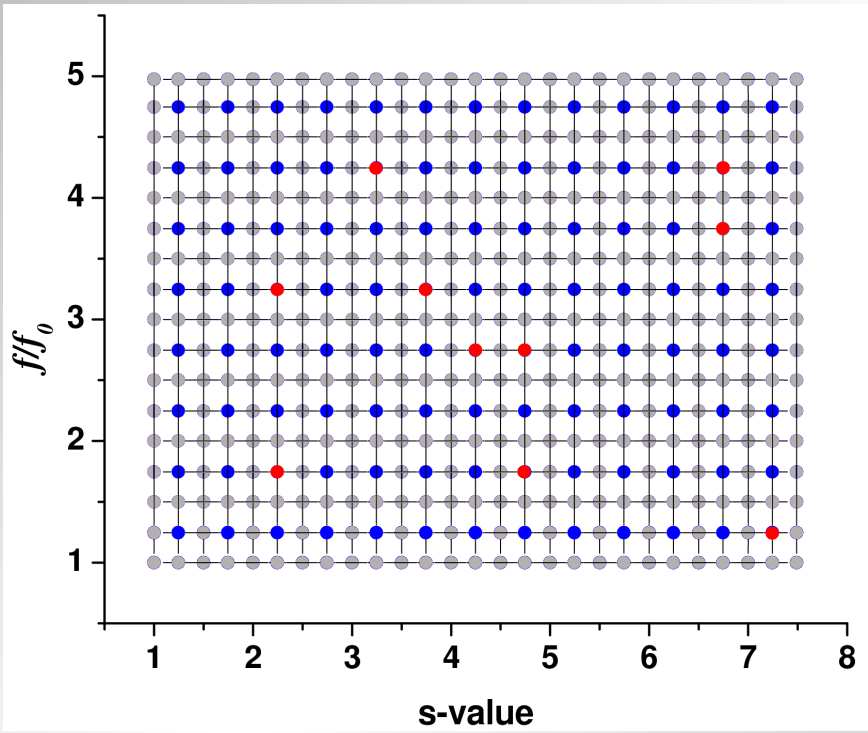
Step 10: Complete the square and shift the grid once more in the Y-direction

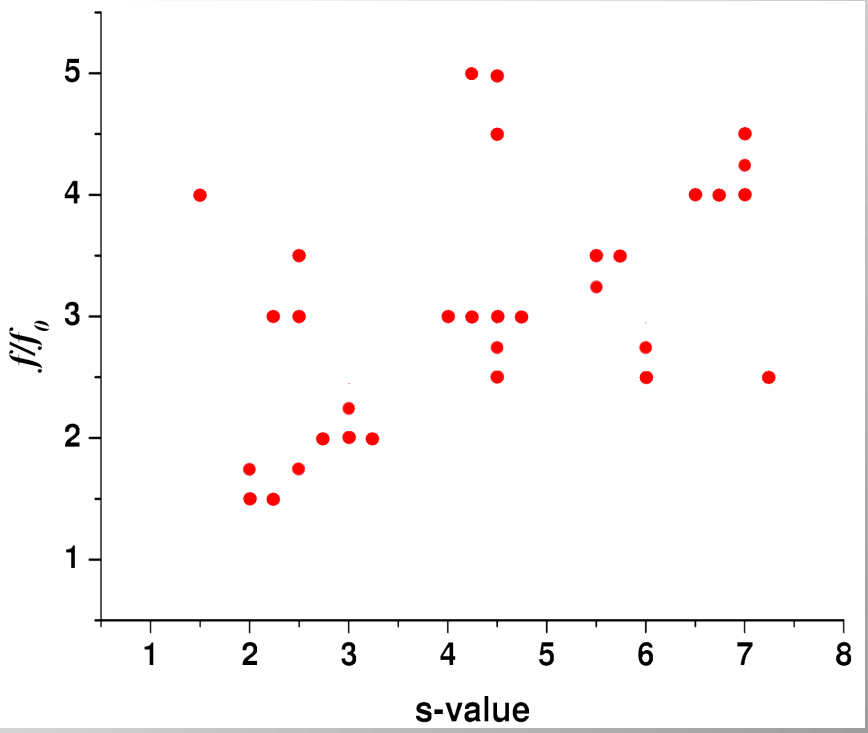




2-D Spectrum Analysis - Refinement:

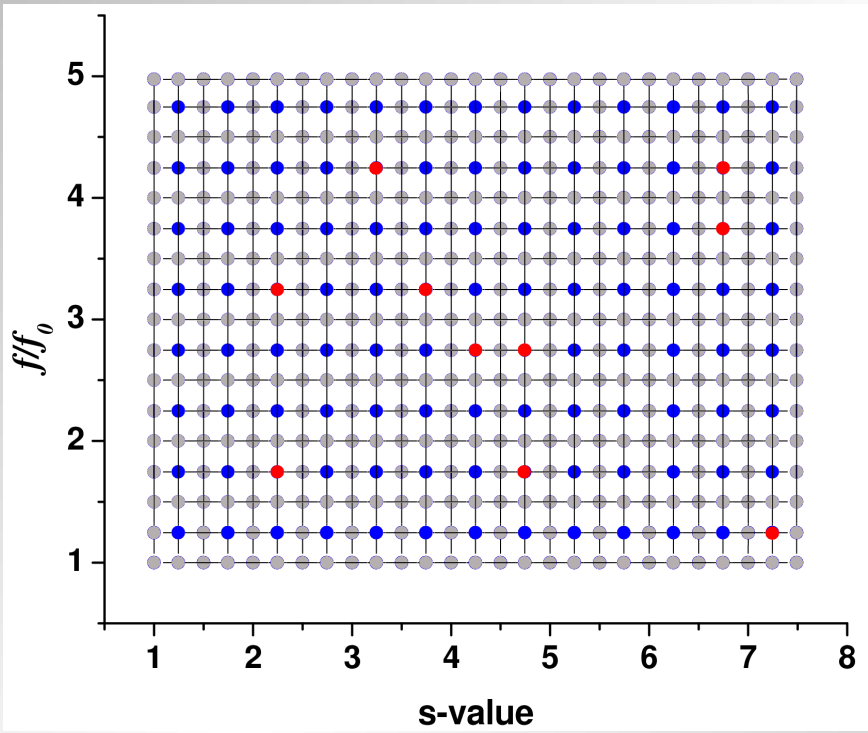
Step 11: Perform NNLS on the new grid

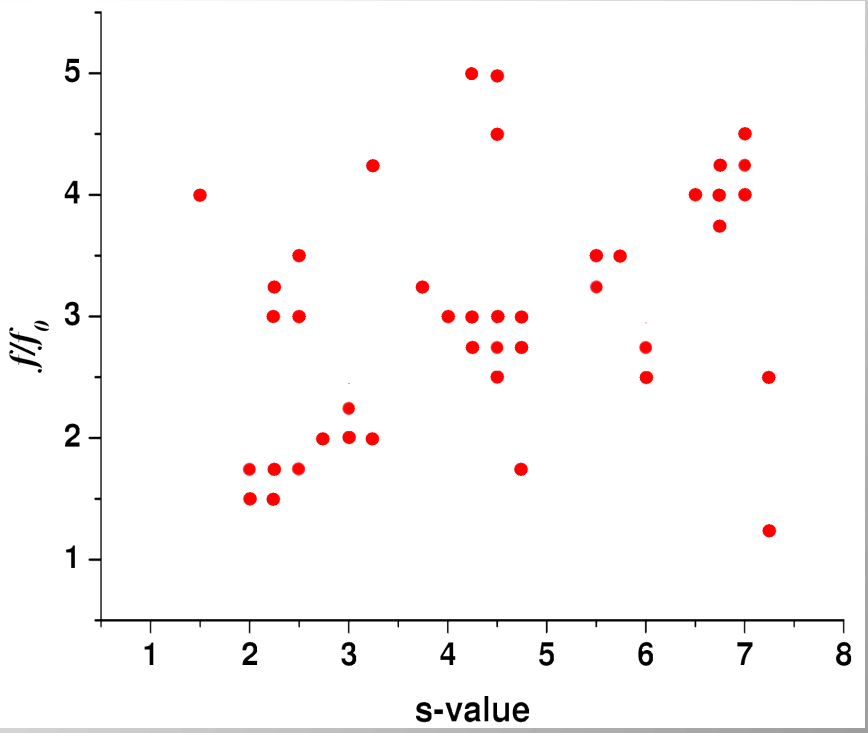




2-D Spectrum Analysis - Refinement:

Step 12: ... and add the non-zero points to the storage array



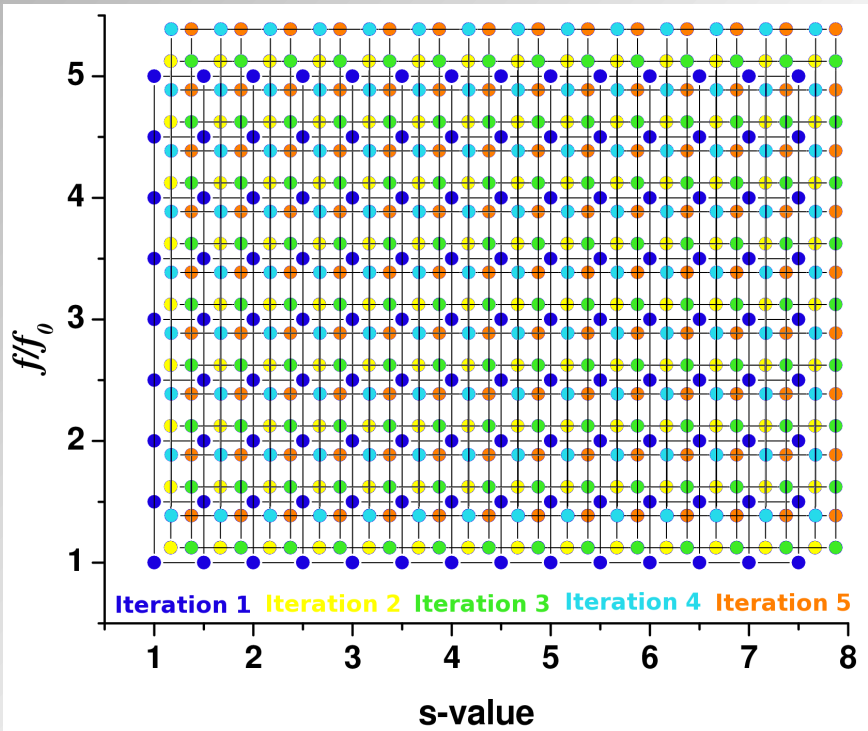


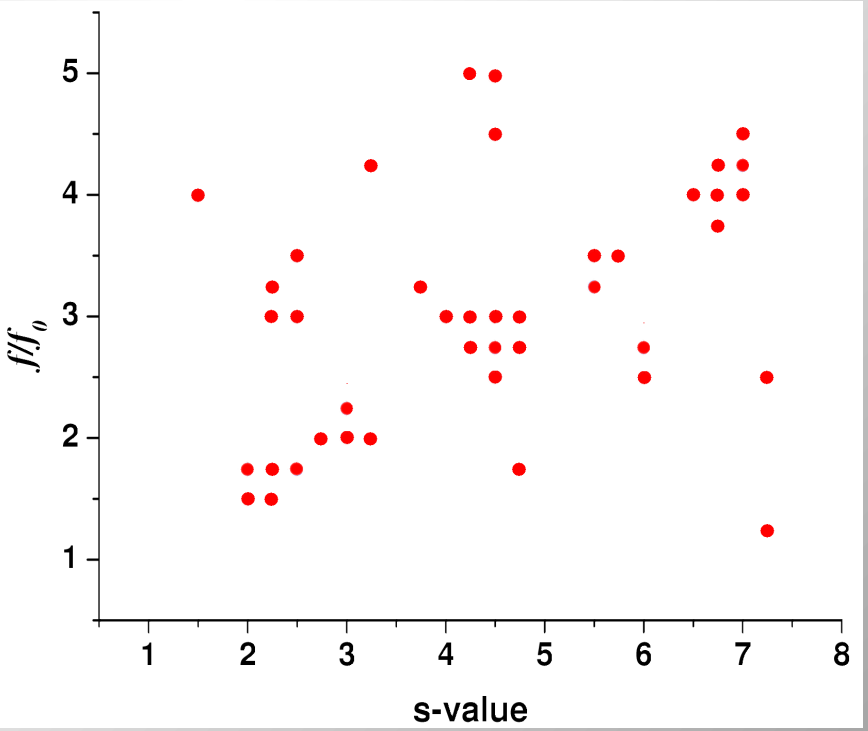
2-D Spectrum Analysis - Refinement:

Repeat this process until the desired grid size has been reached

2-D Spectrum Analysis - Refinement:

Divide and Conquer approach – evaluate multiple grids slightly off-set against each other, and accumulate results:

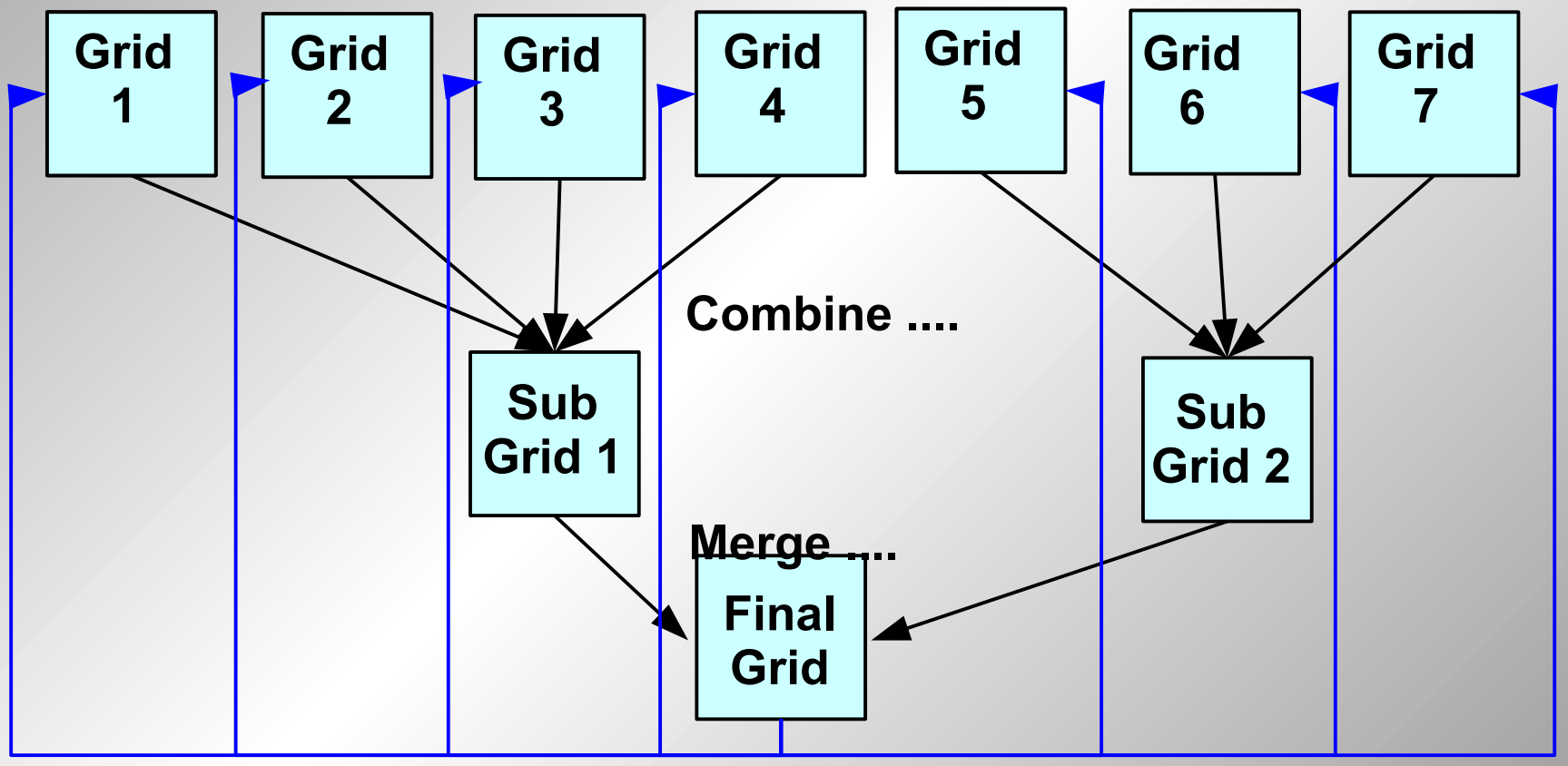




Final result is fairly sparse, but it is also degenerate, includes false positives and needs further refinement. It can be used to identify regions that contain signal.

Moving Grid Approach – parallel HPC implementation

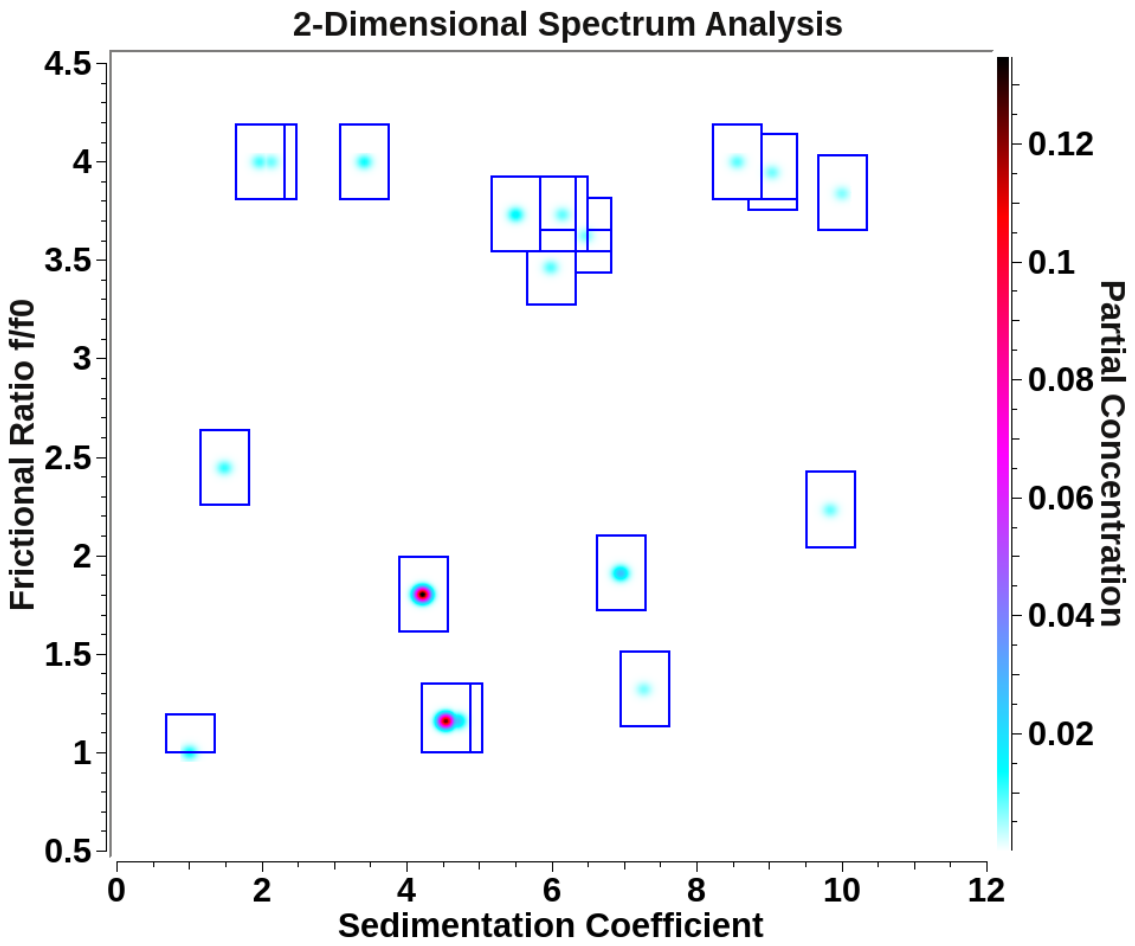
Calculate each individual grid in parallel ....



Evaluate each grid on a different processor, and communicate by MPI

Iterate until there is no more change ....

2DSA Result is used to initialize Genetic Algorithms



The 2DSA finds regions with signal in the parameter space.

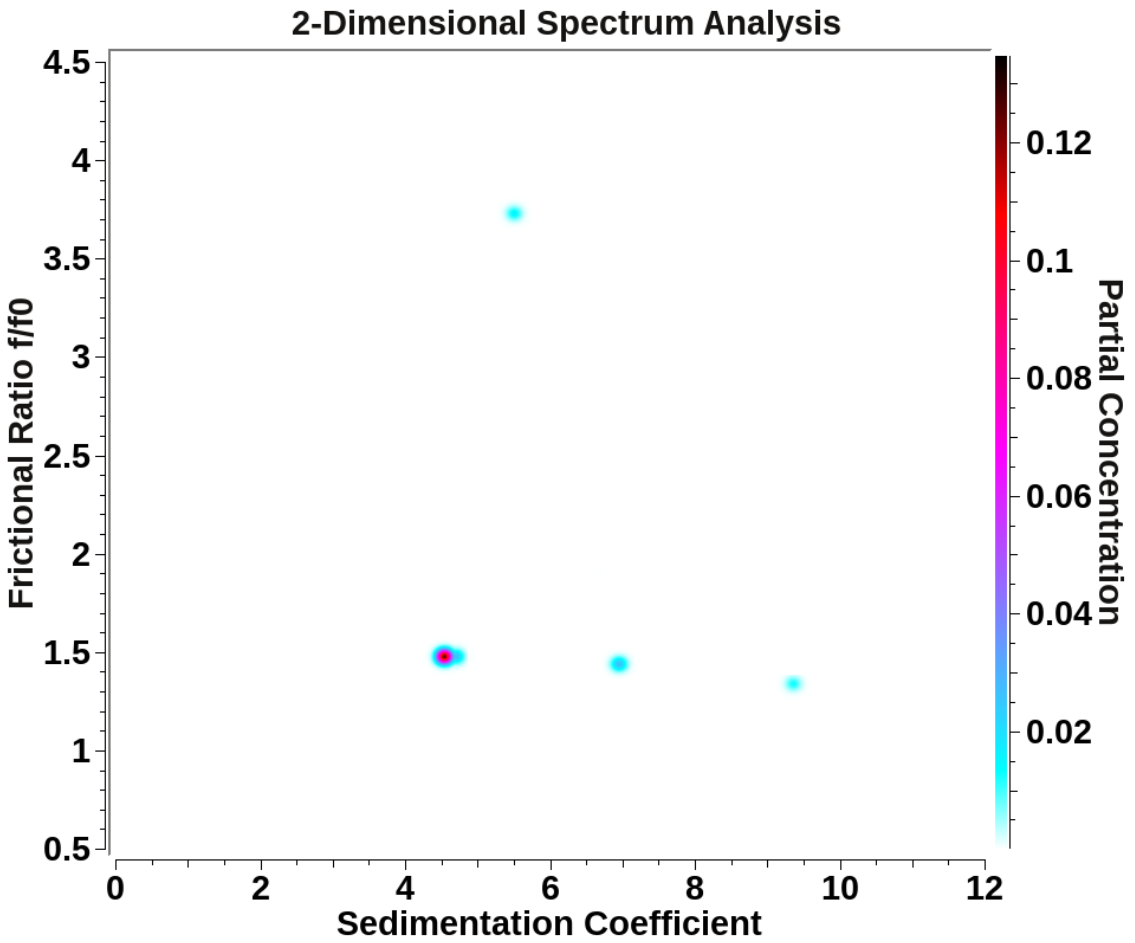
Genetic Algorithms are used to refine 2DSA solutions and to remove false positives through

"Parsimonious Regularization"

Initialize GA with 2DSA results

Perform calculations on a supercomputer

2DSA Result is used to initialize Genetic Algorithms



Genetic algorithms find the parsimonious solution that satisfies Occam's Razor

Genetic Algorithms can be combined with Monte Carlo analysis to explore effect of noise on parameter distributions.

Perform calculations on a supercomputer

Genetic Algorithms (GA)

Genetic Algorithms (also called evolutionary programming) provide a stochastic optimization method

John H Holland, Adaption in Natural and Artificial Systems, 1975, U. of Michigan Press

John R Koza, Genetic Programming: On the Programming of Computers by Means of Natural Selection, 1992, MIT Press

Based on nature – evolutionary paradigm

Mutation, recombination, deletion, insertion, crossover operators

Multiple populations ("demes") are allowed to compete, limited migration rates between demes are allowed.

Random number generators are used to manipulate operators

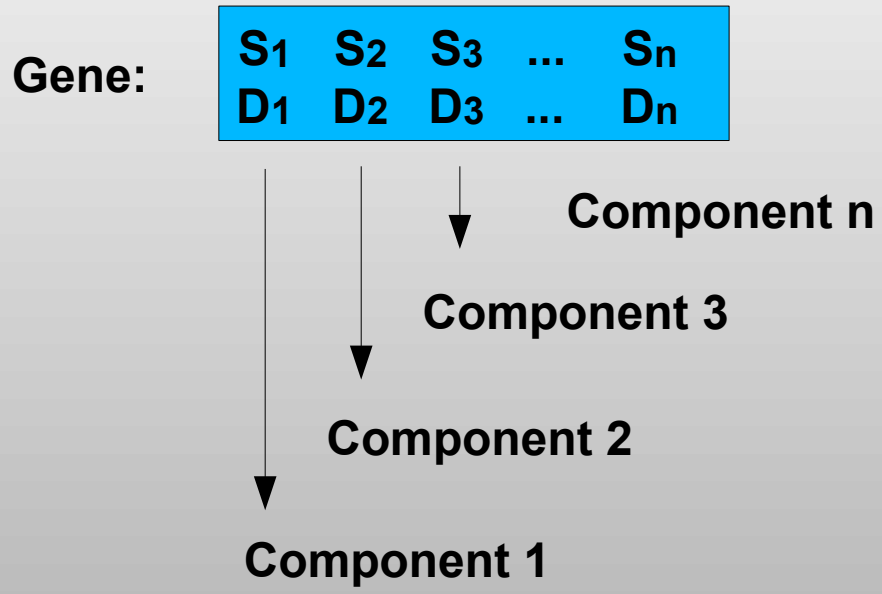
Generational Model – survival of the fittest (...fitting function)

Generation → iterations, genes → parameter strings, bases → s, D

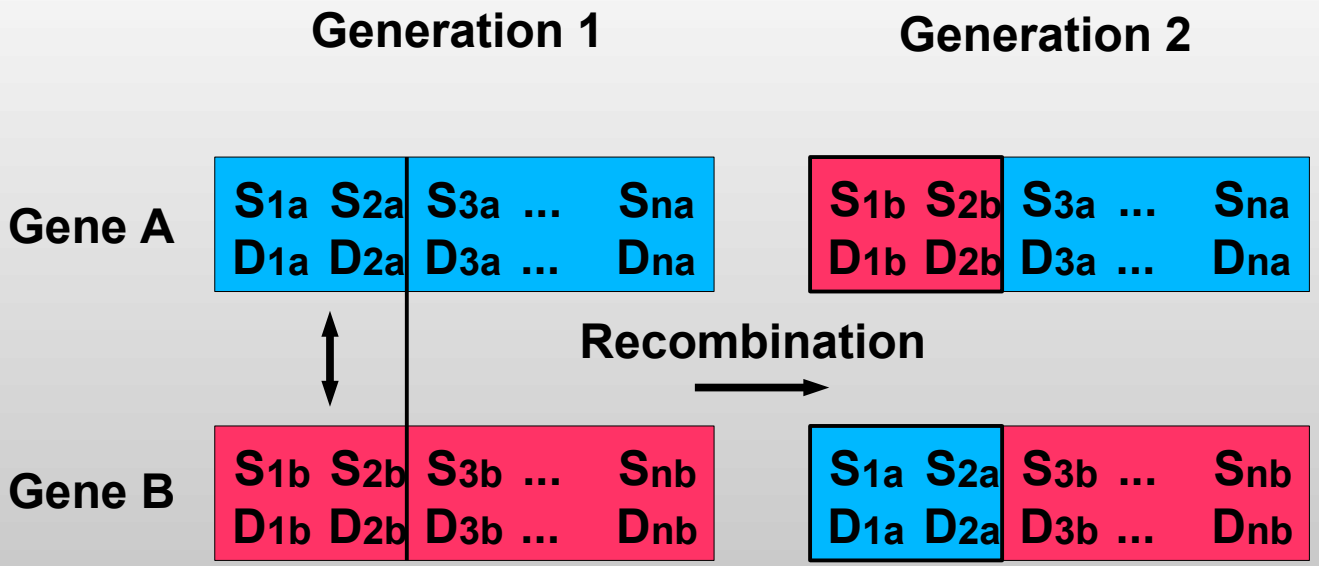
Each solute is simulated with the Lamm equation, solutes are summed

GA genes:

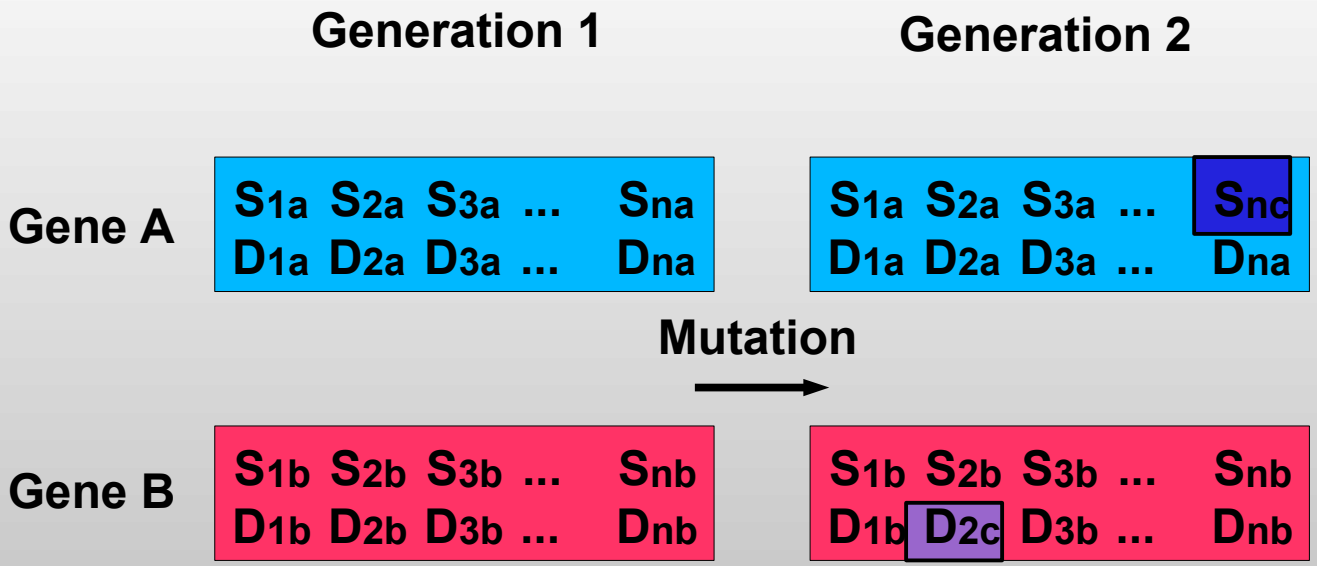
Genes are strings of parameters, each gene consists of a pair of corresponding sedimentation and diffusion coefficients.



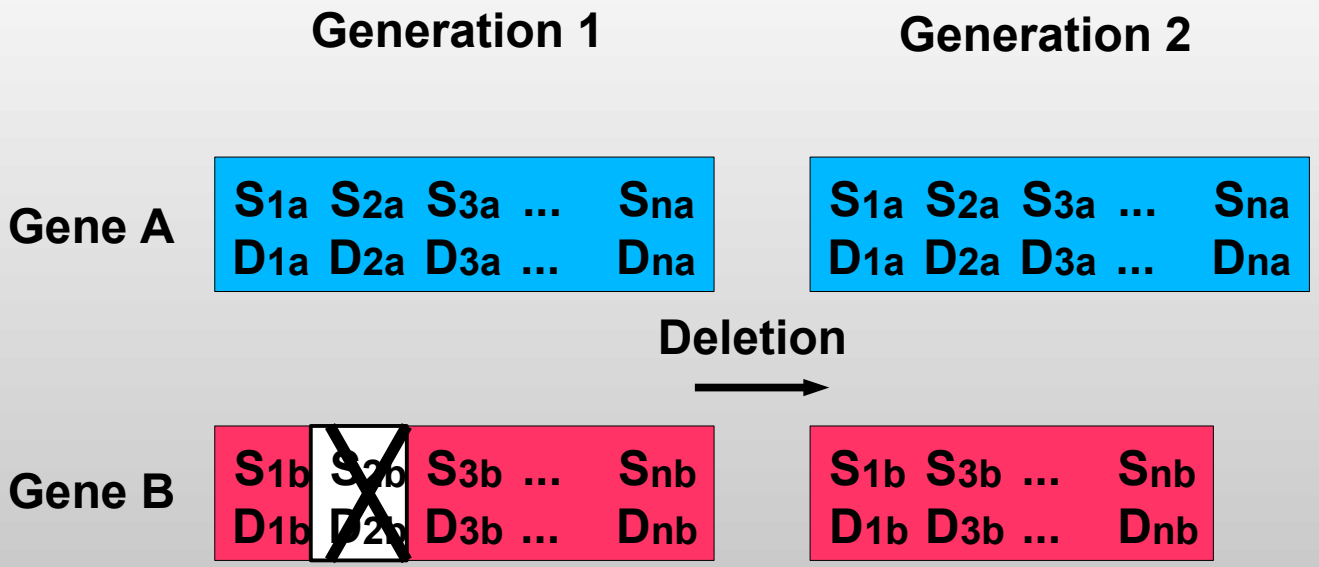
Crossover/Recombination



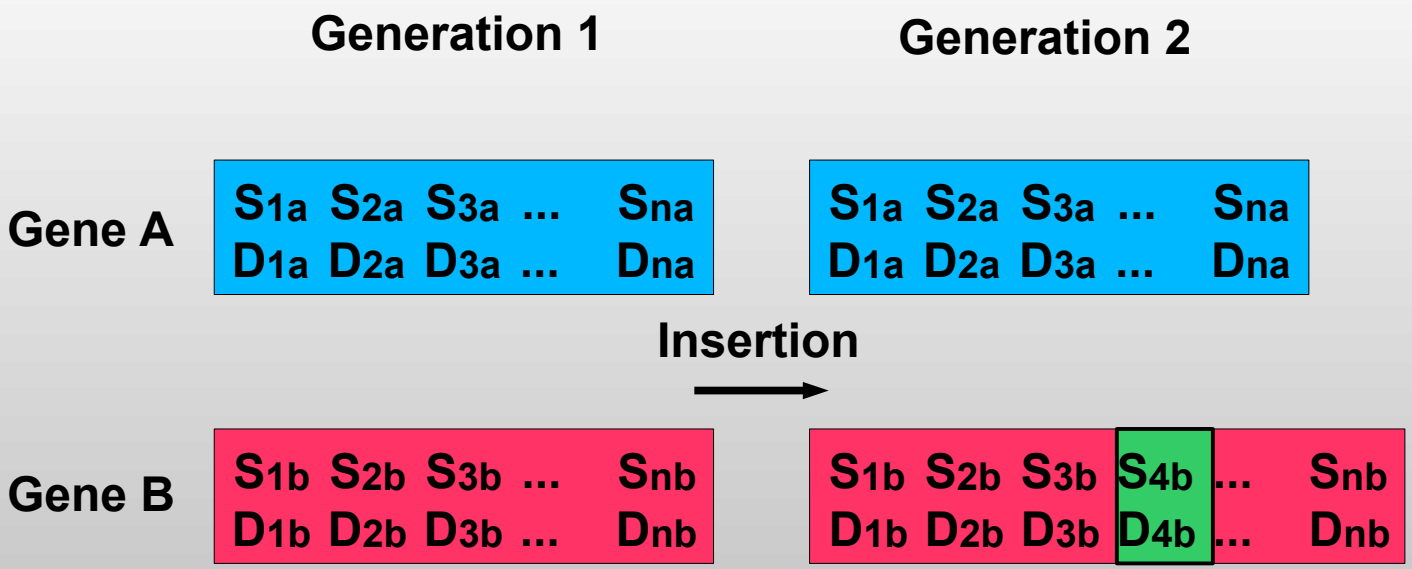
Mutation


Generation 1
Generation 2
Gene A
S1a S2a S3a ... Sna
D1a D2a D3a ... Dna
S1a S2a S3a ... Snc
D1a D2a D3a ... Dna
Mutation
Gene B
S1b S2b S3b ... Snb
D1b D2b D3b ... Dnb
S1b S2b S3b ... Snb
D1b D2c D3b ... Dnb

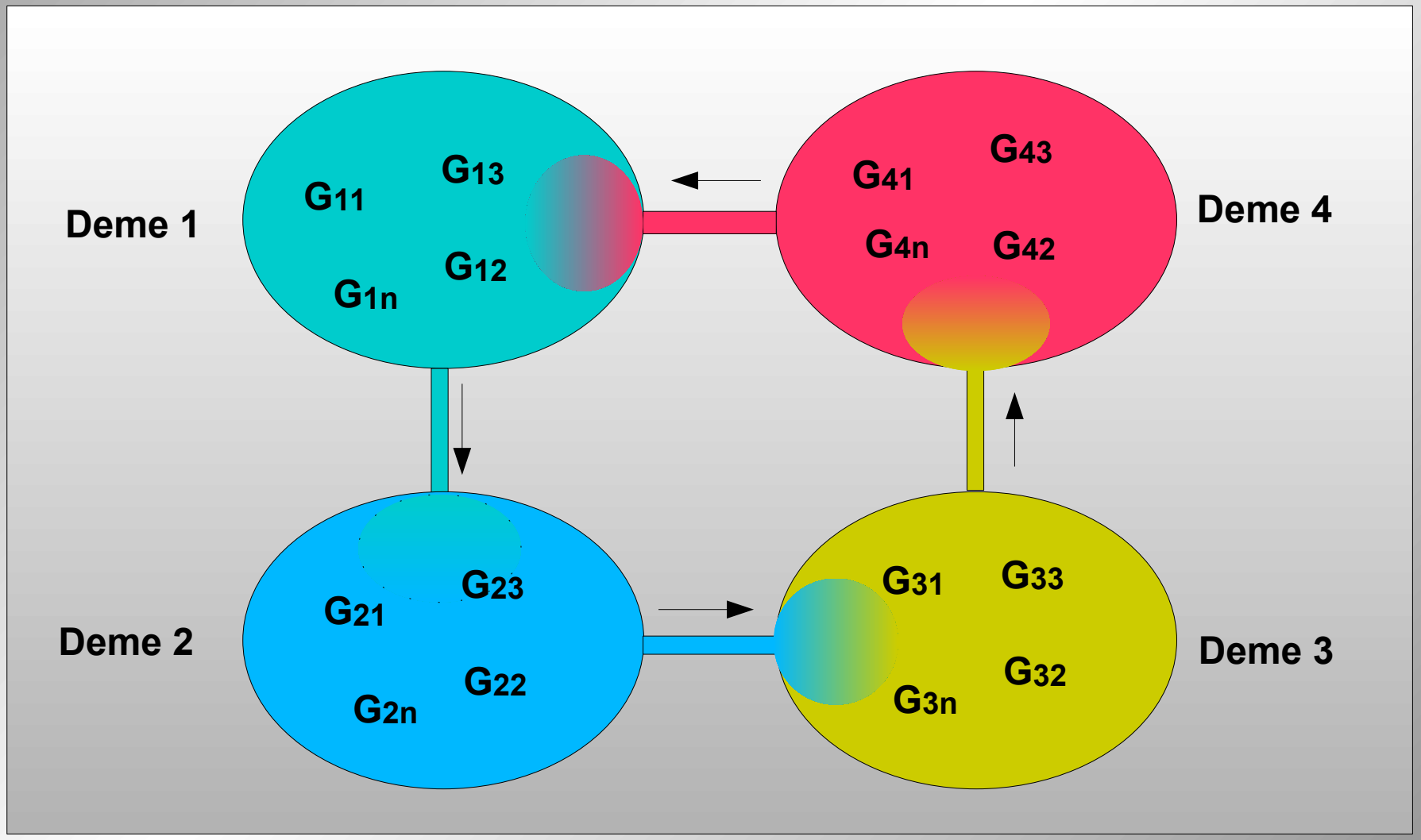
Deletion


Generation 1
Generation 2
Gene A
S1a S2a S3a ... Sna
D1a D2a D3a ... Dna
S1a S2a S3a ... Sna
D1a D2a D3a ... Dna
Deletion
Gene B
S1b S2b S3b ... Snb
D1b D2b D3b ... Dnb
S1b S3b ... Snb
D1b D3b ... Dnb

Insertion


Generation 1
Generation 2
Gene A
S1a S2a S3a ... Sna
D1a D2a D3a ... Dna
S1a S2a S3a ... Sna
D1a D2a D3a ... Dna
Insertion
Gene B
S1b S2b S3b ... Snb
D1b D2b D3b ... Dnb
S1b S2b S3b S4b ... Snb
D1b D2b D3b D4b ... Dnb

Deme Topology



Initialization of Genetic Algorithms

Parameters from all populations are initialized with reasonable starting guesses to create "genes".

s-values are initialized using the model independent van Holde – Weischet analysis*. It provides a good way to assess the limits and possible number of components.

Corresponding diffusion coefficients are randomly assigned based on a reasonable range for $k=f/f0$ values between given limits (i.e. 1-4):

$$D = \frac{RT}{18\pi N\left(k\eta\right)^{2/3}}\sqrt{\frac{2\left(1-\bar{\nu}\rho\right)}{s\bar{\nu}}}$$

**Demeler, B. and K. E. van Holde. Sedimentation velocity analysis of highly heterogeneous systems. (2004). Anal. Biochem. Vol 335(2):279-288*

Approach and Implementation - Initialization

Concentration values are determined with NNLS*, components with values below a threshold are eliminated.

Demes are initially kept isolated

Mutation/Crossover/Recombination operators are applied

Progeny is calculated and this process is iterated

After some iterations, migration rates are applied and nonlinear optimization (Quasi-Newton/Inverse Hessian) is applied for a few iterations.

** Lawson, C. L. and Hanson, R. J. 1974. Solving Least Squares Problems. Prentice-Hall, Inc. Englewood Cliffs, New Jersey*

Parametrically Constrained Spectrum Analysis

Biophysical Journal Volume 106 April 2014 1741–1750

A Parametrically Constrained Optimization Method for Fitting Sedimentation Velocity Experiments

Gary Gorbet,[†] Taylor Devlin,[†] Blanca I. Hernandez Uribe,[†] Aysha K. Demeler,[†] Zachary L. Lindsey,[‡] Suma Ganji,[†] Sabrah Breton,[†] Laura Weise-Cross,[§] Eileen M. Lafer,[†] Emre H. Brookes,[†] and Borries Demeler[†*]

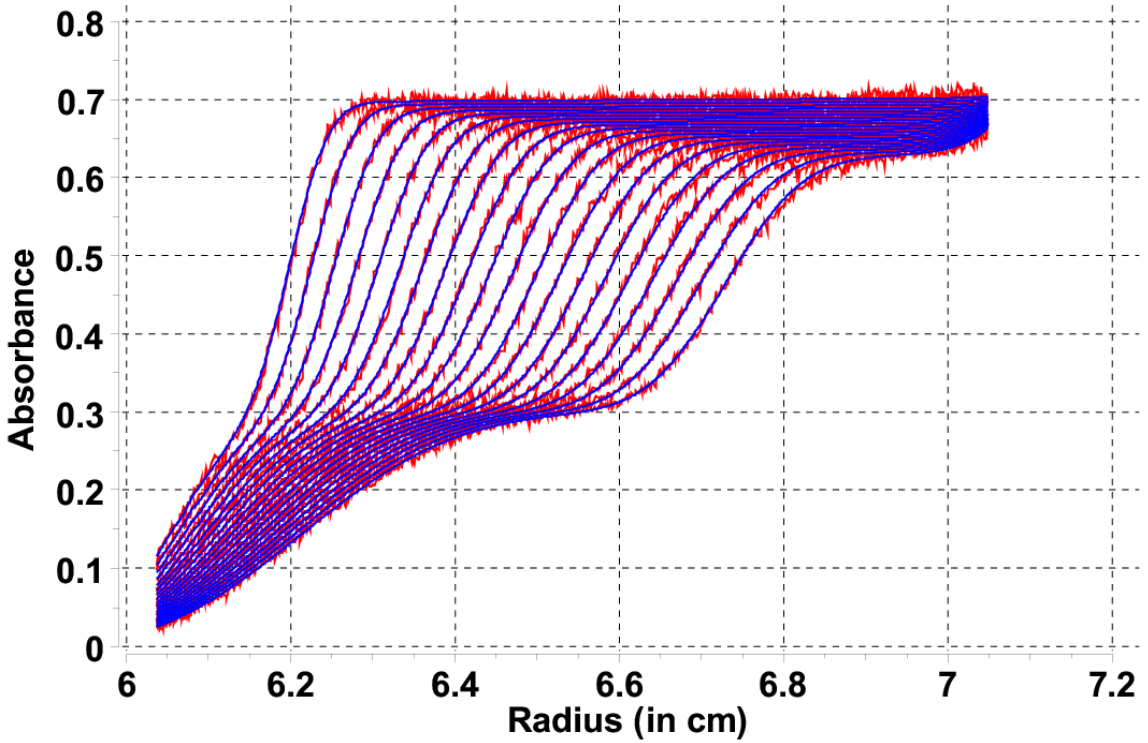
[†]The University of Texas Health Science Center at San Antonio, Department of Biochemistry, San Antonio, Texas; [‡]Texas A&M University, Department of Mechanical Engineering, College Station, Texas; and [§]University of North Carolina at Chapel Hill, Department of Pathology and Laboratory Medicine, Chapel Hill, North Carolina

Motivation:

We want a method that can model polymerizing systems that follow a systematic size/shape growth function (for example, end-to-end polymerization) where the anisotropy for each size changes in a predictable fashion

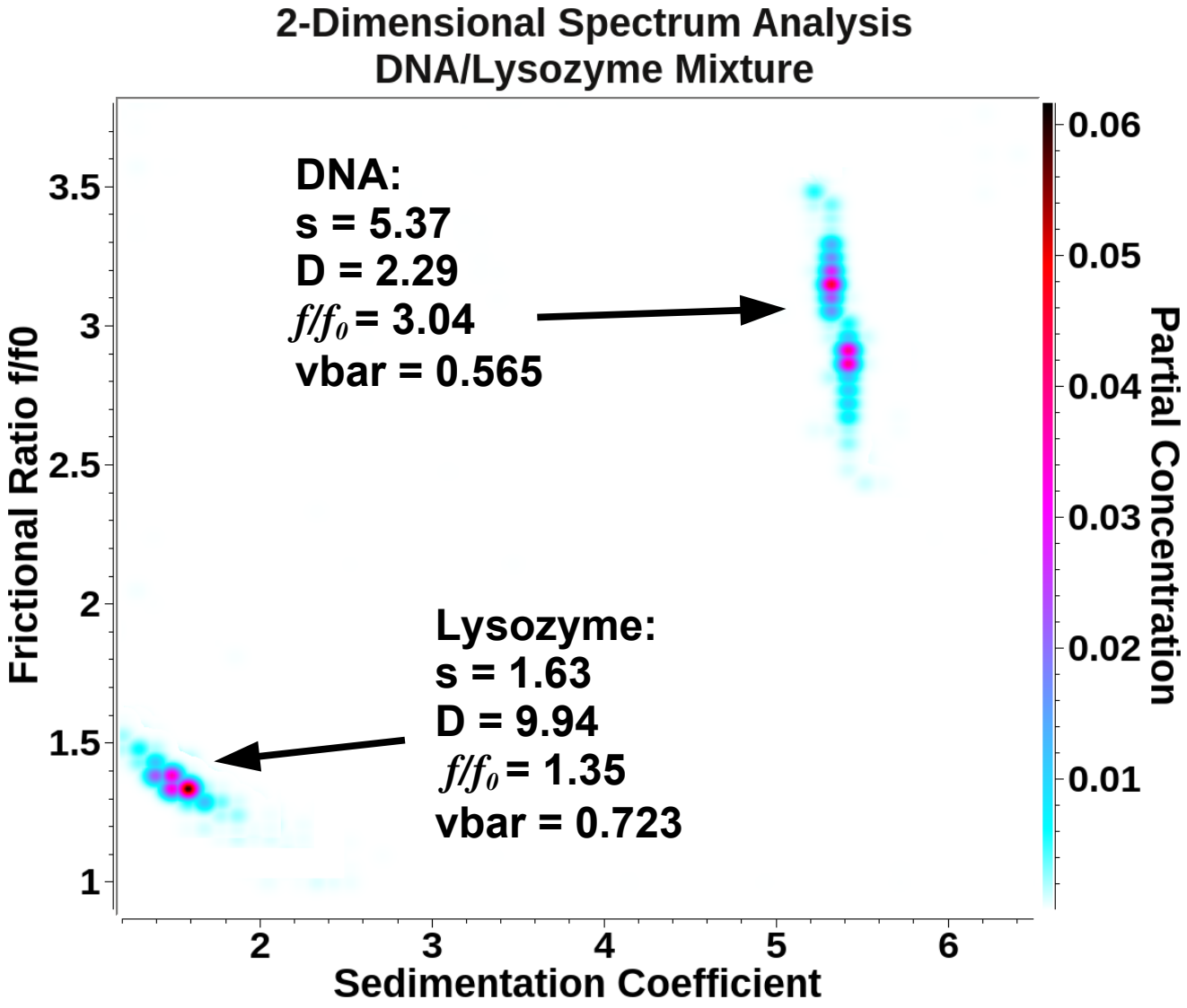
Parametrically Constrained Spectrum Analysis

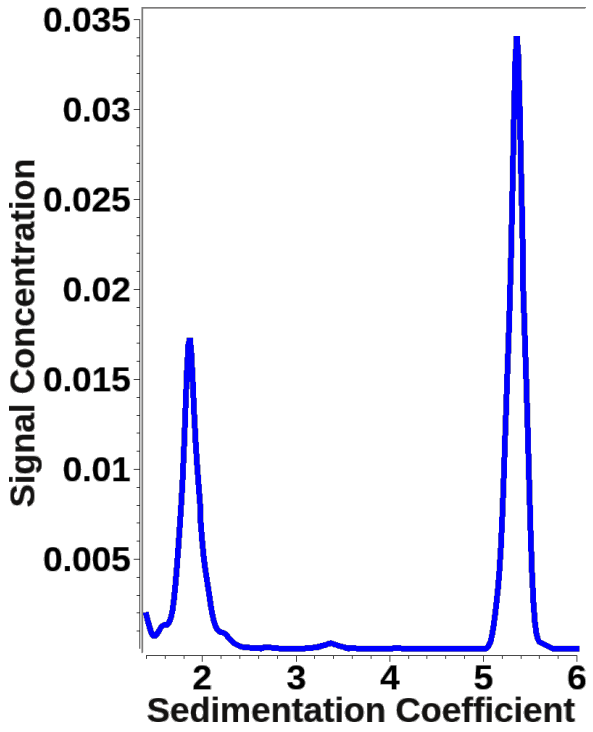
Simple two component system where both components have different anisotropy, fitted with a nonlinear method:



Lysozyme: 14.3 kDa (globular protein)

208 bp DNA: 131.0 kDa (extended linear dsDNA)



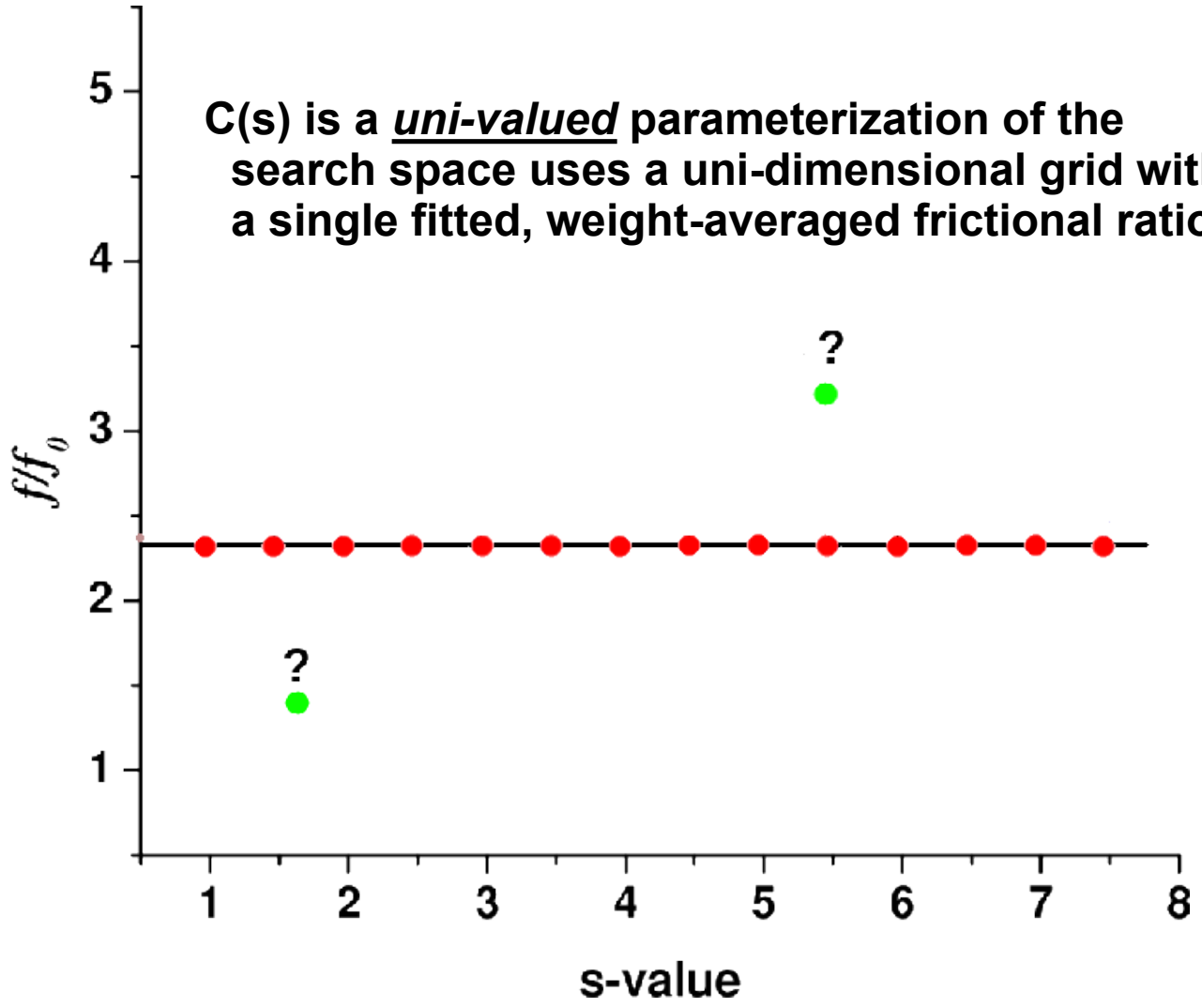


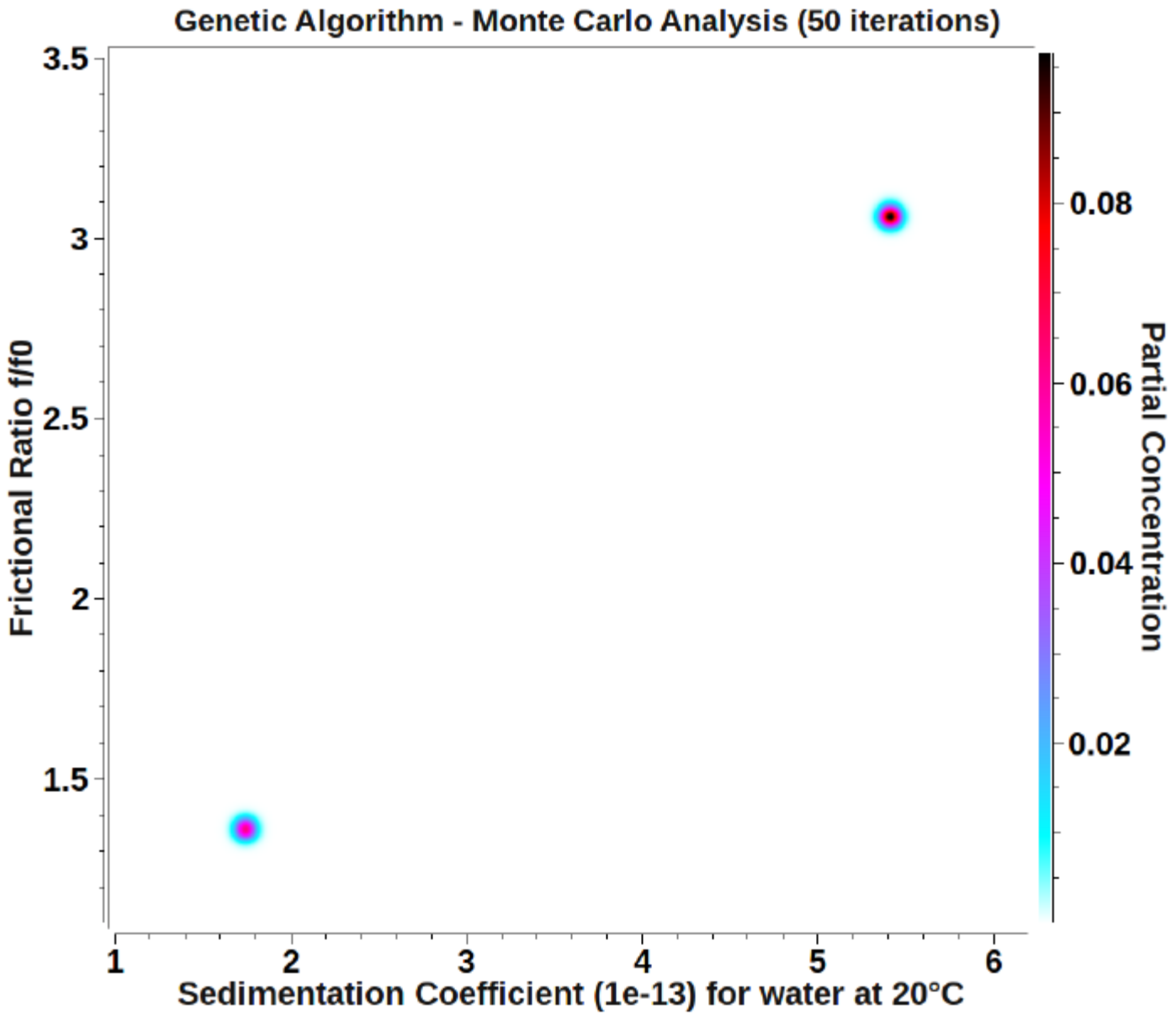
Goal:

Identify a *uni-valued* parameterization for the 2-dimensional size and shape domain that models polymer growth as function of its intrinsic shape changes. Constrain molecular weight to a single anisotropy.

Parametrically Constrained Spectrum Analysis

C(s) is a ***uni-valued*** parameterization of the search space uses a uni-dimensional grid with a single fitted, weight-averaged frictional ratio:





Genetic algorithms give the right answer, but computationally expensive

Parametrically Constrained Spectrum Analysis

Biophysical Journal Volume 106 April 2014 1741–1750

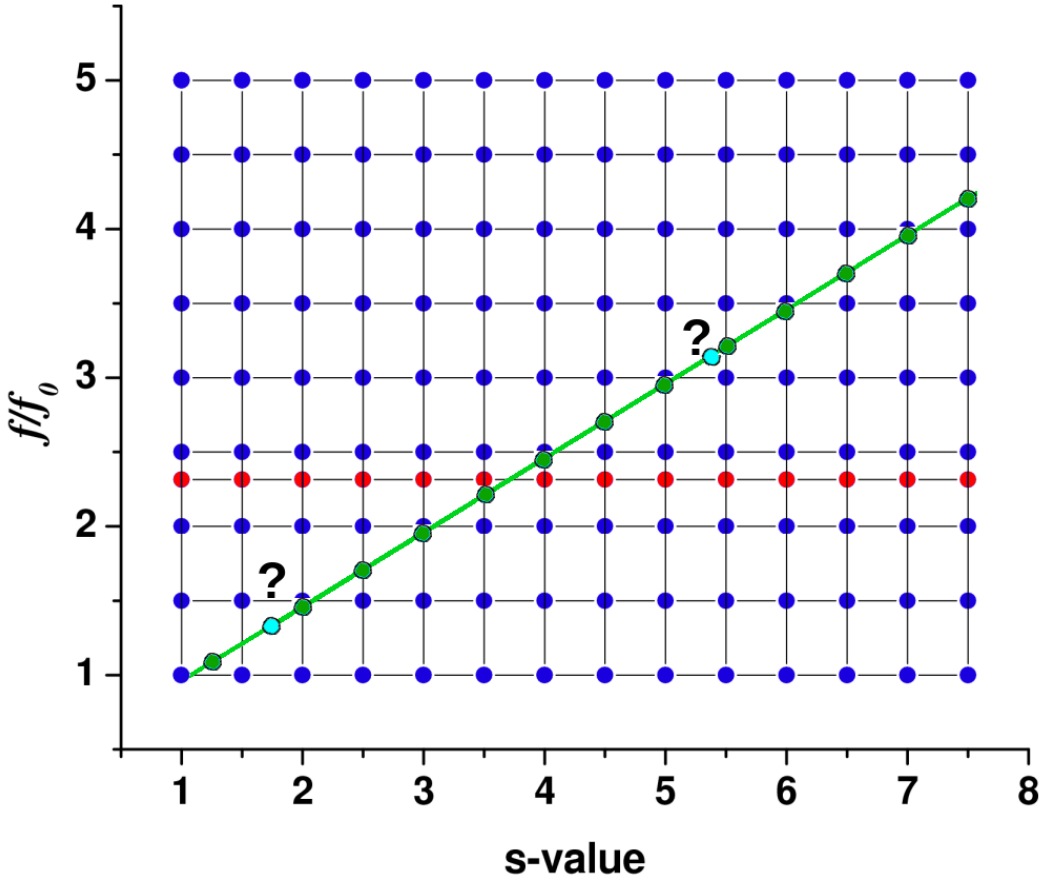
A Parametrically Constrained Optimization Method for Fitting Sedimentation Velocity Experiments

Gary Gorbet,[†] Taylor Devlin,[†] Blanca I. Hernandez Uribe,[†] Aysha K. Demeler,[†] Zachary L. Lindsey,[‡] Suma Ganji,[†] Sabrah Breton,[†] Laura Weise-Cross,[§] Eileen M. Lafer,[†] Emre H. Brookes,[†] and Borries Demeler[†*]

[†]The University of Texas Health Science Center at San Antonio, Department of Biochemistry, San Antonio, Texas; [‡]Texas A&M University, Department of Mechanical Engineering, College Station, Texas; and [§]University of North Carolina at Chapel Hill, Department of Pathology and Laboratory Medicine, Chapel Hill, North Carolina

Motivation:

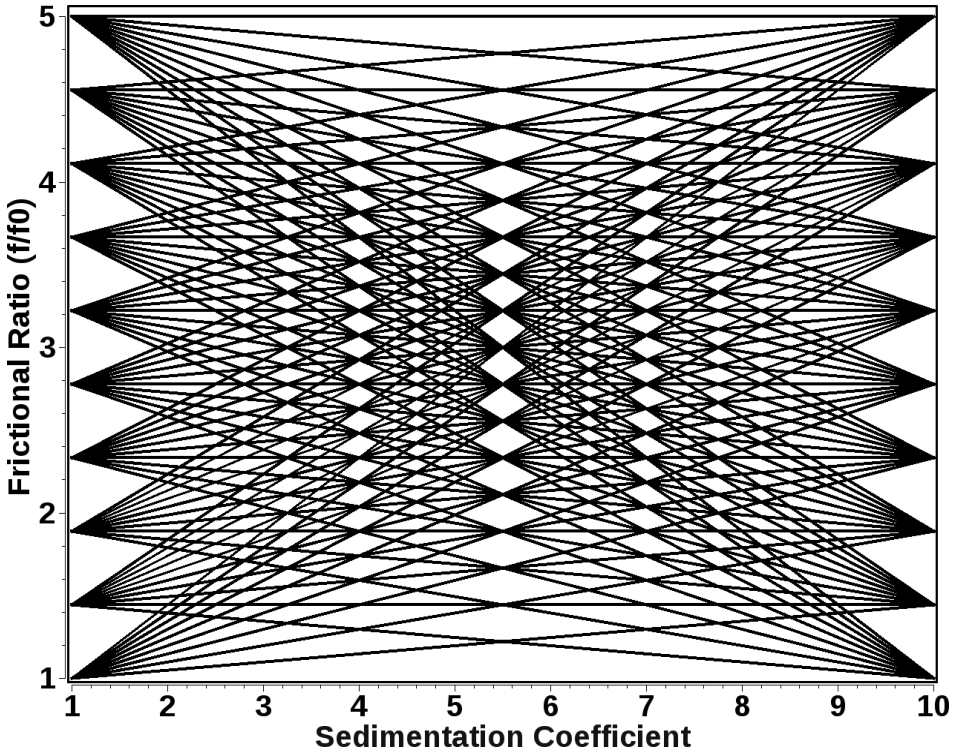
We want a *general* method that can model polymerizing systems that follow a systematic size-anisotropy growth function (e.g., end-to-end polymerization) where the anisotropy for each size changes in a predictable fashion, using a *uni-valued* relationship that maps one size to one anisotropy value.

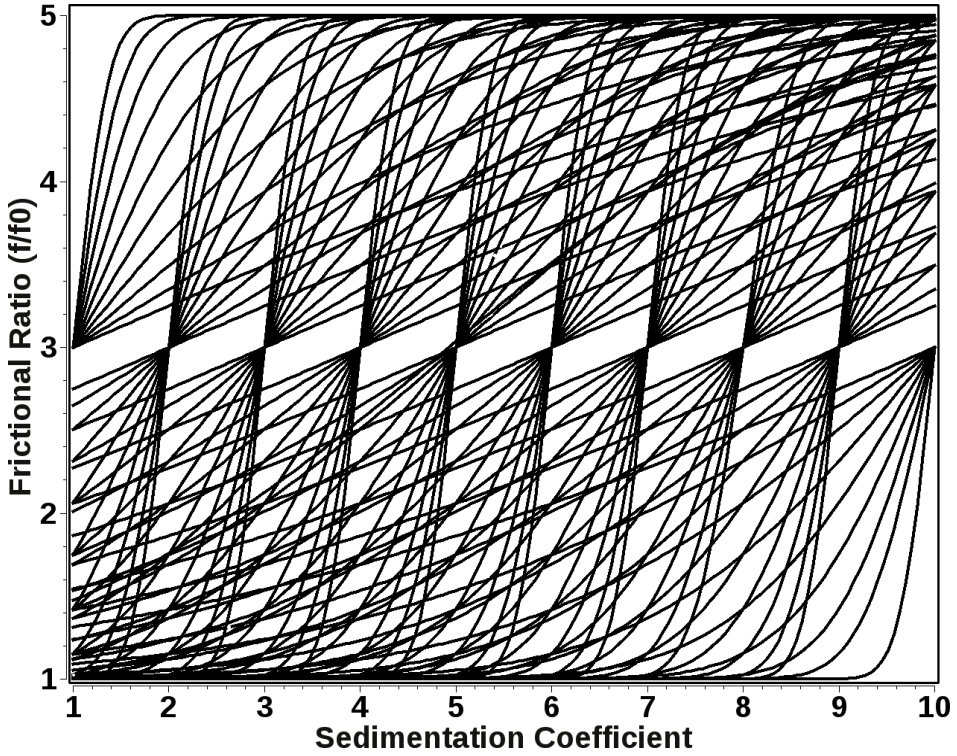


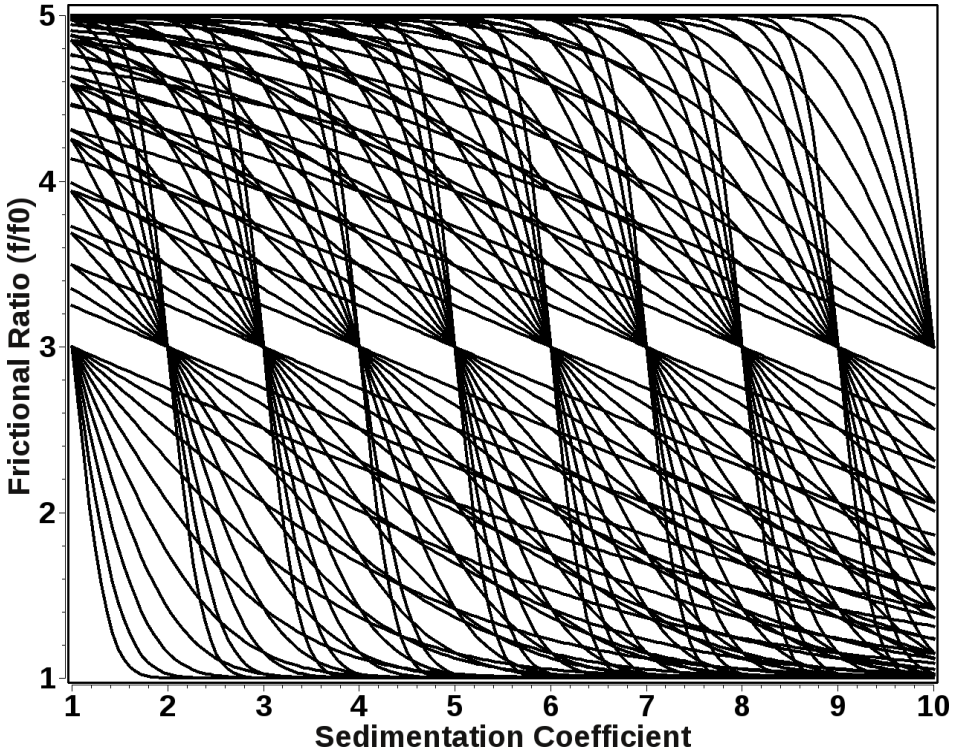
Parametrically Constrained Spectrum Analysis

PCSA Approach:

- **Select any single-valued function (straight line, hyperbolic functions, increasing/decreasing sigmoid, exponential growth/decay, etc.)**
- **Generate a discrete grid of functions by varying the function's parameters to achieve a good coverage between the user-selected limits for the 2-dimensional range $\langle f/f_{0,min}, f/f_{0,max} \rangle$, $\langle s_{min}, s_{max} \rangle$.**
- **Discretize each function over the 2-dimensional parameter space and solve with finite element and NNLS.**

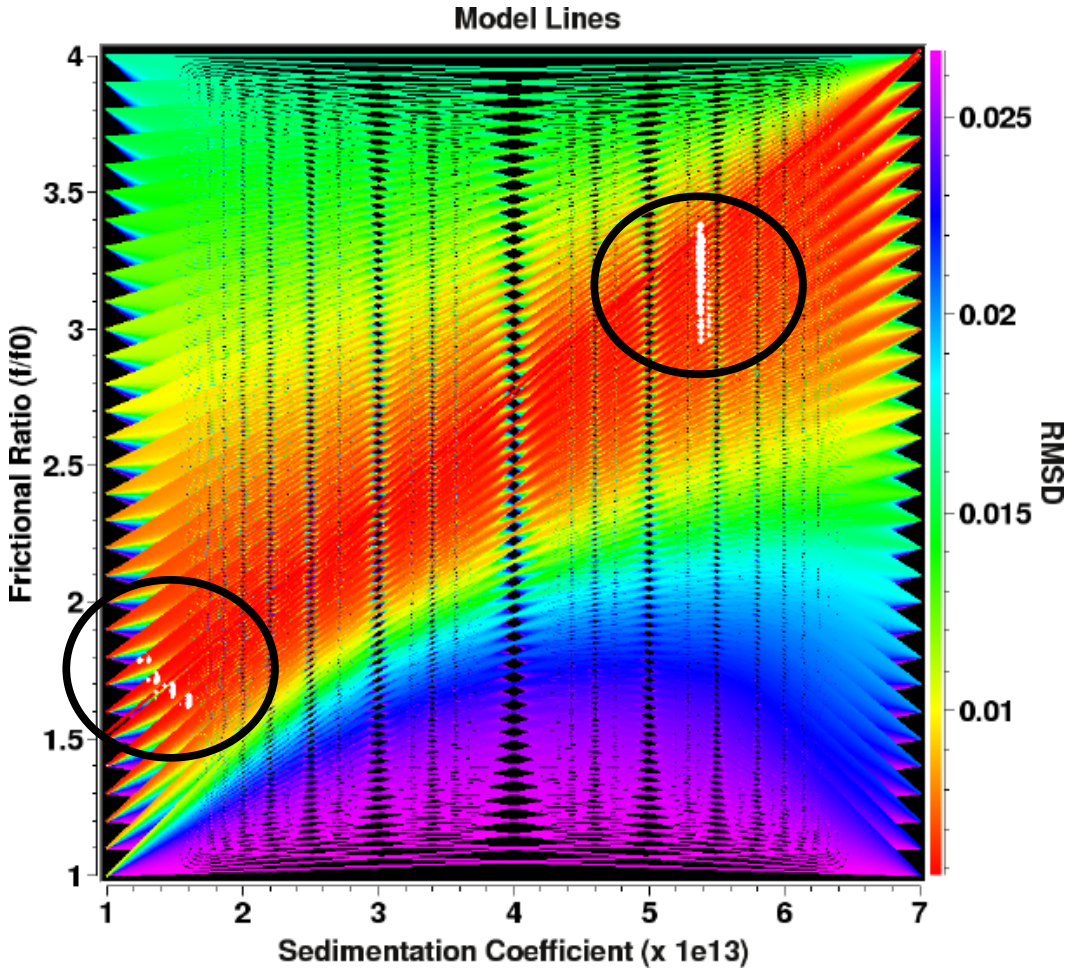

Frictional Ratio (f/f0)
Sedimentation Coefficient
1
2
3
4
5
6
7
8
9
10


Frictional Ratio (f/f0)
Sedimentation Coefficient
1
2
3
4
5
6
7
8
9
10

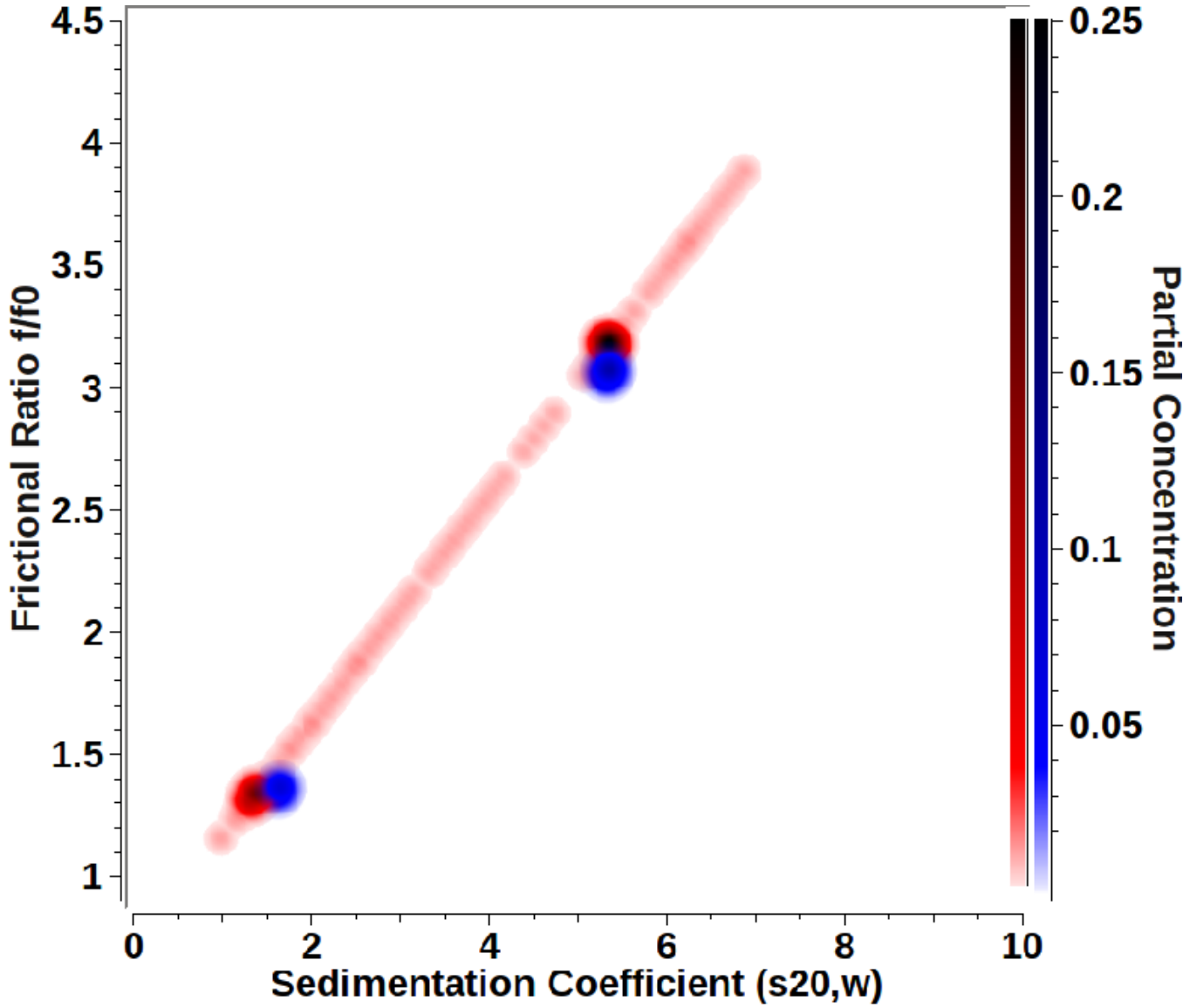

Frictional Ratio (f/f0)
Sedimentation Coefficient
1
2
3
4
5
6
7
8
9
10

Parametrically Constrained Spectrum Analysis

Select the NNLS fit with the lowest RMSD and perform a Levenberg-Marquardt fit of the function's parameters to find the best model.

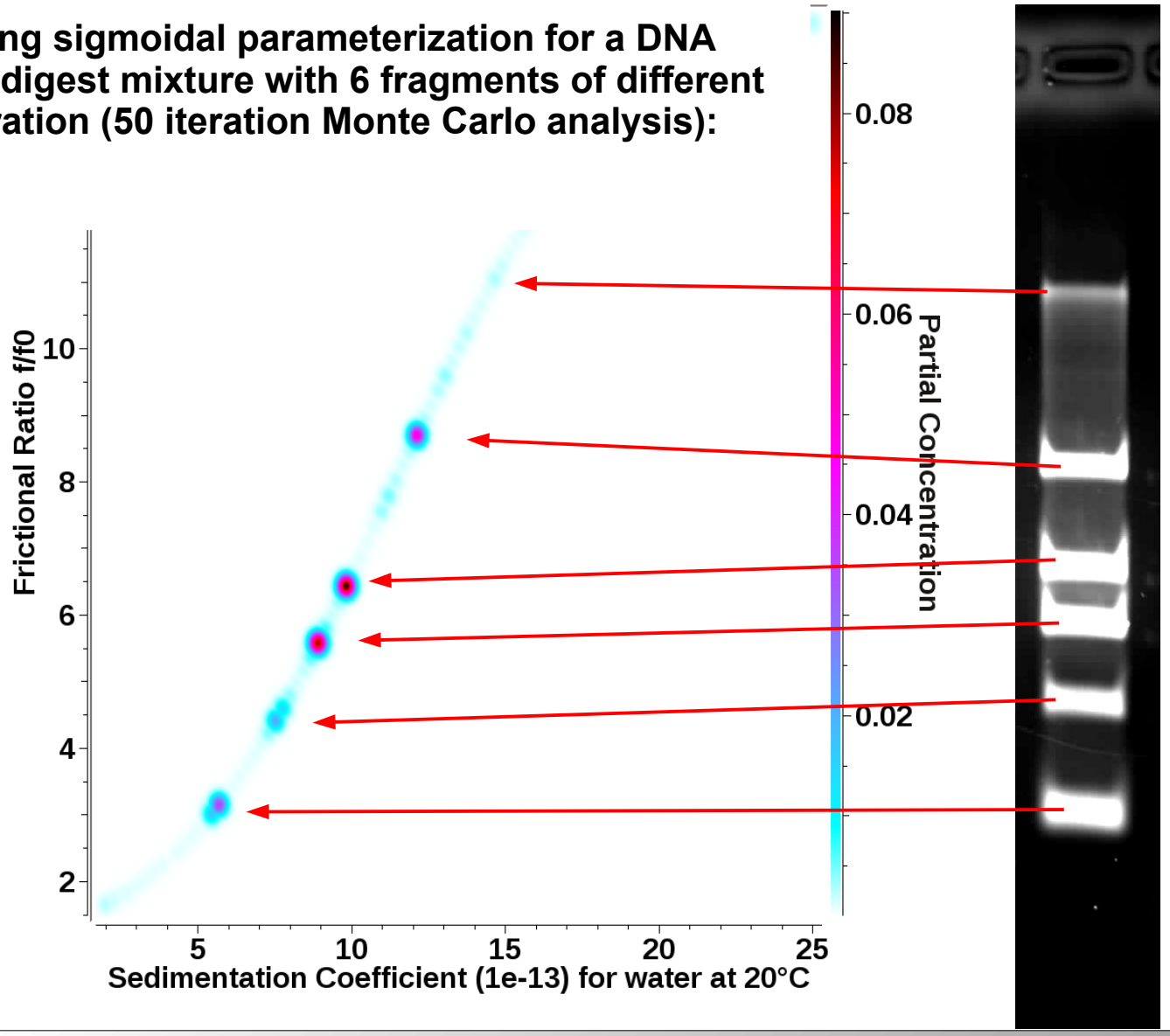


Overlay plots for PCSA (red) with Genetic Algorithm - Monte Carlo (blue)

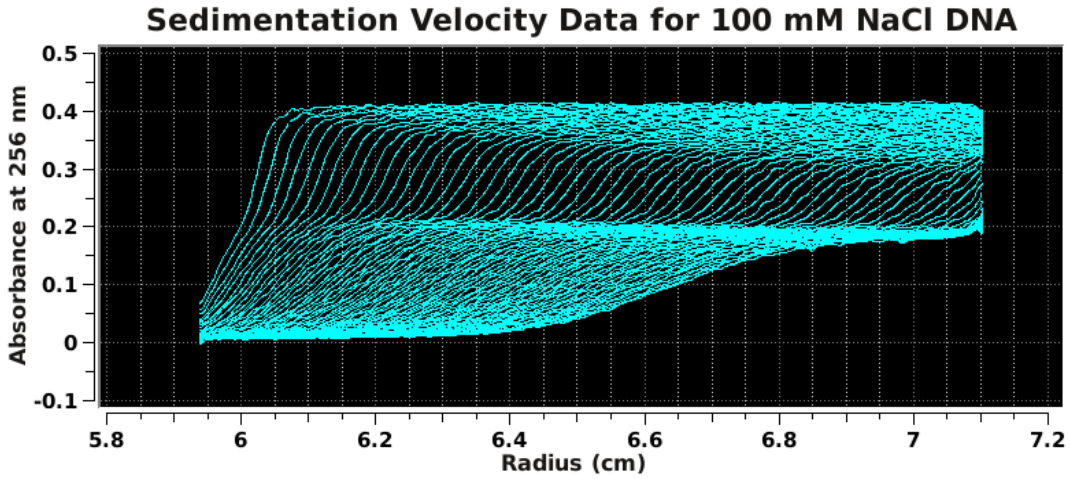


Parametrically Constrained Spectrum Analysis

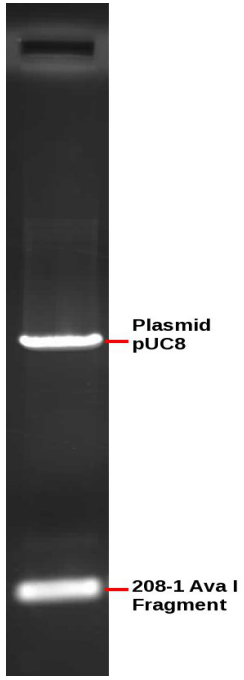
Increasing sigmoidal parameterization for a DNA restriction digest mixture with 6 fragments of different concentration (50 iteration Monte Carlo analysis):

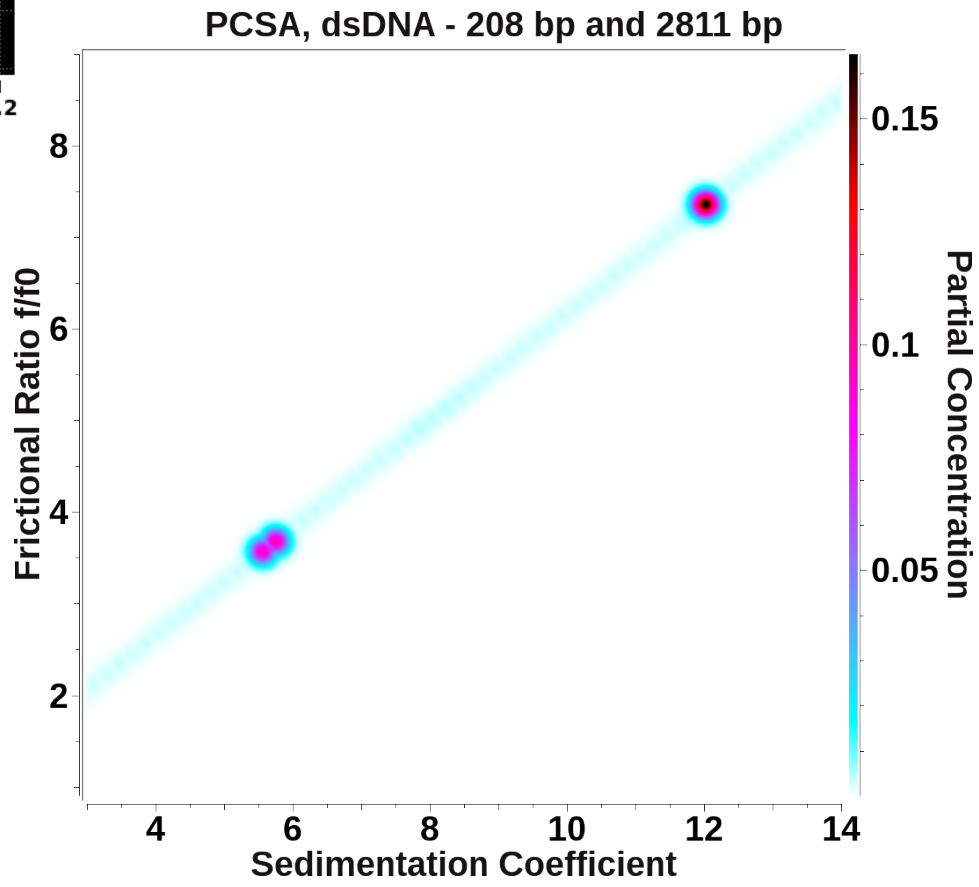


Parametrically Constrained Spectrum Analysis

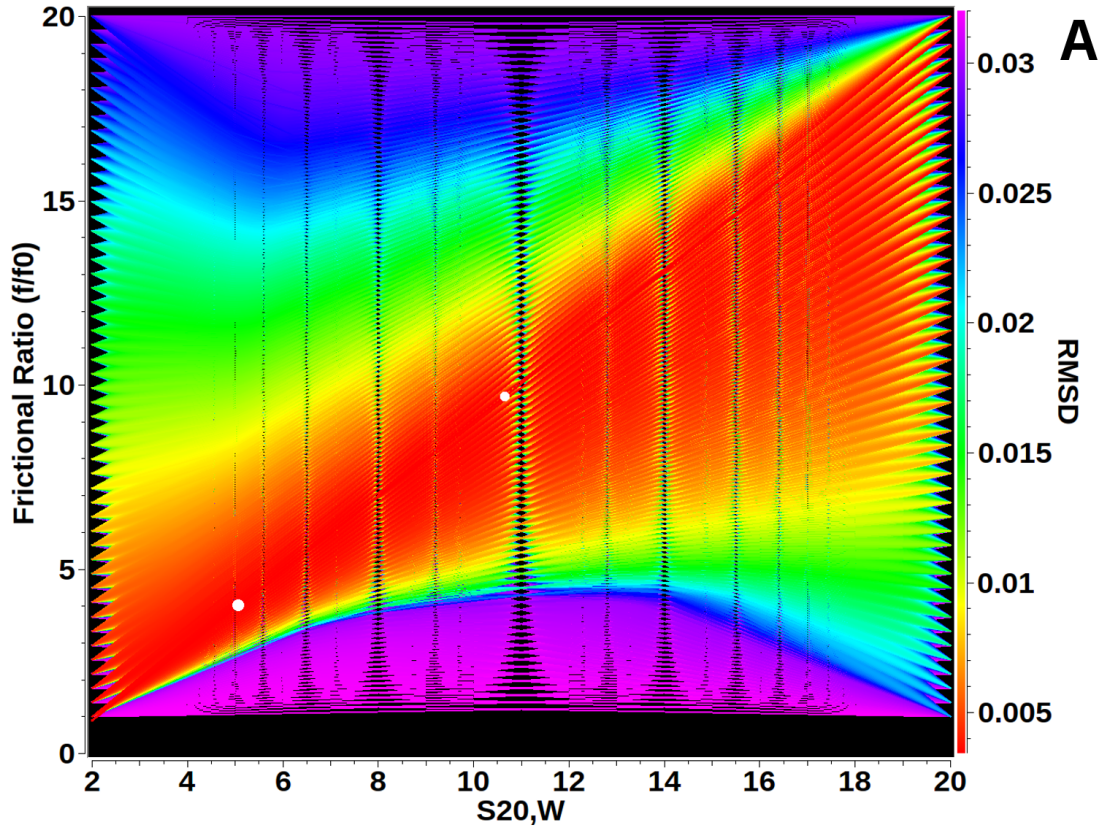


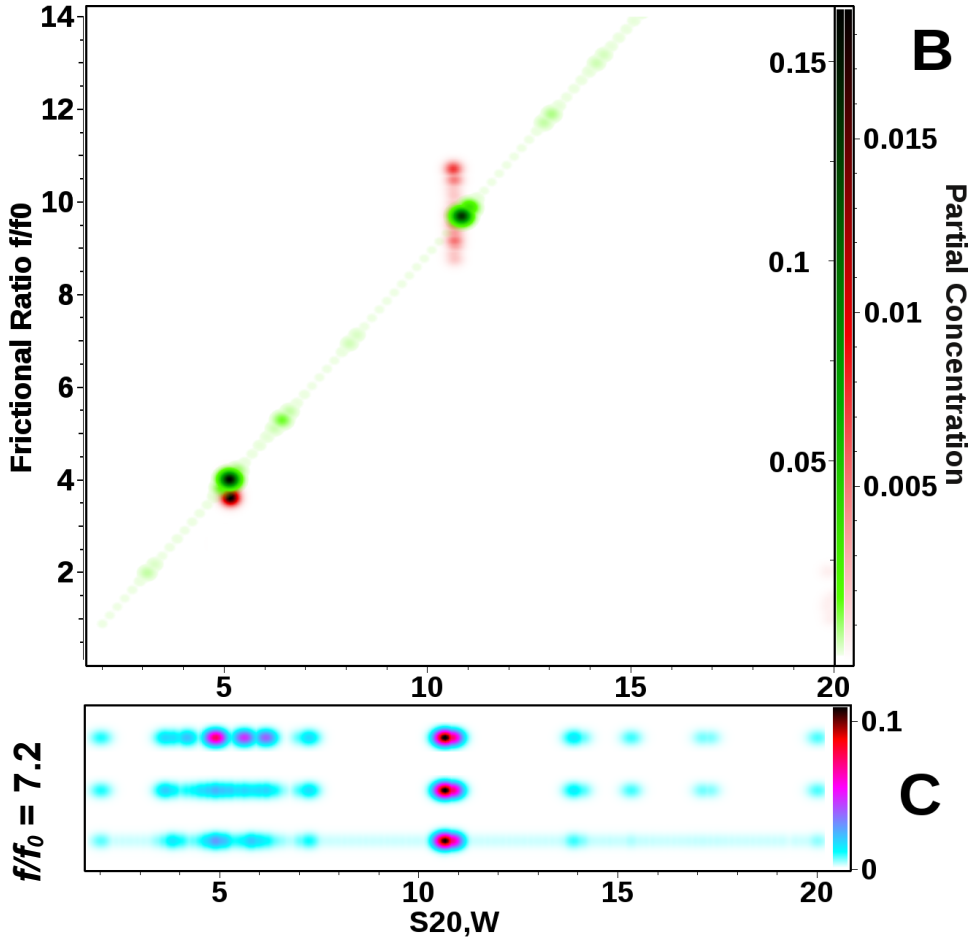
Straight line PCSA Monte Carlo results for two DNA fragments in 150 mM NaCl

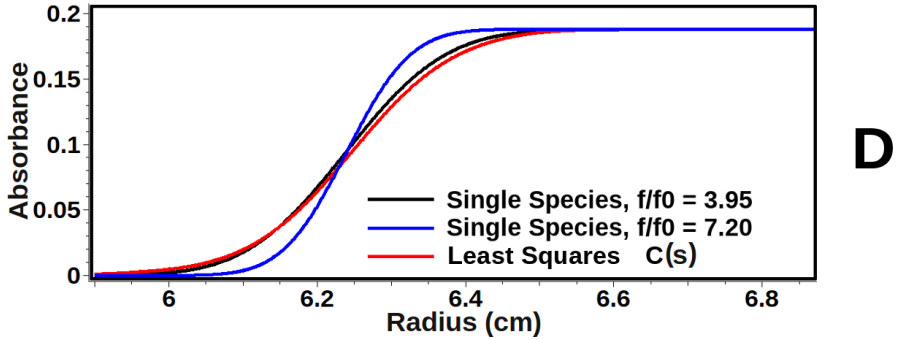


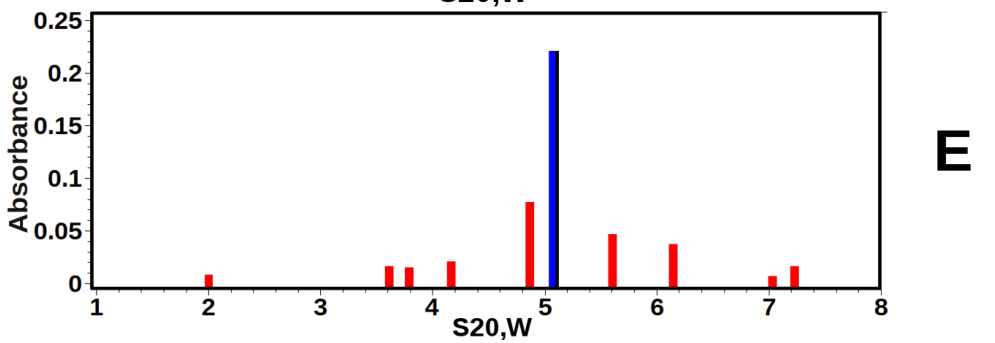


Parametrically Constrained Spectrum Analysis

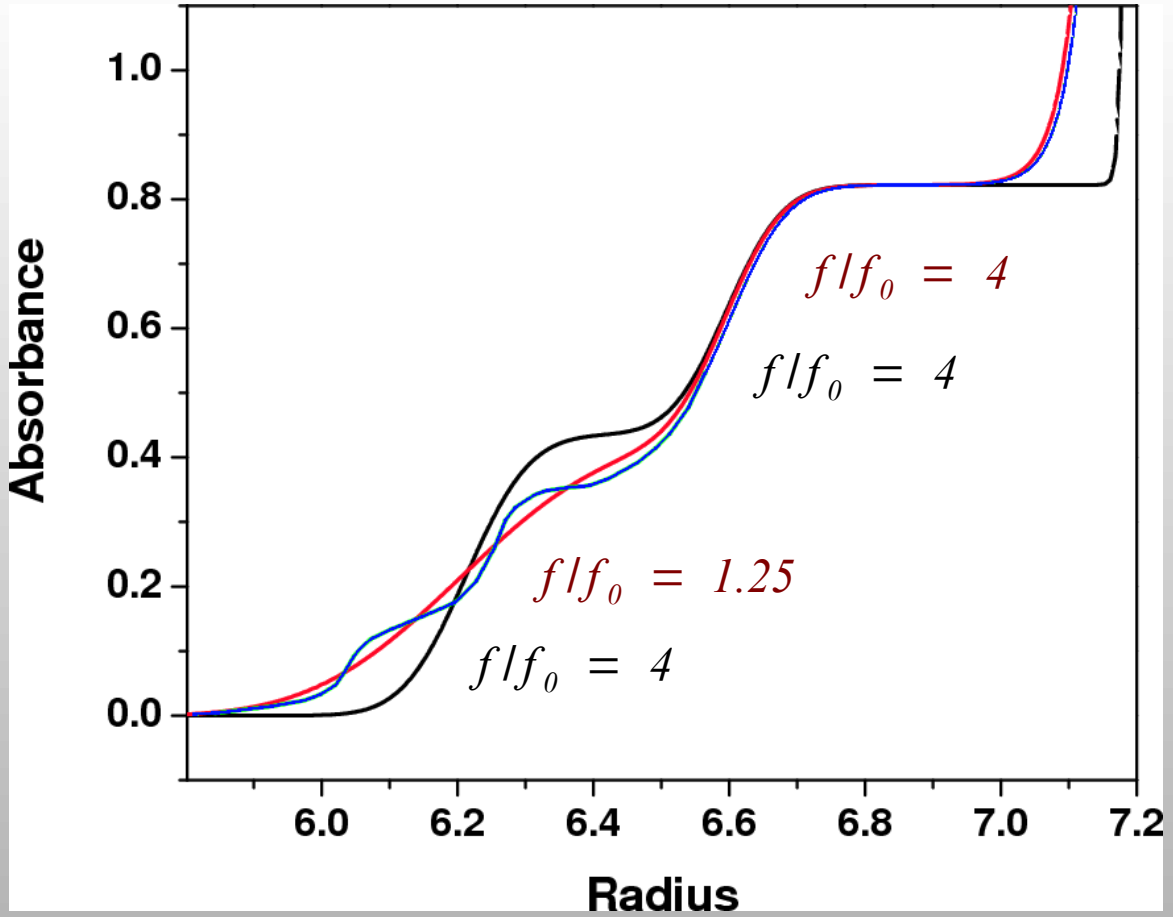




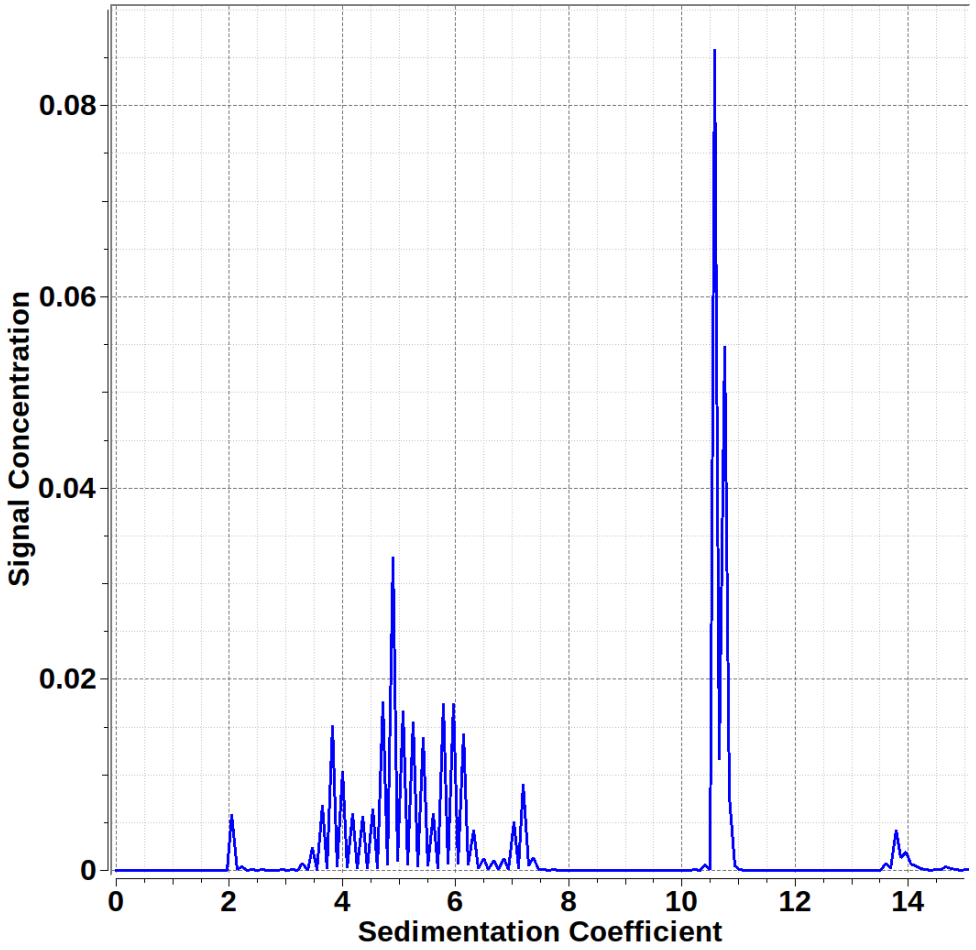




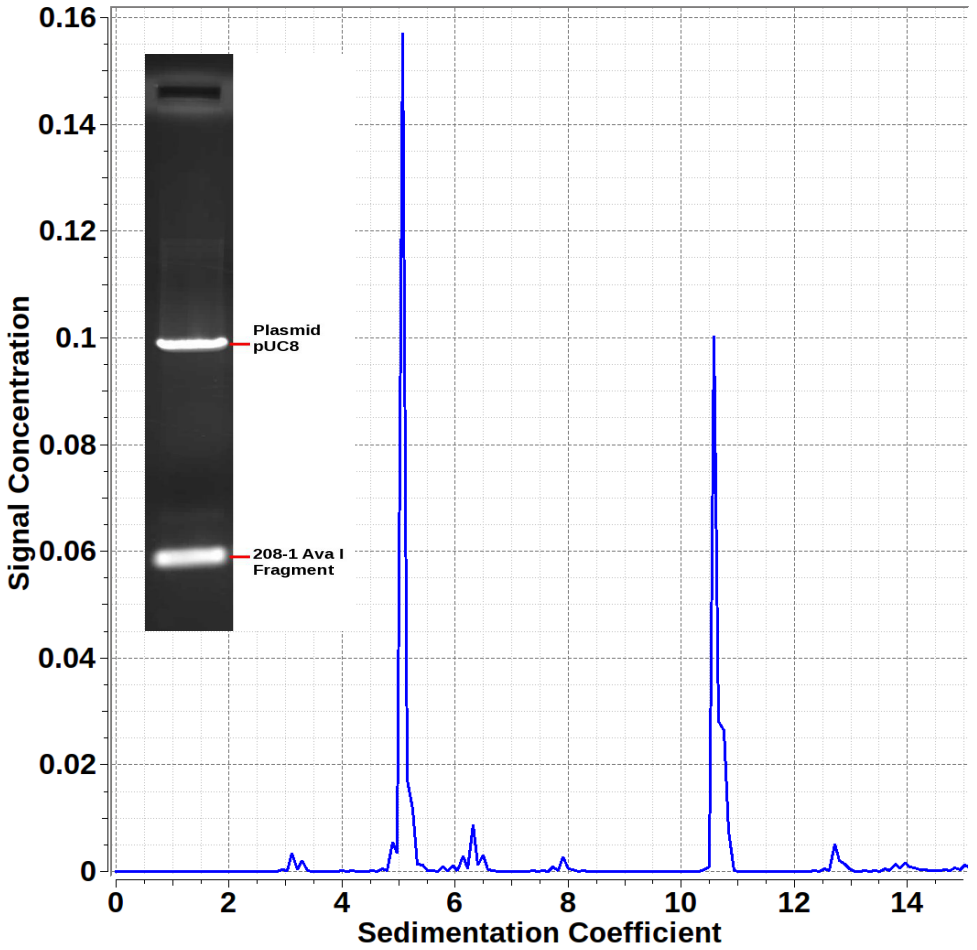
C(s)/C(MW) Method (P. Schuck)



Parametrically Constrained Spectrum Analysis

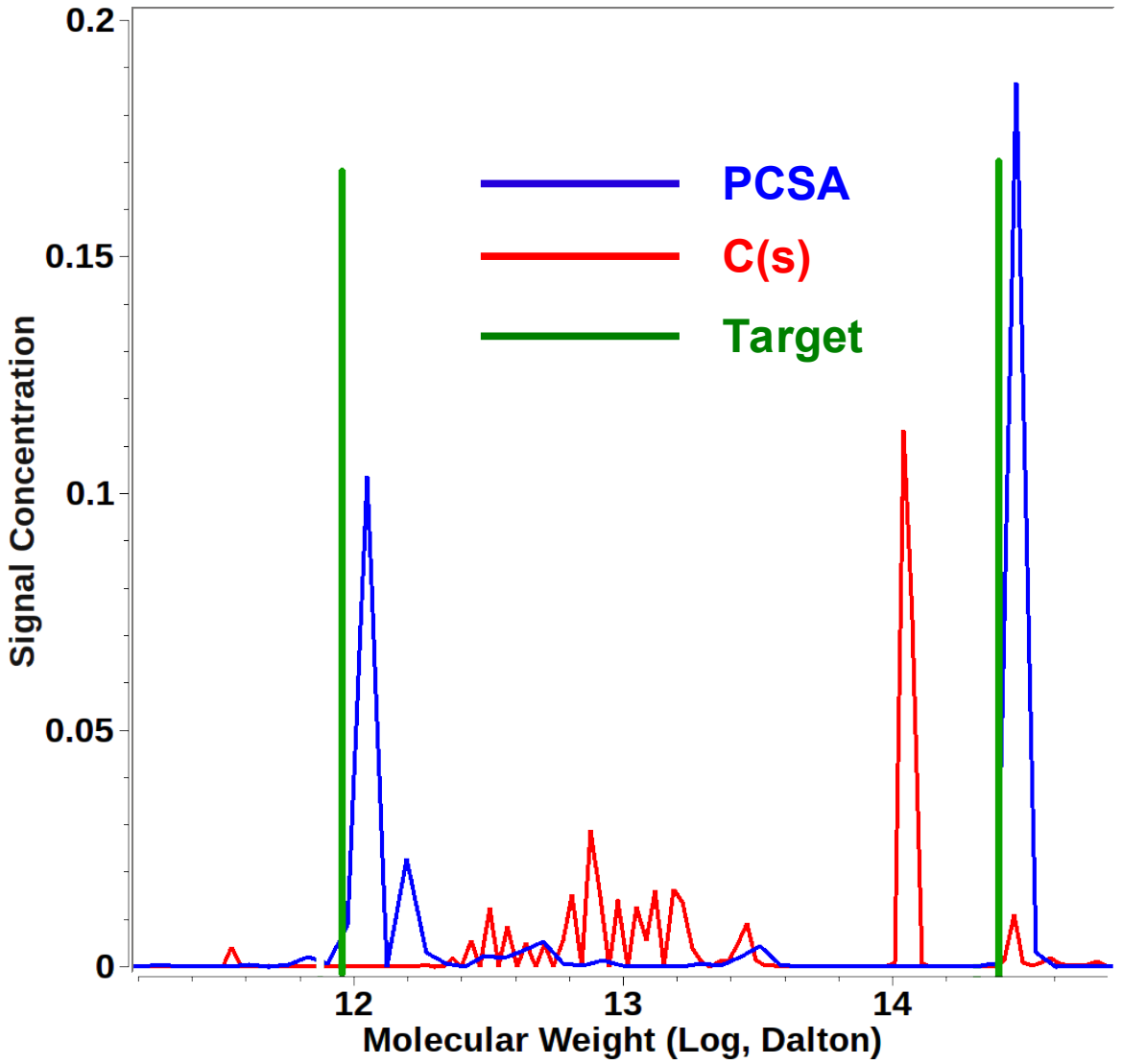


C(s) analysis,
high RMSD



PCSA analysis,
low RMSD

Parametrically Constrained Spectrum Analysis



C(s) is unreliable for fitting any velocity data except when anisotropy is constant. The PCSA method produces more reliable distributions and molar mass