Presenter: Borries Demeler

Topic:
Sedimentation III

Copy of Lecture at:

https://demeler.uleth.ca/biophysics/archive/Demeler/

Sedimentation Velocity:

Optimization Methods:

2-dimensional Spectrum Analysis (2DSA):

Provides degenerate, linear fit to experimental data over a finite domain, identifying regions with signal in the mass/shape domain, used to remove systematic noise contributions

Genetic Algorithms (GA):

Provides parsimonious regularization of 2DSA spectrum. Satisfies Occam's razor. Also used for fitting of discrete, non-linear models (reversible association, non-ideality, co-sedimenting solutes)

Monte Carlo Analysis (MC)

Used to measure the effect of noise on the fitted parameters, yields parameter distribution statistics

Parametrically Constrained Spectrum Analysis (PCSA)

Used to regularize 2-dimensional spectrum analysis. Enforce a unique mapping of one molar mass/sedimentation coefficient per frictional ratio.

Flow in the Ultracentrifuge Cell

Total Flow:

$$J = s\omega^2 r C - D\frac{\partial C}{\partial r}$$

Sedimentation Diffusion

Boundary Condition:

$$C = 0 \; for \; r < a \wedge r > b$$

Flow in sector-shaped analytical Ultracentrifuge cell:

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

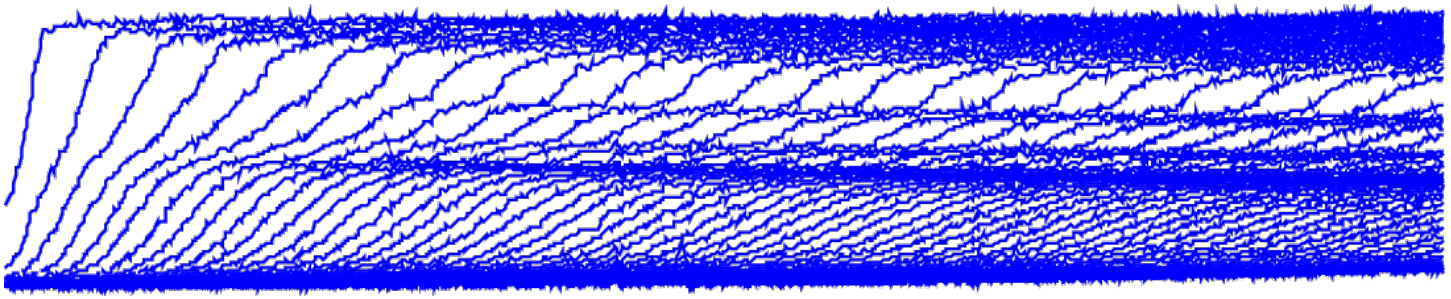
(Lamm Equation)

The Lamm Equation can be solved with the finite element Method

Cao W and 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.

Cao, W and Demeler B. Modeling Analytical Ultracentrifugation Experiments with an Adaptive Space-Time Finite Element Solution for Multi-Component Reacting Systems. Biophys. J. (2008) 95(1):54-65

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:

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

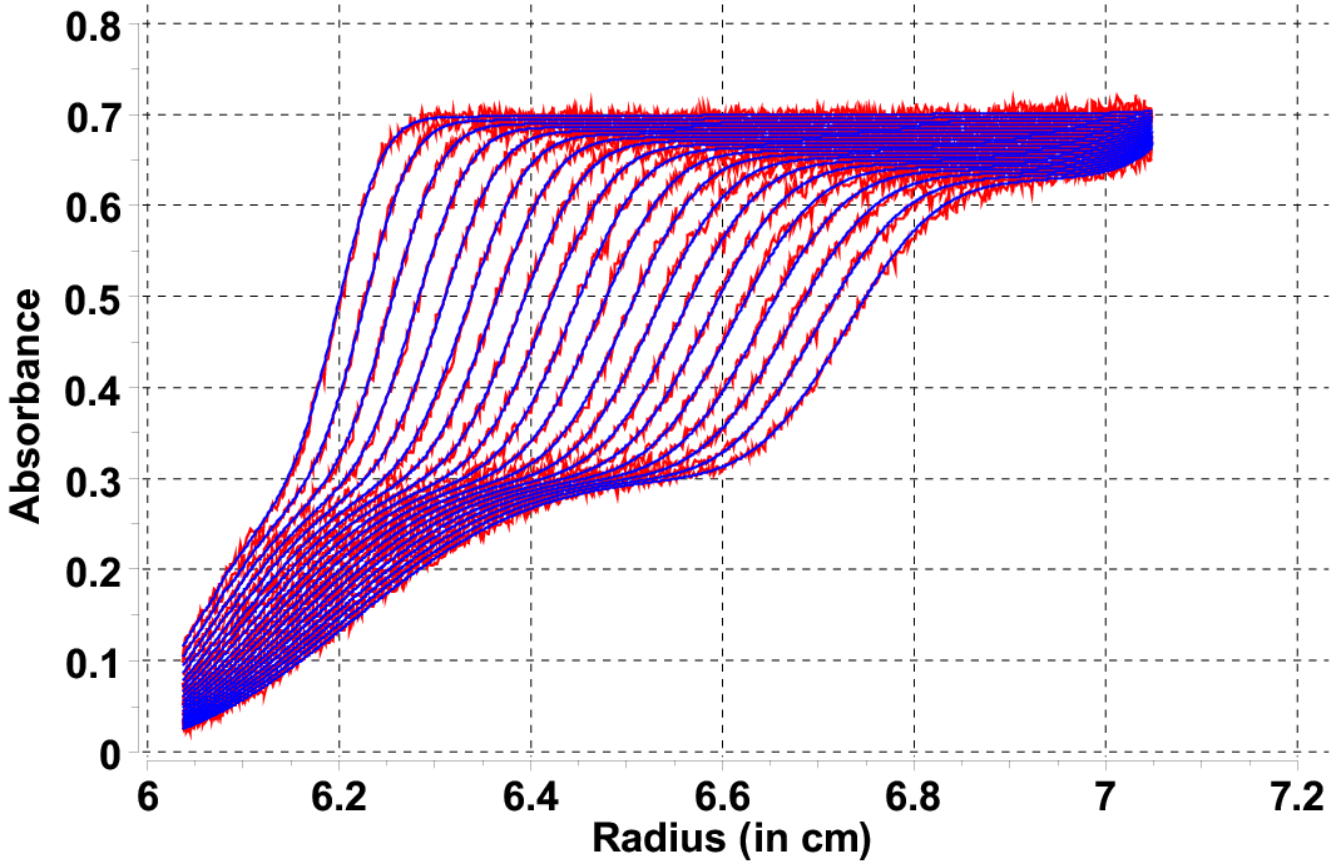
The model is compared to the experimental data in the least squares sense for each data point in the experiment (over time and radius)

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

here, c, b, s and D are nonlinear parameters, and are adjusted independently in an iterative fit.

Nonlinear Least Squares Finite Element Fitting

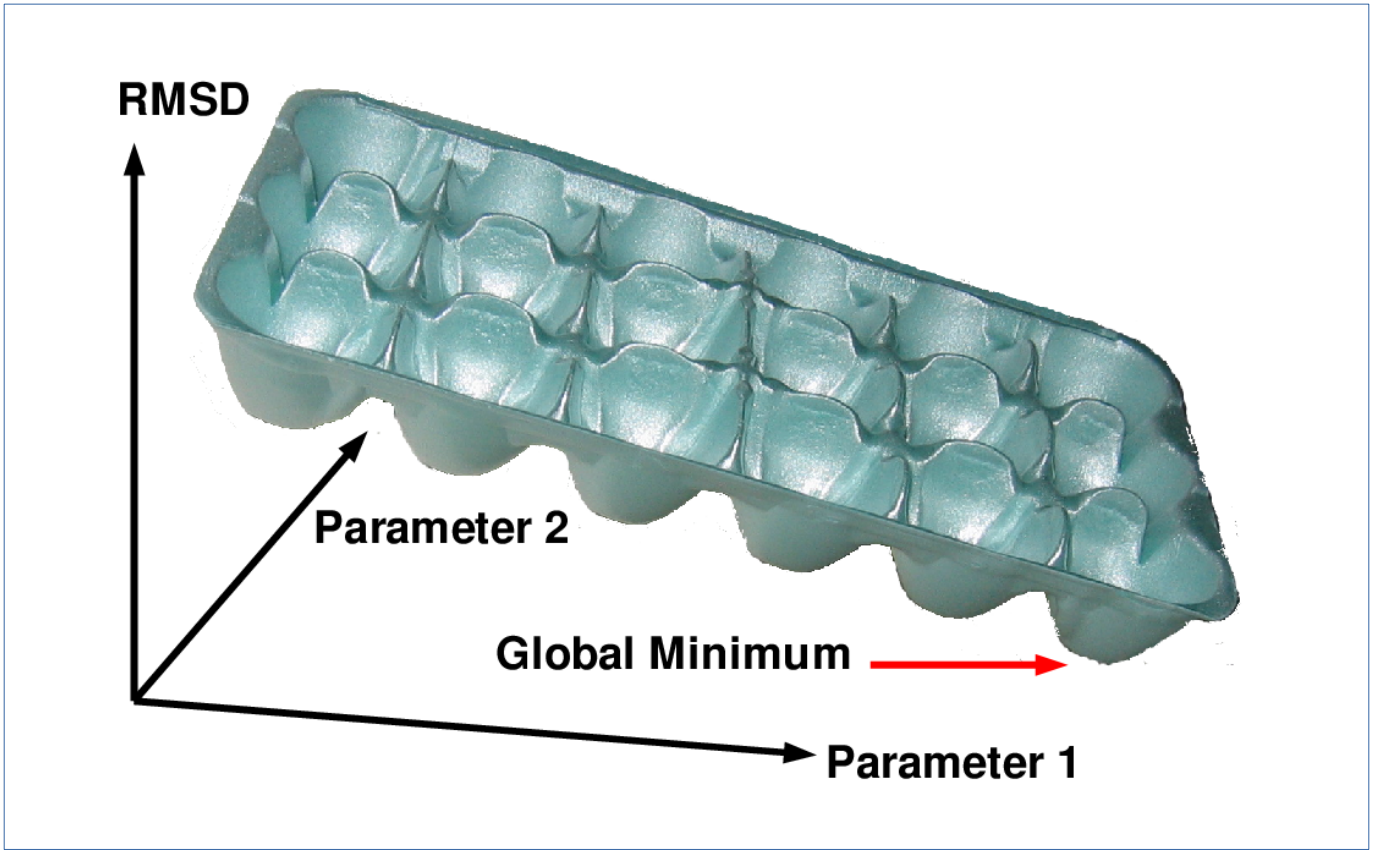
Finite Element - Nonlinear Least Squares (RMSD: 4.61 x 10^{-3}) Monte Carlo is needed to define statistical confidence of fitted parameters.



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.

Linear Approach

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

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

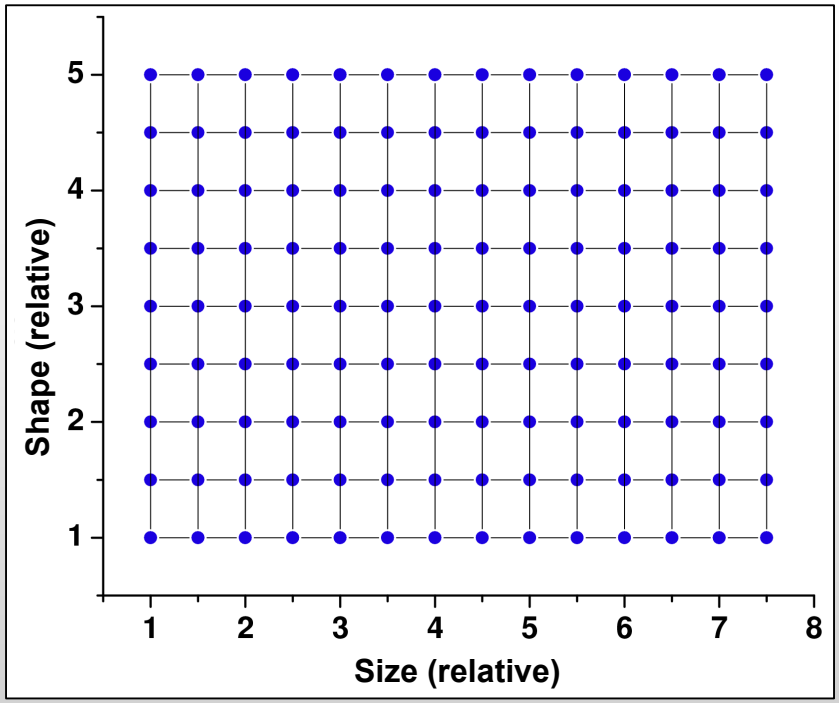
$$Min \sum_{i=1}^{m}\left[\sum_{j=1}^{n}\left[\,c_j\,L\left(s_j, D_j\right)\right]_i - \boldsymbol{Y_i}\right]^2$$

Note: This generates a grid over all possible s and D values. Each s and D pair in the grid represents a complete solution of the Lamm equation, and the Amplitude c_j defines the partial concentration of each pair.

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*

2-Dimensional Spectrum Analysis



Build a 2-dimensional grid of shape (s and D) and size (s) values

Find the concentrations of the parameter pairs that match the original data when modeled by least squares. This is a linear problem that avoids the pitfalls of nonlinear LS optimization

Locate signal in the 2-dimensional spectrum space

$$M = \sum_{i}^{size} \sum_{j}^{shape} c_{i,j} L\left(s_{i,j},\ D_{i,j}\right)$$

$$Min \sum_{l}^{radius} \sum_{k}^{time} \left[M_{lk} - b_{lk} \right]^2$$

Solve $\| Mc - b \|_2$ with NNLS

2-Dimensional Spectrum Analysis

This is 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:

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

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

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

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

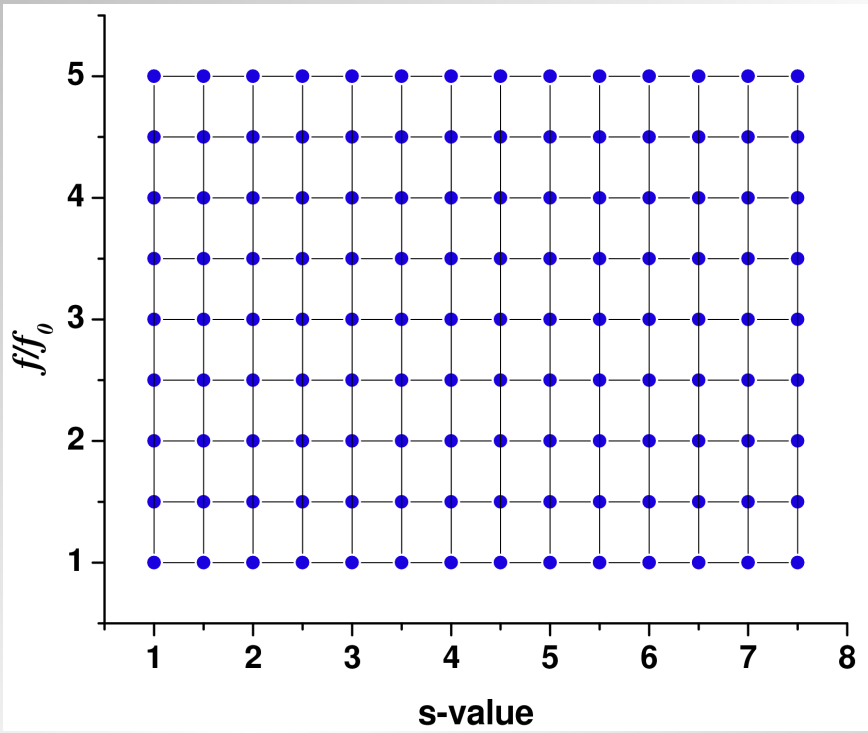
n = # of sedimentation value grid points (~30 - 50)

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

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

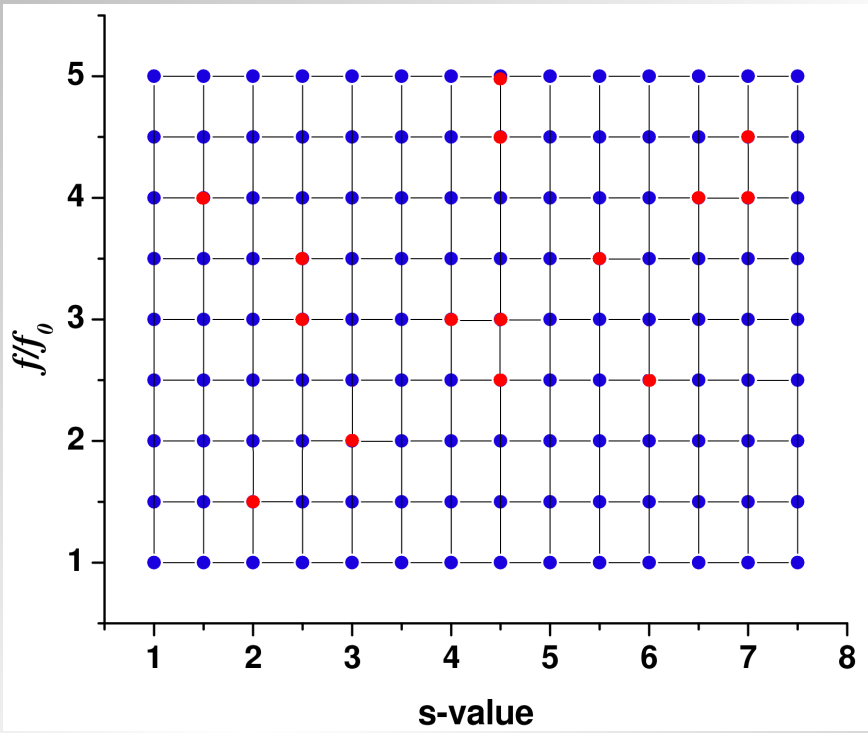
2-D Spectrum Analysis - Refinement:

Step 1: Start with a coarse grid definition:



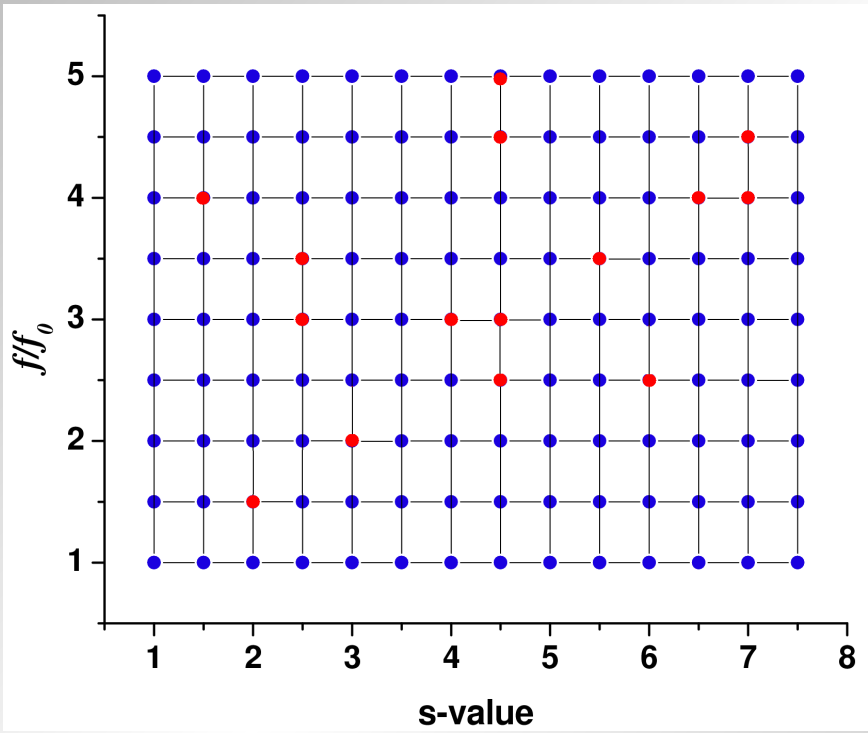
2-D Spectrum Analysis - Refinement:

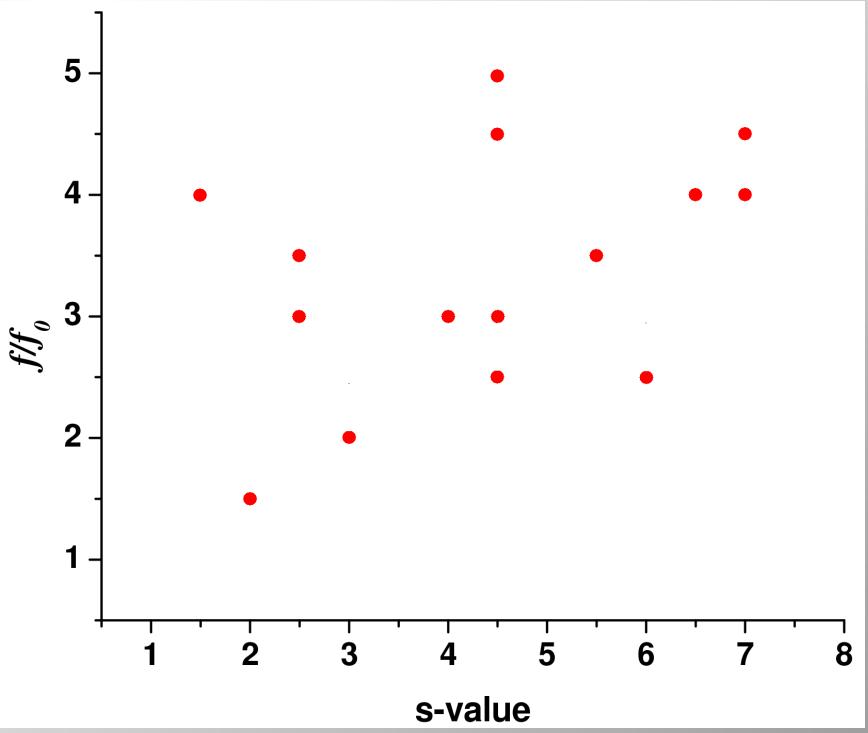
Step 2: Perform NNLS



2-D Spectrum Analysis - Refinement:

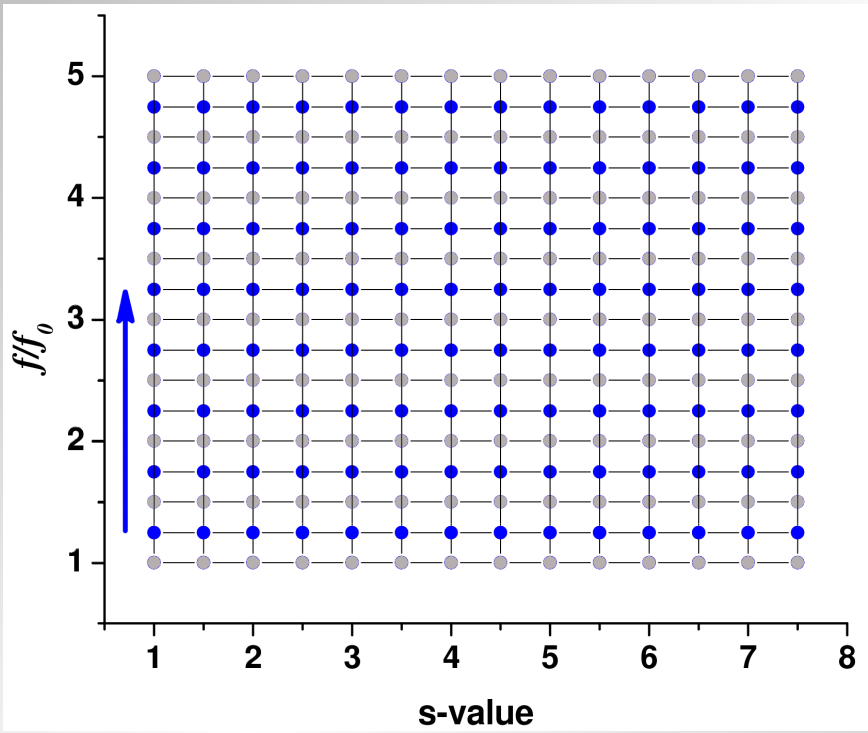
Step 3: Save non-zero elements into a separate array

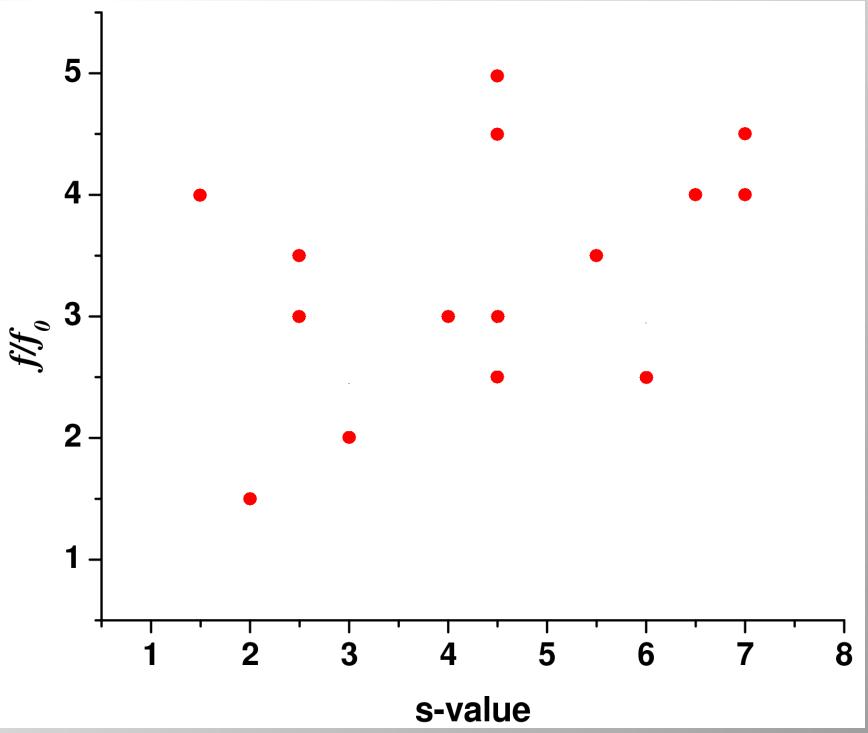




2-D Spectrum Analysis - Refinement:

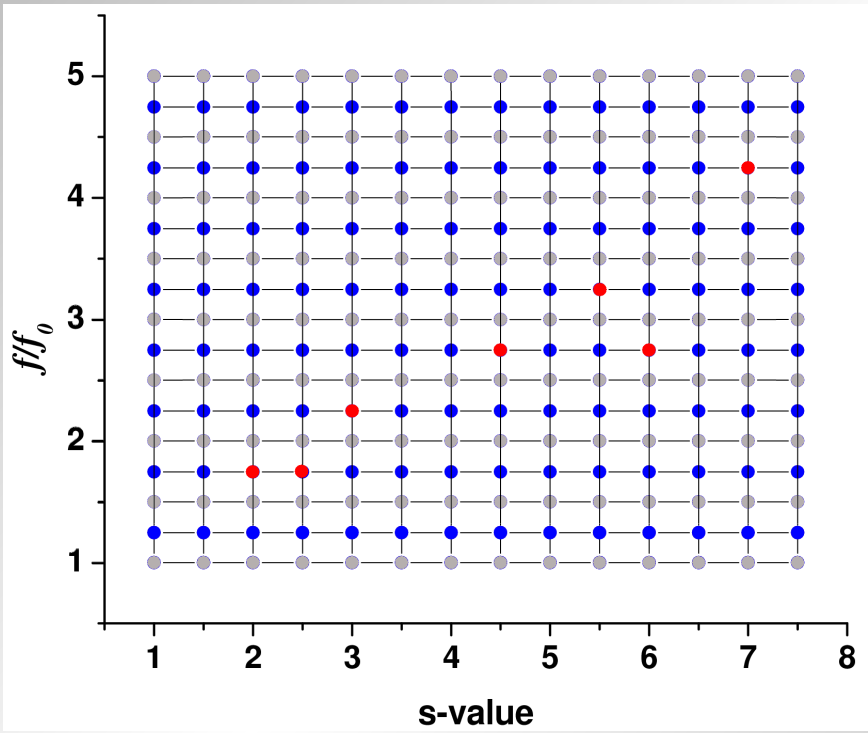
Step 4: Shift grid into Y-direction

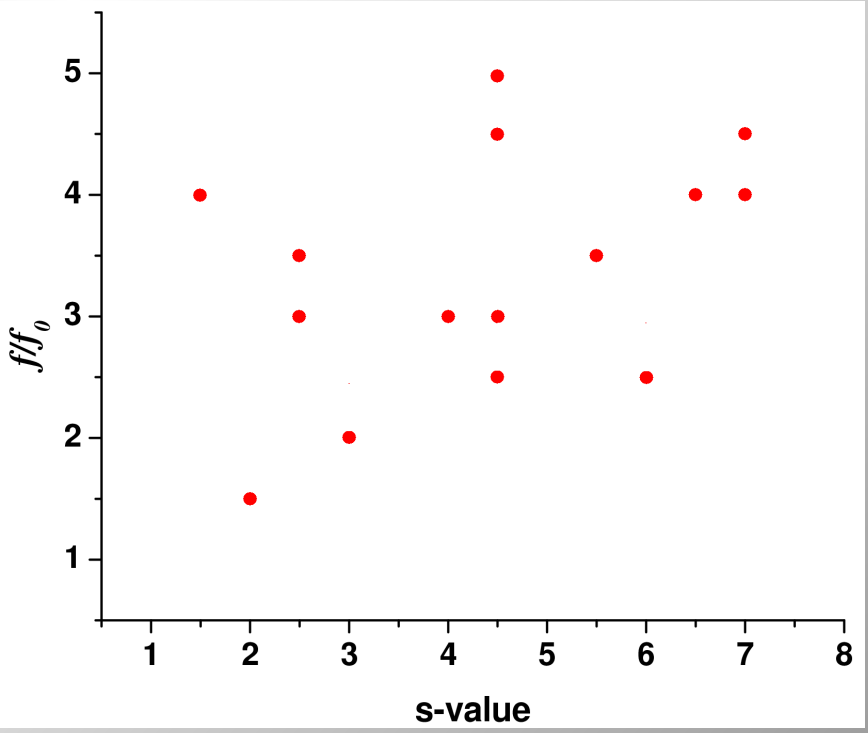




2-D Spectrum Analysis - Refinement:

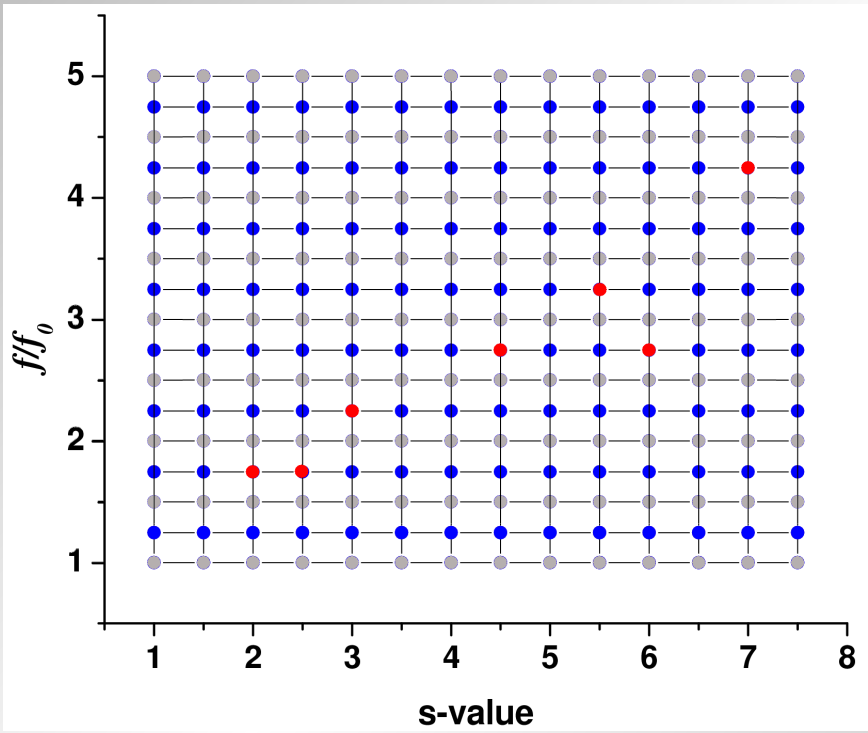
Step 5: Perform NNLS again, but only on the shifted grid (blue)

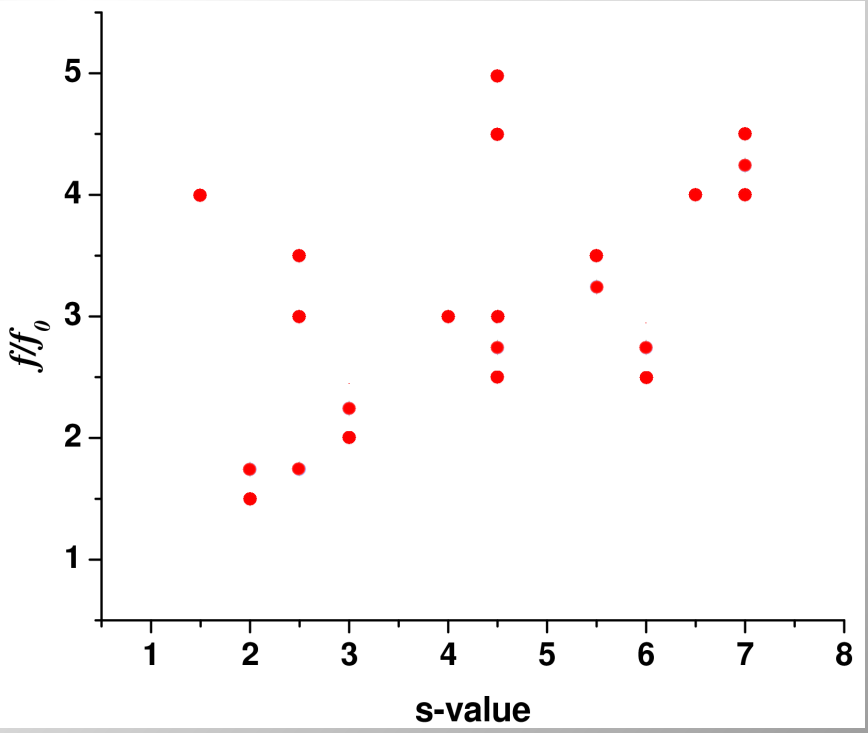




2-D Spectrum Analysis - Refinement:

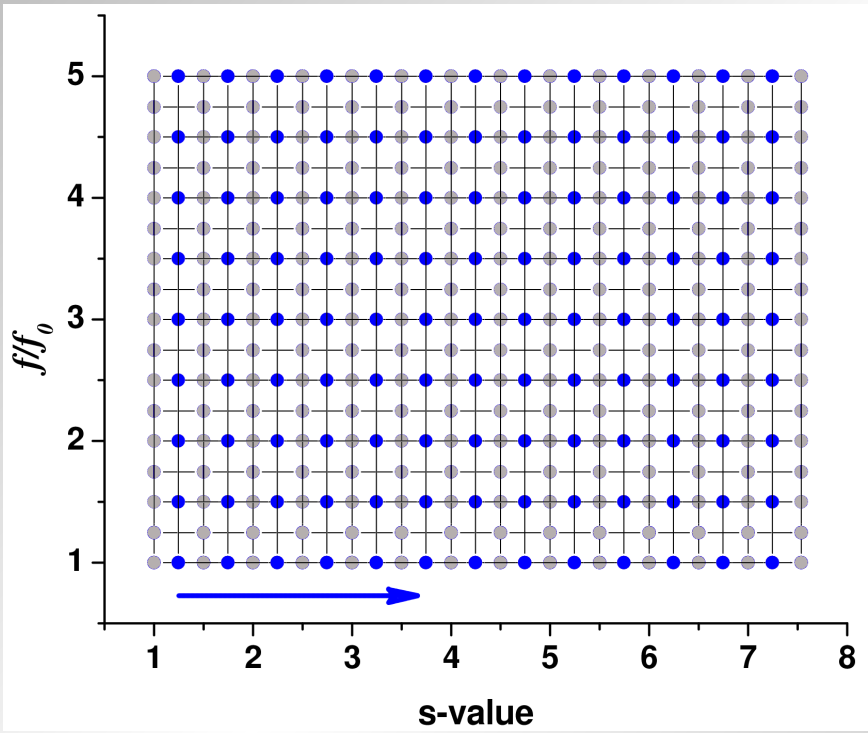
Step 6: Add the newly found non-zero elements to the stored array

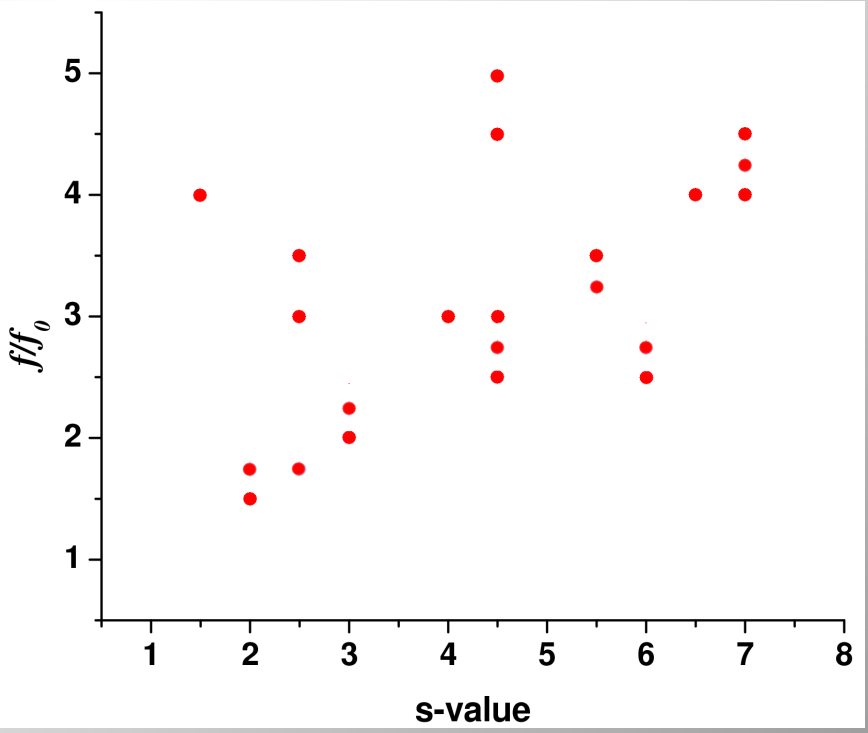




2-D Spectrum Analysis - Refinement:

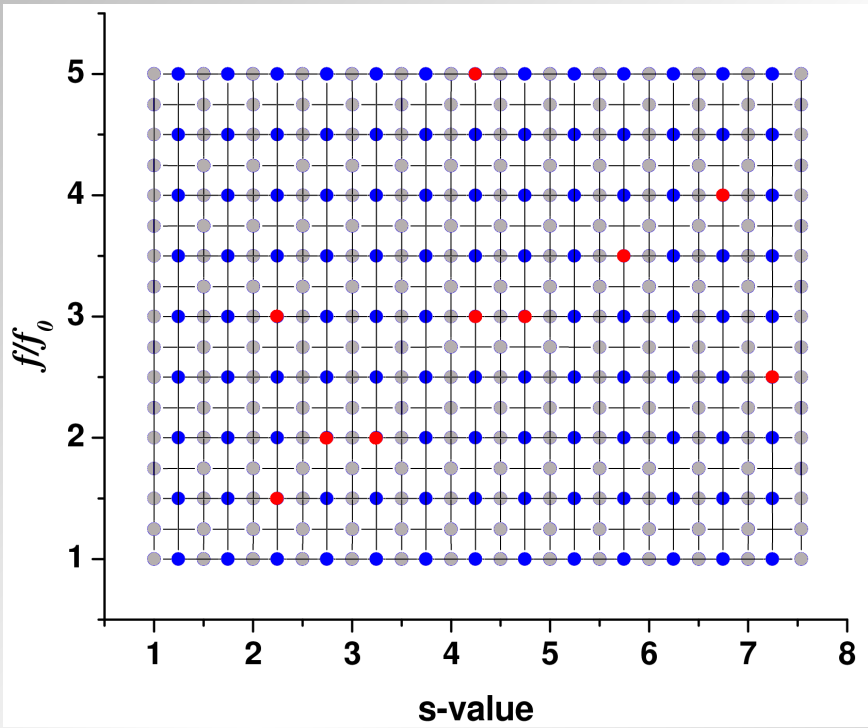
Step 7: Now shift the grid into the X-direction

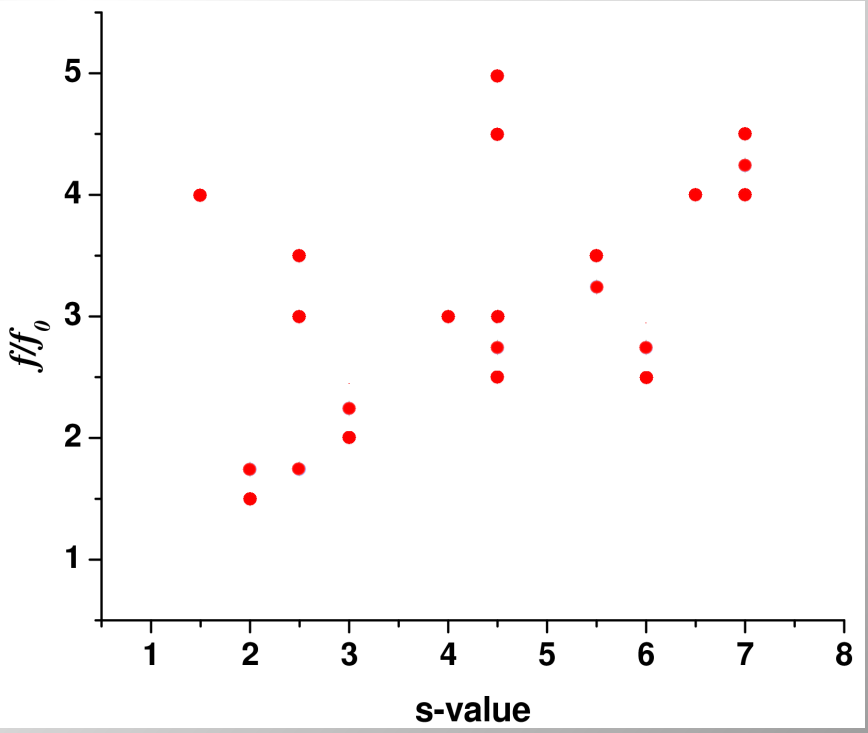




2-D Spectrum Analysis - Refinement:

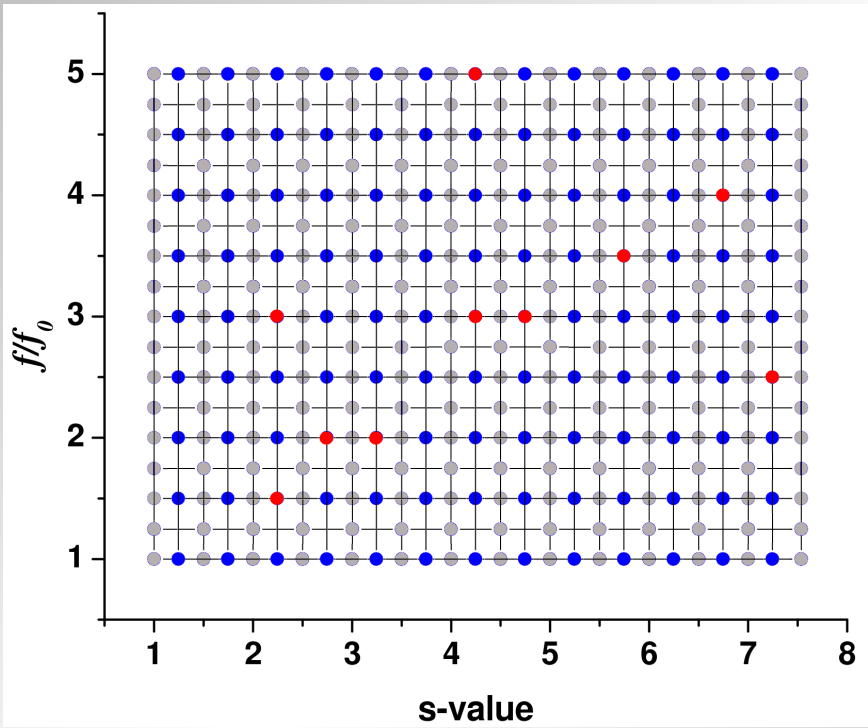
Step 8: Perform NNLS on the shifted grid again

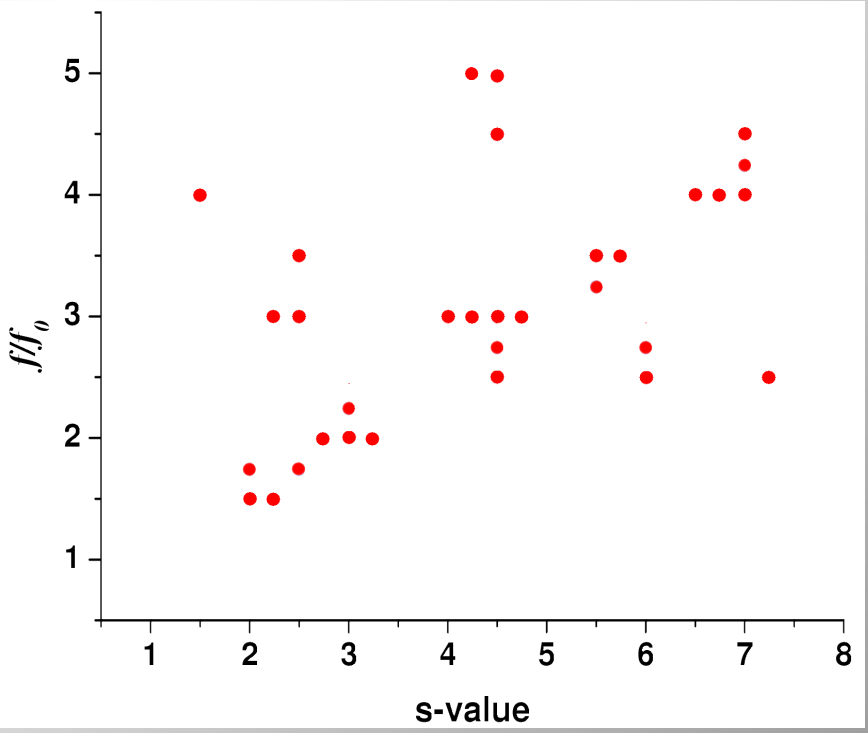




2-D Spectrum Analysis - Refinement:

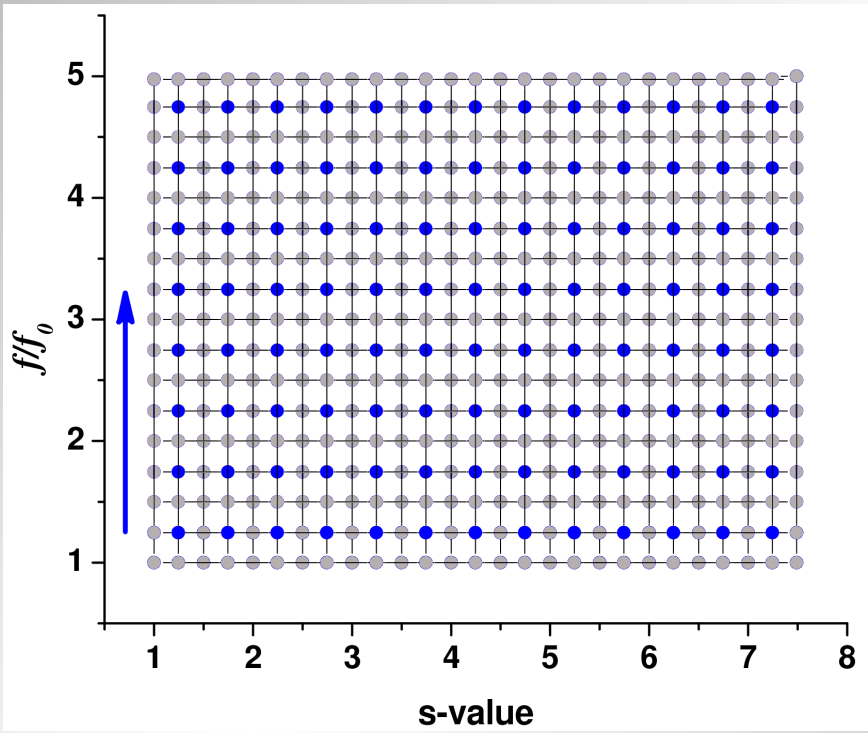
Step 9: Add the new non-zero elements to the stored array

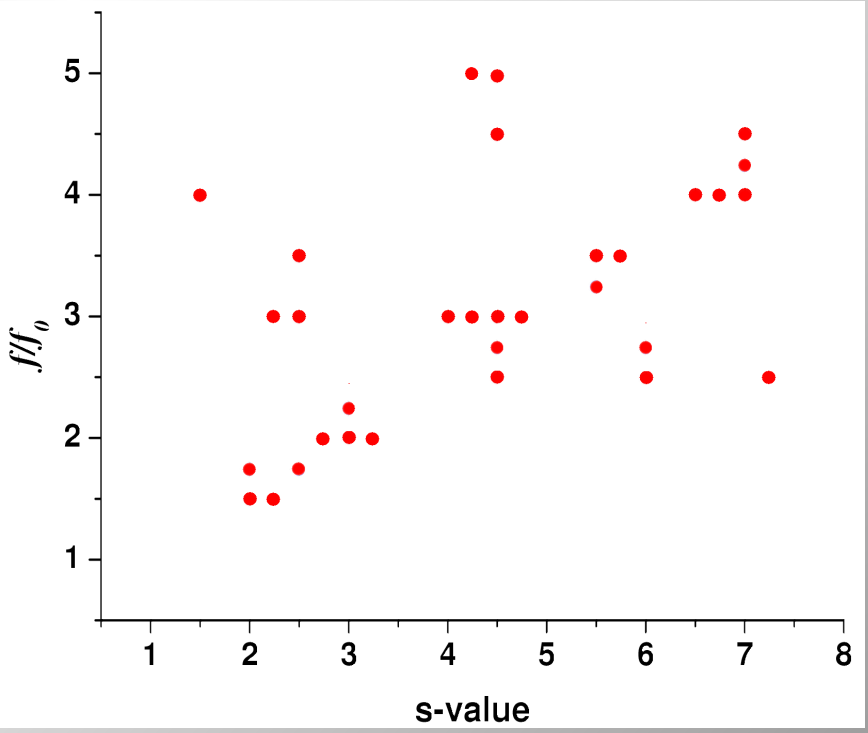




2-D Spectrum Analysis - Refinement:

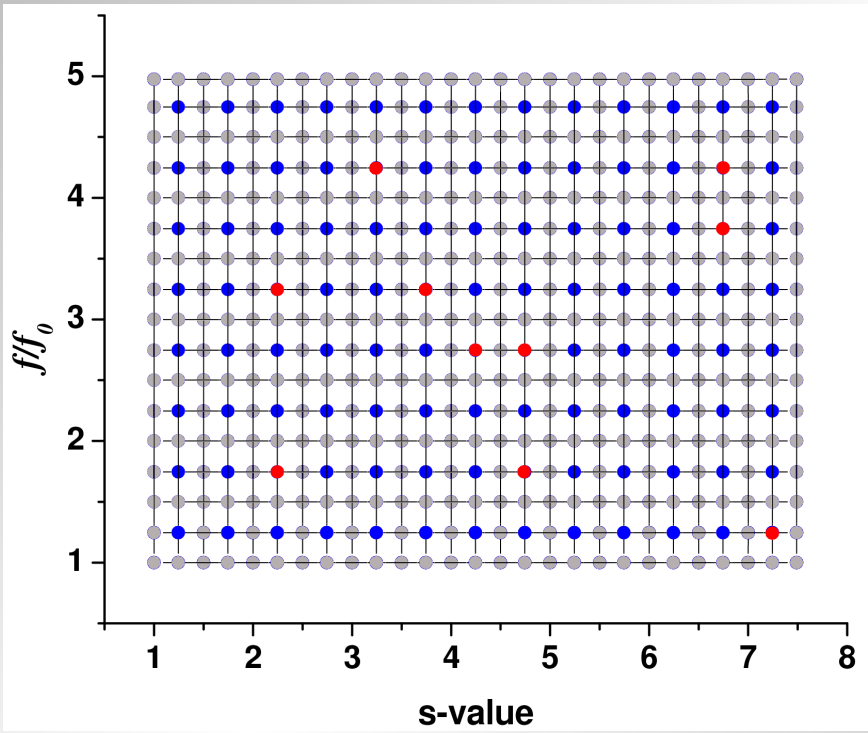
Step 10: Complete the square and shift the grid once more in the Y-direction

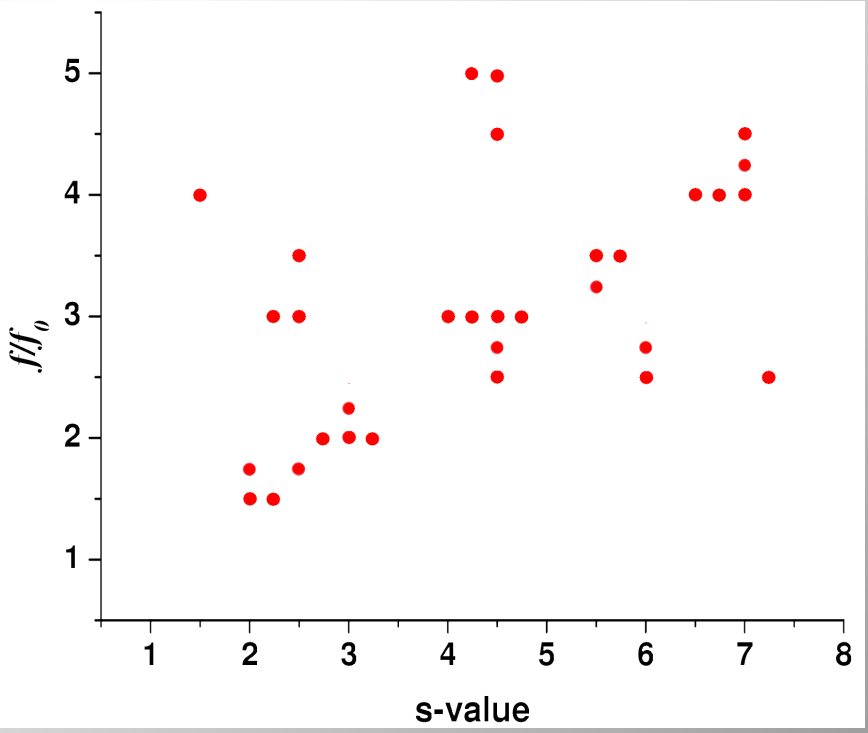




2-D Spectrum Analysis - Refinement:

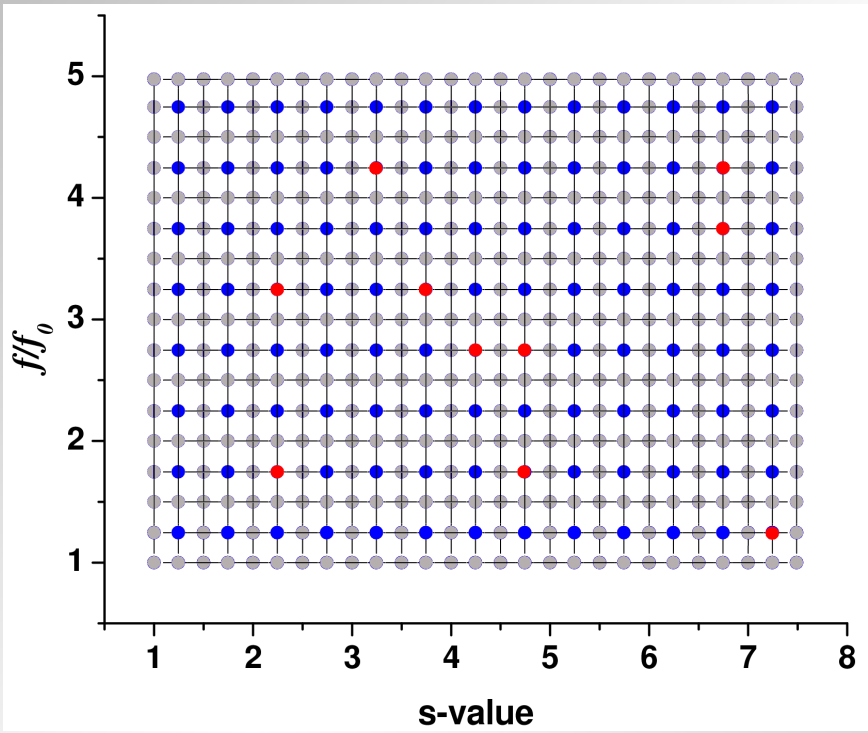
Step 11: Perform NNLS on the new grid

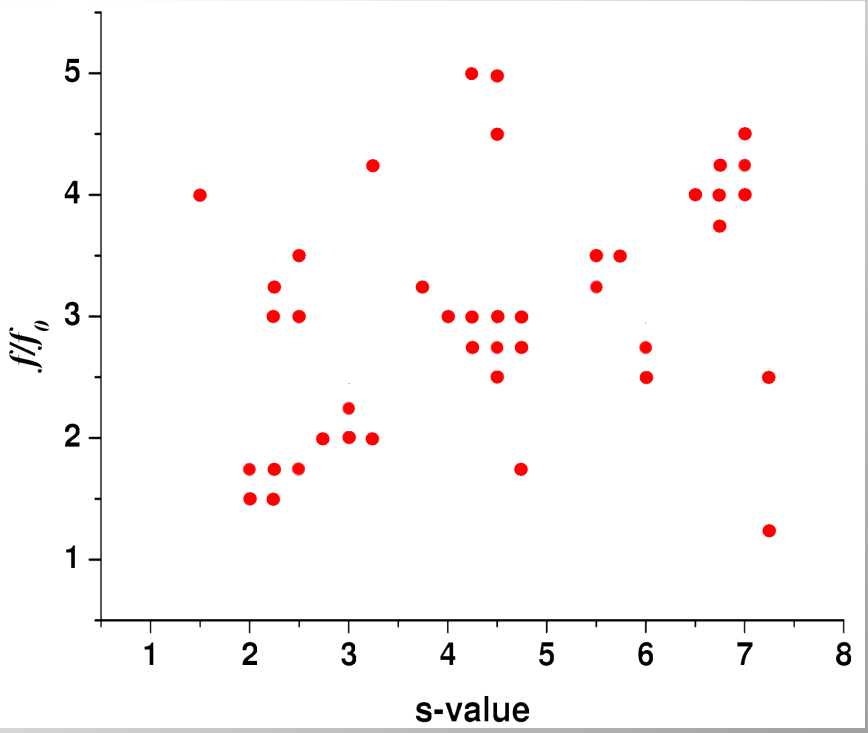




2-D Spectrum Analysis - Refinement:

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



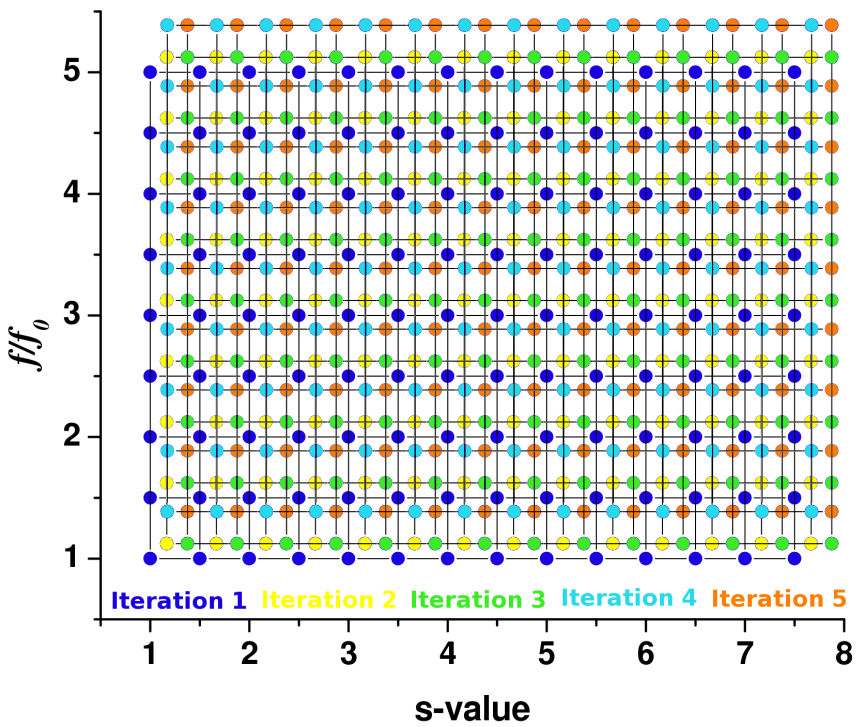


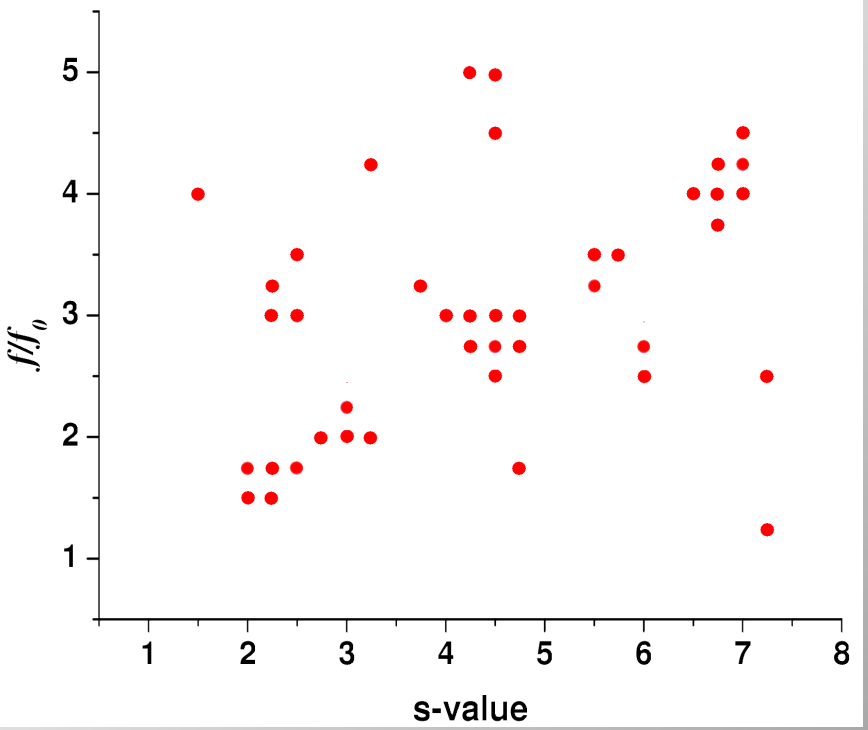
2-D Spectrum Analysis - Refinement:

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:

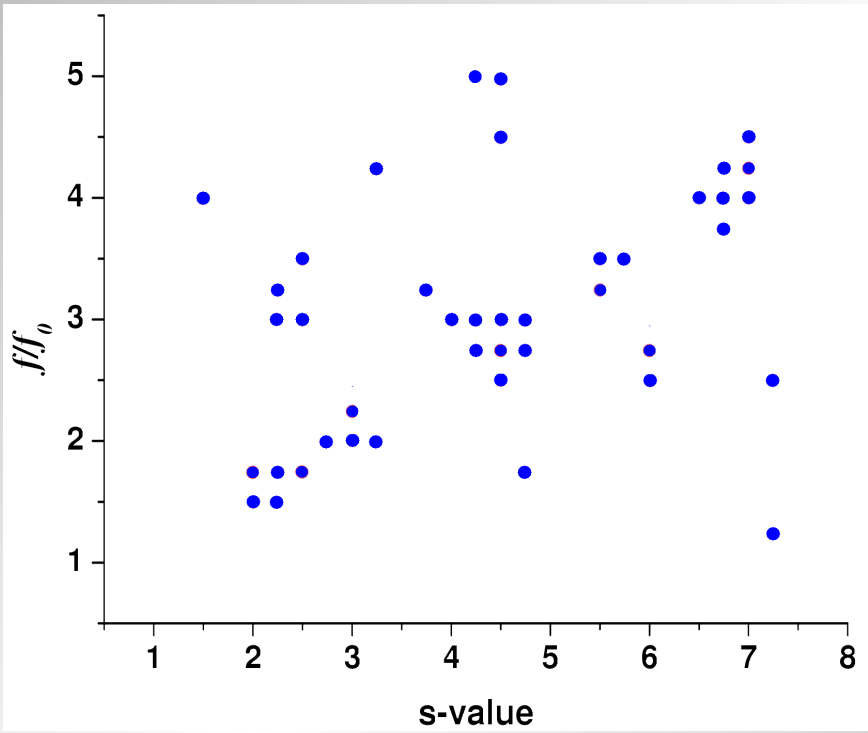


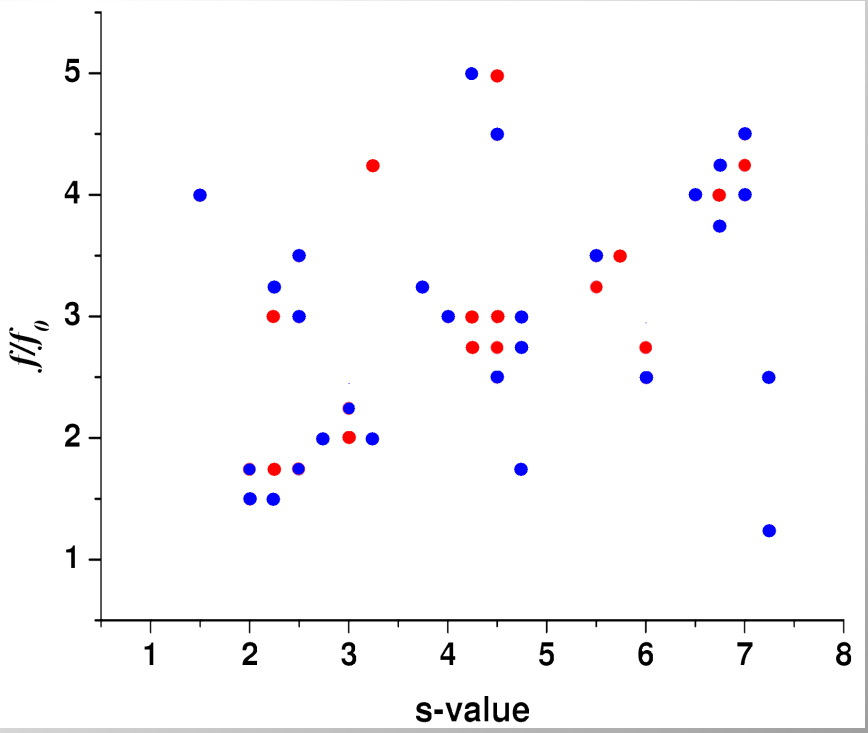


Final result is fairly sparse, but it is also degenerate, includes false positives and needs further refinement. It can be used to identify regions that contain signal.

2-D Spectrum Analysis - Refinement:

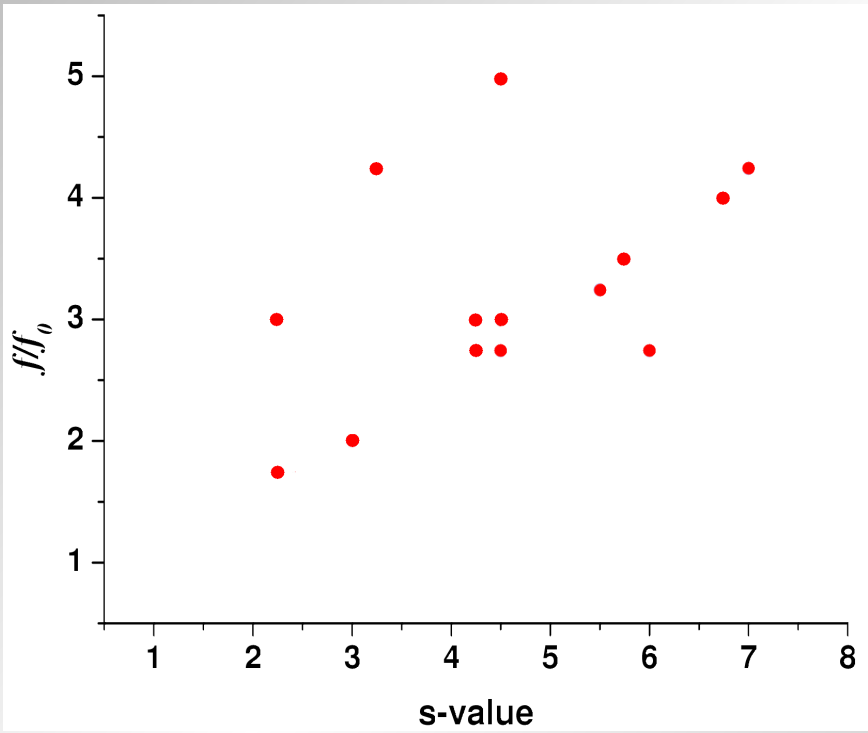
Step 13: Now take the storage array and perform one last NNLS iteration on it to filter out unnecessary elements

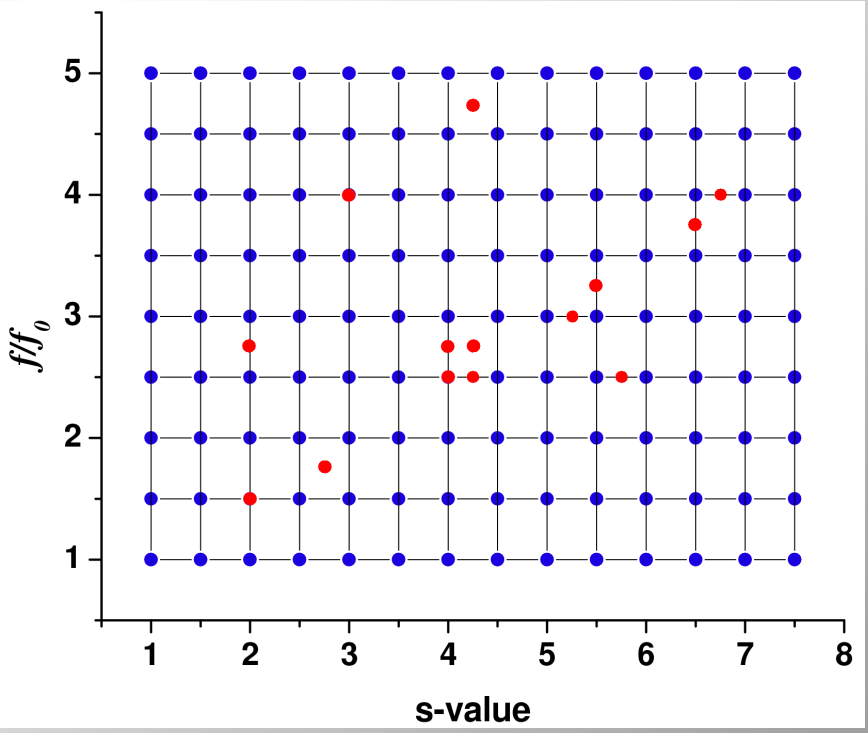




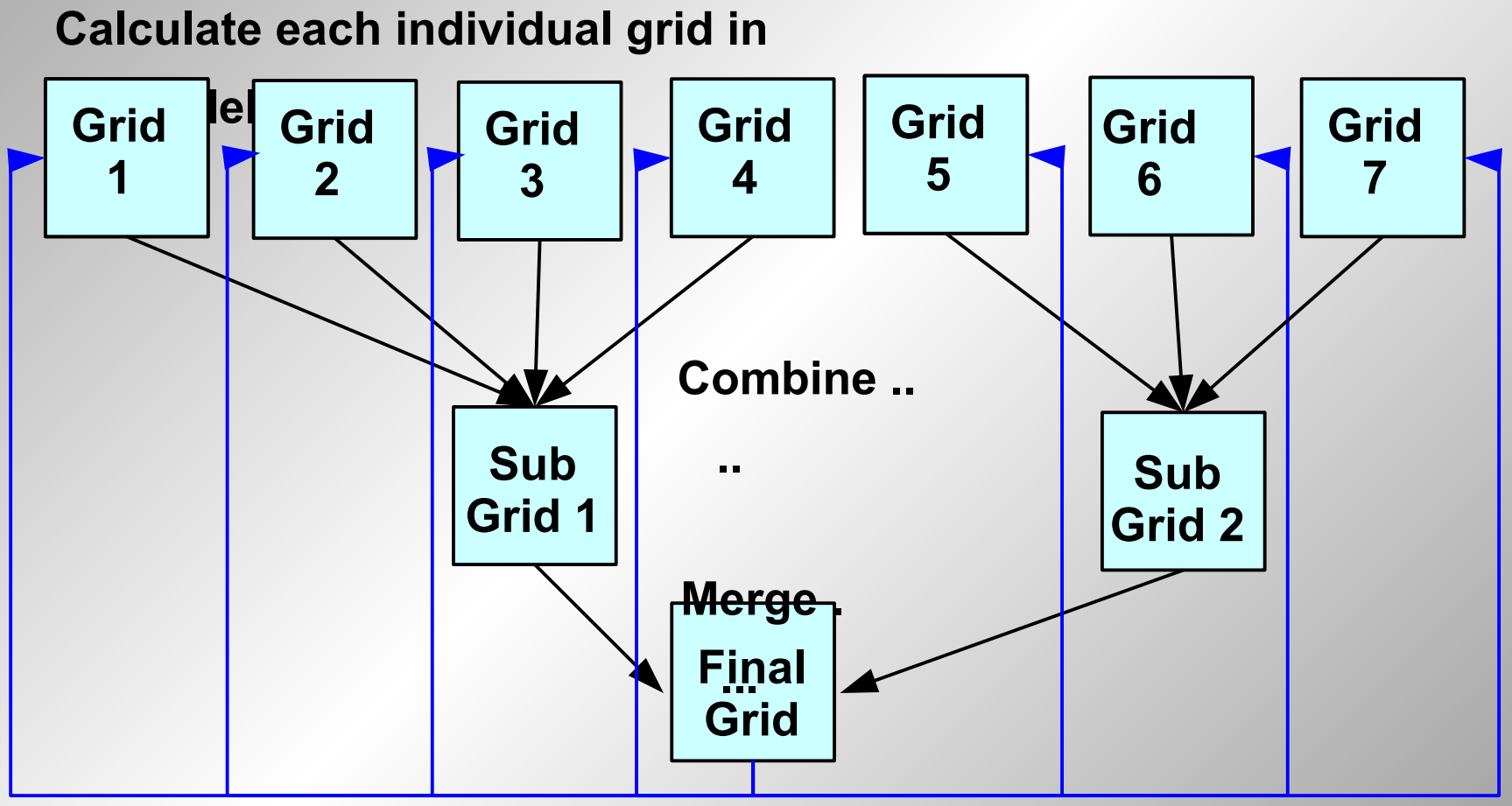
2-D Spectrum Analysis - Refinement:

Step 14: Take the final grid and add it back to all starting grids. Redo the analysis until there is no more change to guarantee equivalence with the original fine-grained grid.





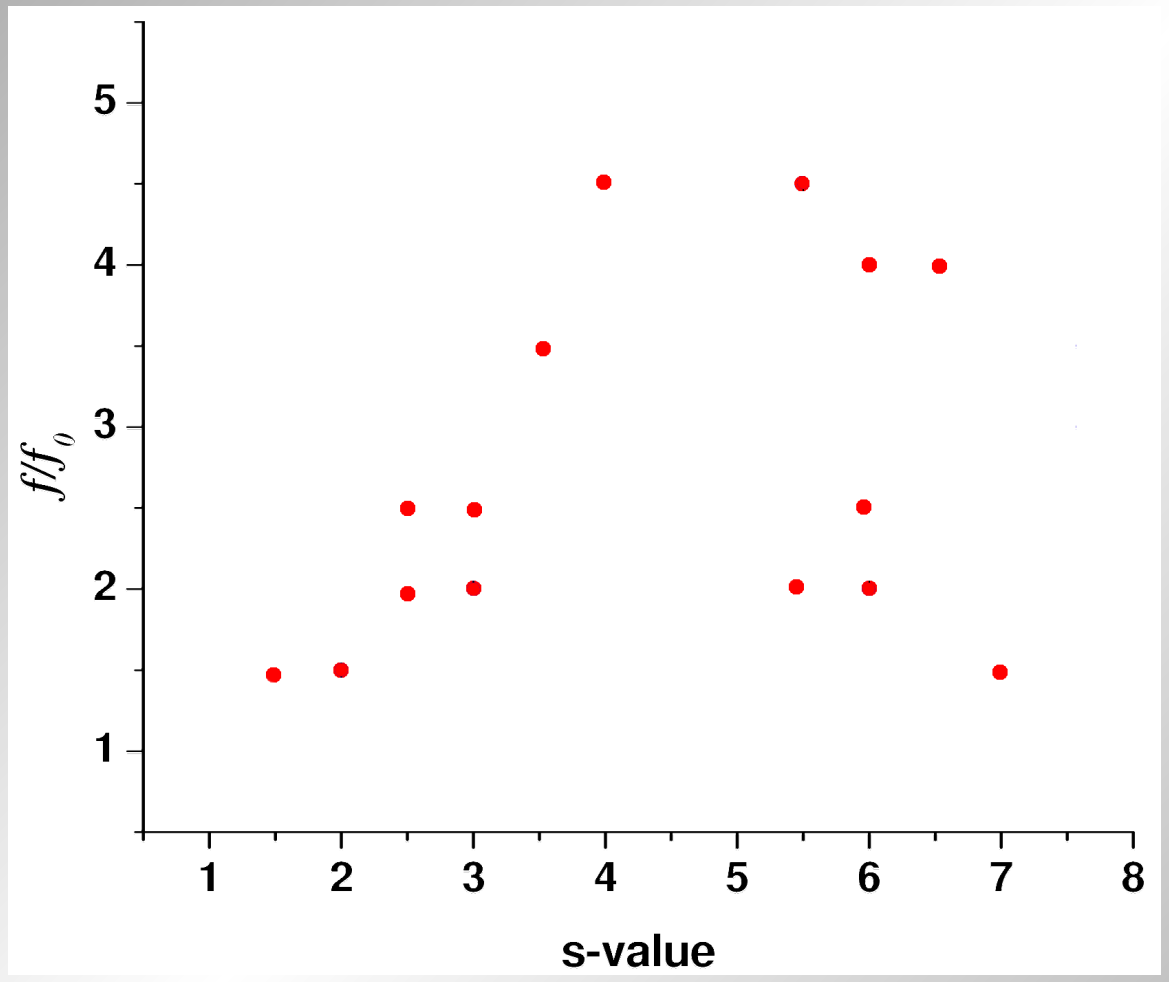
Moving Grid Approach – parallel HPC implementation



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

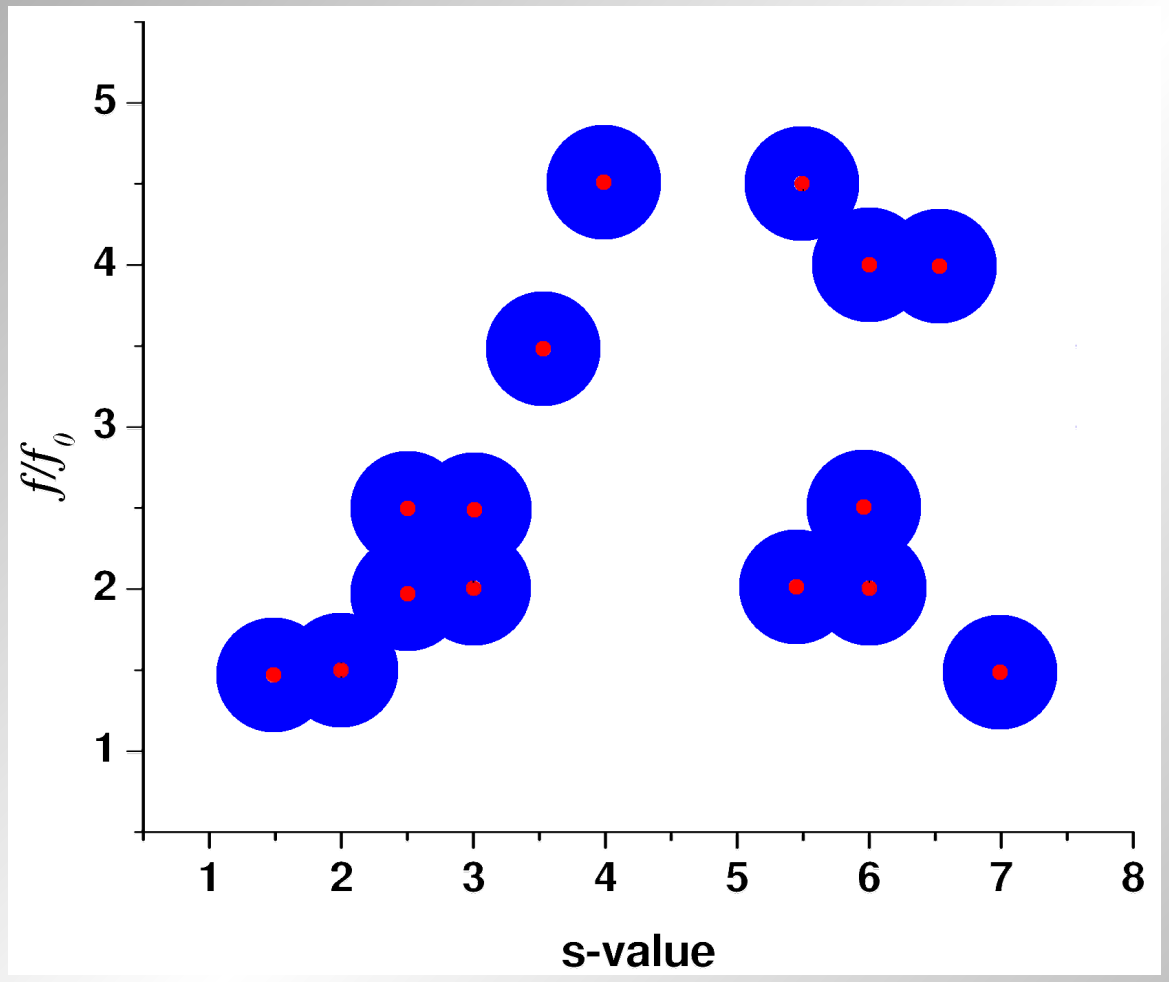
Iterate until there is no more change ....

Final Result is used to initialize GA



Final Solution is sparse, but still on a grid...

Final Result is used to initialize GA



Idea:

Build probability surfaces around each non-zero entry and use the surface to initialize the GA.

Surfaces can be circular, elliptical, or rectangular

Probabilities from neighboring points add up.

Genetic Algorithms (GA)

Genetic algorithms provide a stochastic optimization method

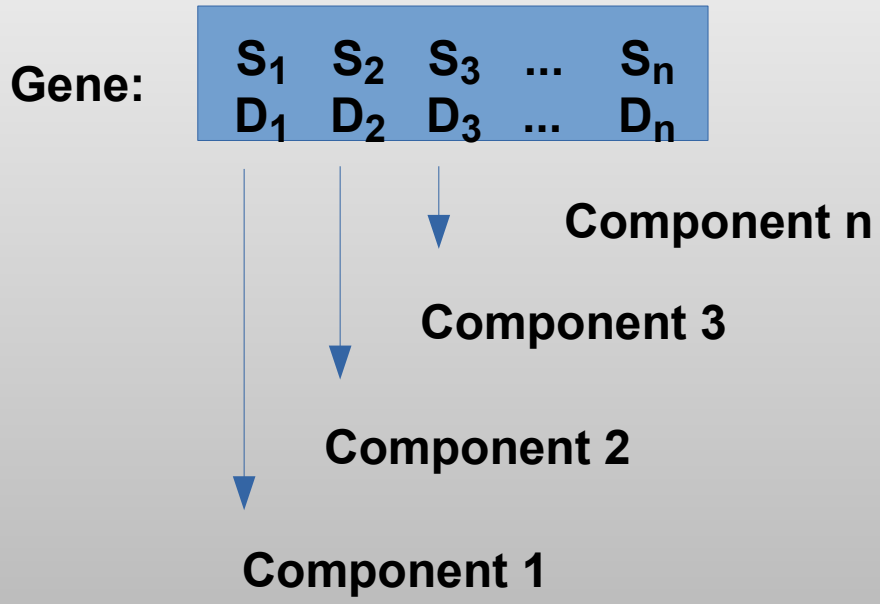
John H Holland, Adaption in Natural and Artificial Systems, 1975, U. of Michigan Press

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

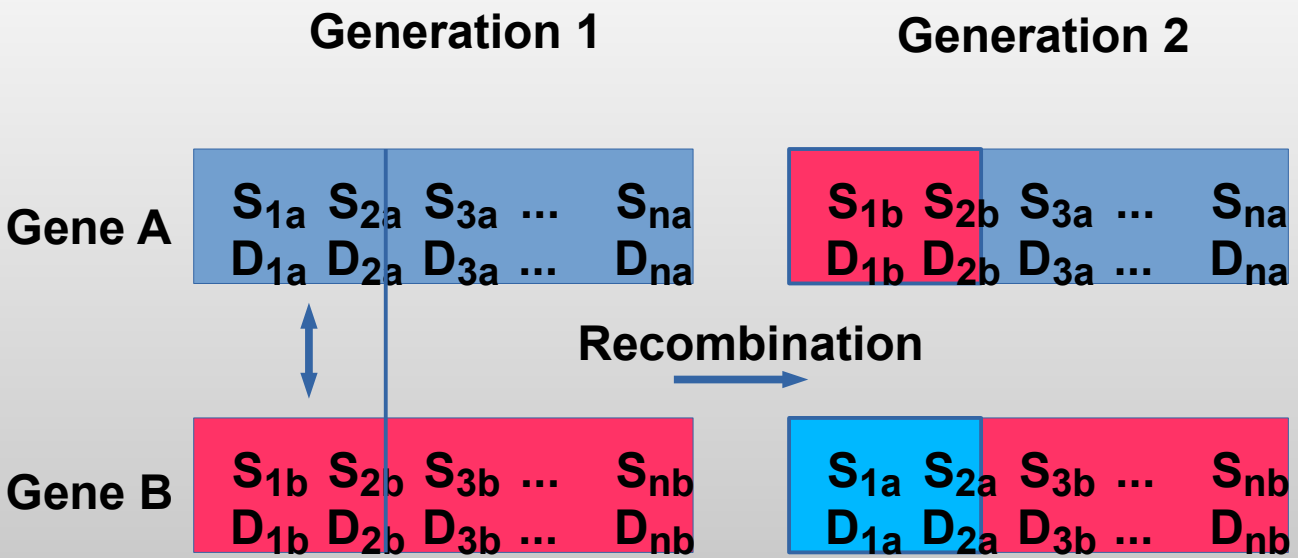
- **Evolutionary paradigm - mutation, recombination, deletion, insertion, and crossover operators are used for adjusting parameters**
- **Random number generators are used to manipulate operators**
- **Generational Model – survival of the fittest (...fitting function)**
- **Generation → iterations, genes → parameter strings, bases → s, D**

GA genes:

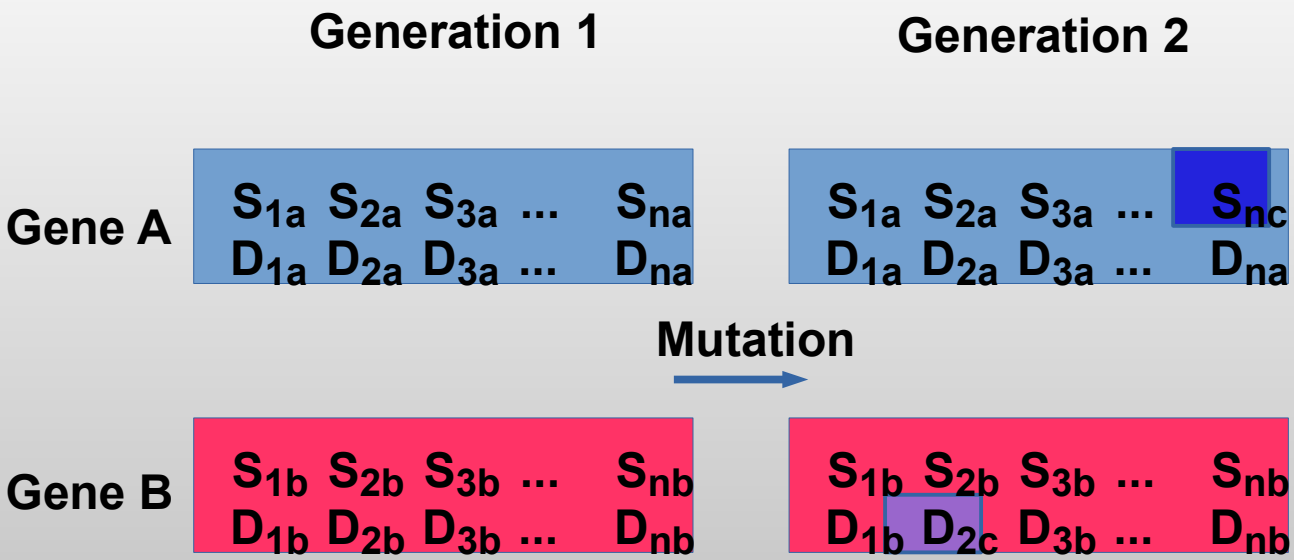
Genes are strings of parameters defining all components in the finite element solution, each component is represented by a pair of corresponding sedimentation and diffusion coefficients.



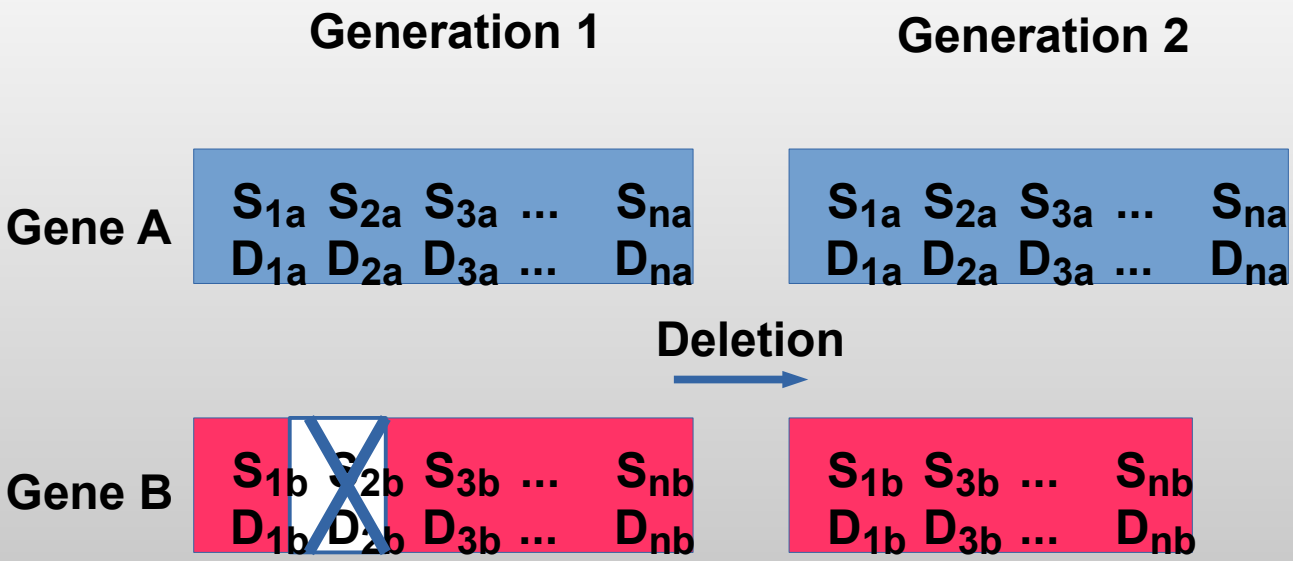
Crossover/Recombination



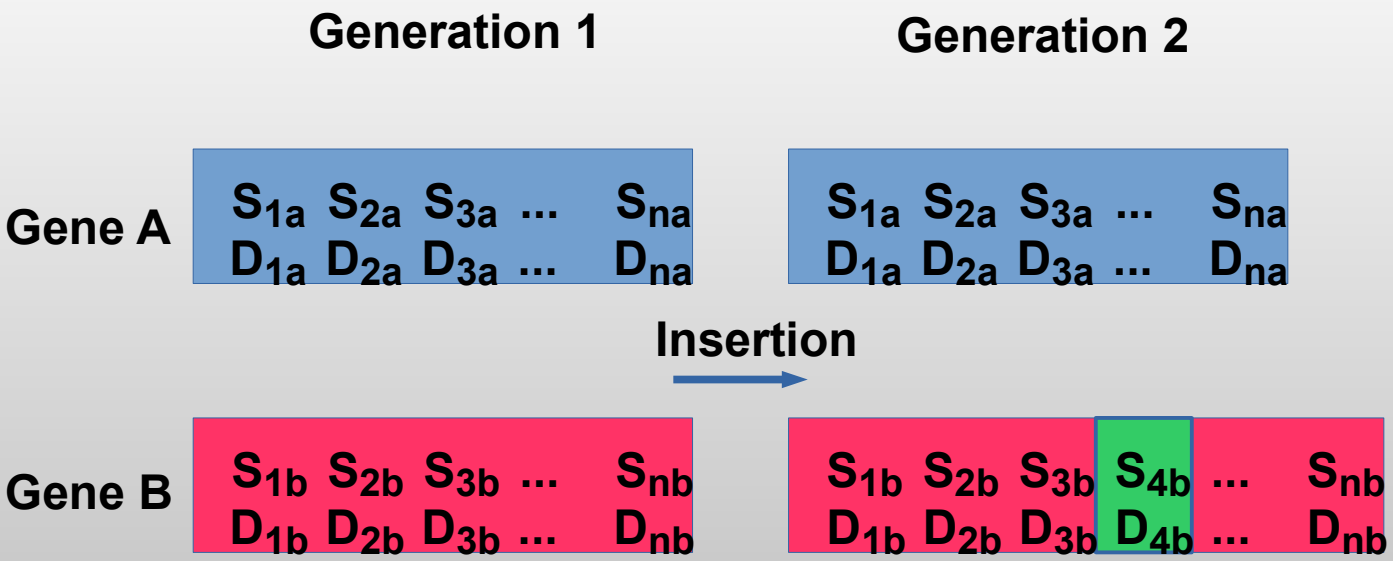
Mutation


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
S_{1a} S_{2a} S_{3a} ... S_{na}
D_{1a} D_{2a} D_{3a} ... D_{na}
S_{1a} S_{2a} S_{3a} ... S_{na}
D_{1a} D_{2a} D_{3a} ... D_{na}
Deletion
Gene B
S_{1b} S_{2b} S_{3b} ... S_{nb}
D_{1b} D_{2b} D_{3b} ... D_{nb}
S_{1b} S_{3b} ... S_{nb}
D_{1b} D_{3b} ... D_{nb}

Insertion


Generation 1
Generation 2
Gene A
S_{1a} S_{2a} S_{3a} ... S_{na}
D_{1a} D_{2a} D_{3a} ... D_{na}
S_{1a} S_{2a} S_{3a} ... S_{na}
D_{1a} D_{2a} D_{3a} ... D_{na}
Insertion
Gene B
S_{1b} S_{2b} S_{3b} ... S_{nb}
D_{1b} D_{2b} D_{3b} ... D_{nb}
S_{1b} S_{2b} S_{3b} S_{4b} ... S_{nb}
D_{1b} D_{2b} D_{3b} D_{4b} ... D_{nb}

Parsimonious Regularization

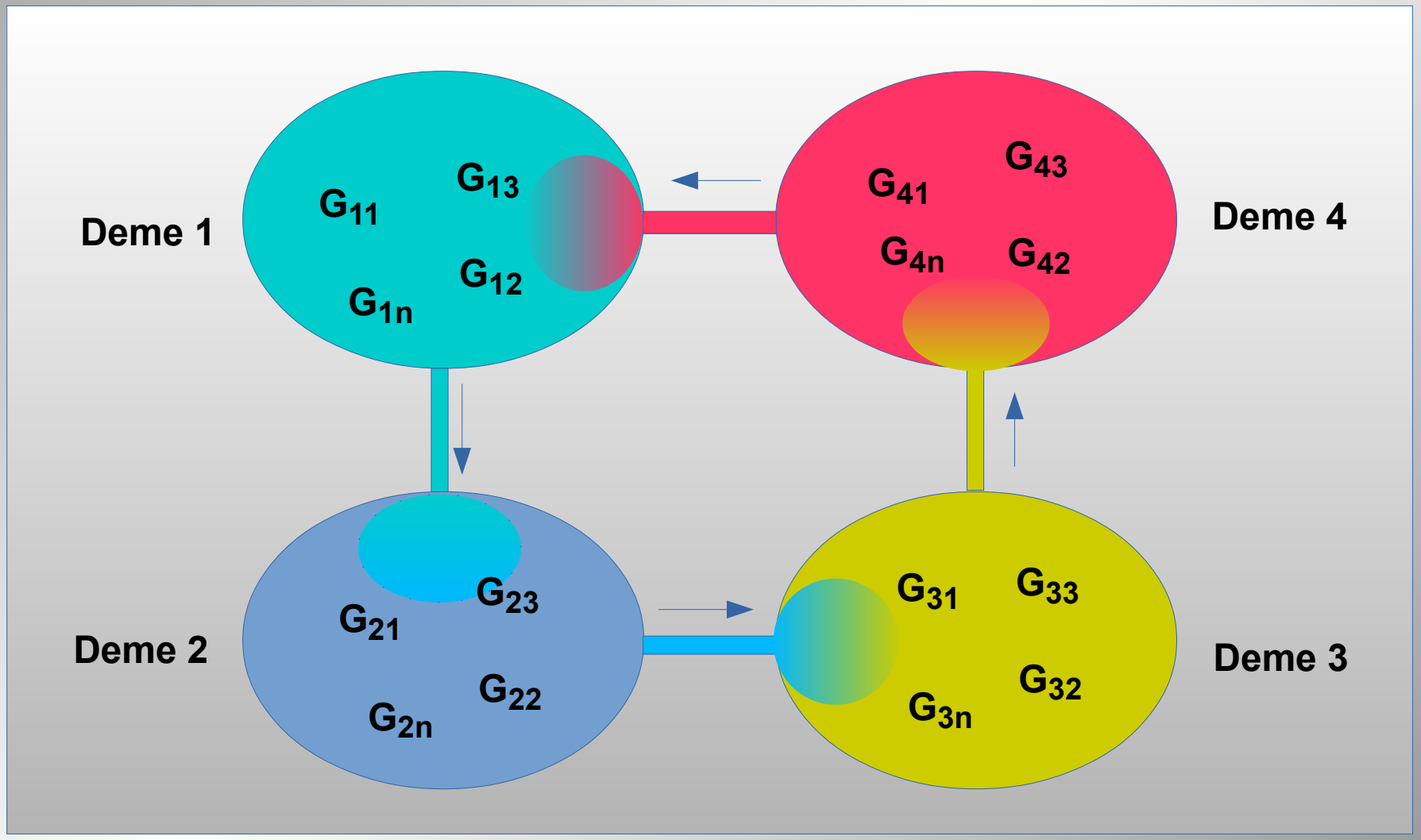
When fitting noisy data with least squares, the fit with the optimal RMSD does NOT represent a unique solution.

According to Occam's Razor, the simplest solution with an RMSD close to the best RMSD solution (which may be more complex) is to be preferred.

In the optimization process, parsimony in the parameterization is favored by applying a penalty (a) to the fitness value (j = number of components, or basis functions, m is the model and d is the data)

$$min \sum_{i=0}^{n} \left(m_i - d_i \right)^2 + aj$$

Deme Topology



Results:

Genetic Algorithms...

...have excellent convergence properties

...find a *parsimonious* solution and provide good regularization

...resolve more solutes reliably than any other method

...solve the general Lamm equation optimization problem

...are robust and can be used to solve highly nonlinear problems

…can be implemented on a parallel supercomputer

…work well for fitting large, global systems

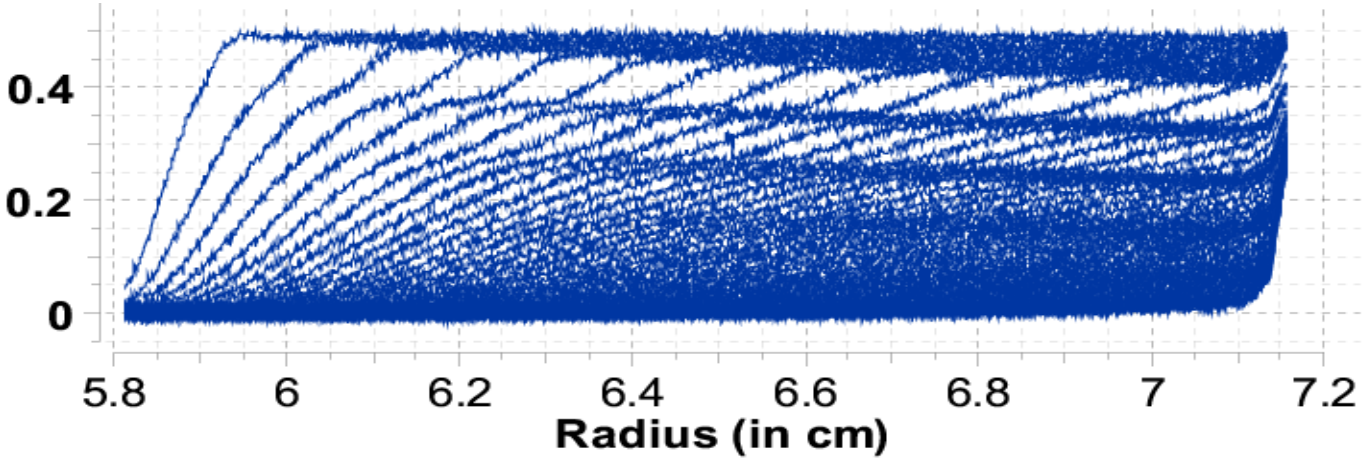
Brookes, E. and B. Demeler. Genetic Algorithm Optimization for obtaining accurate Molecular Weight Distributions from Sedimentation Velocity Experiments. Analytical Ultracentrifugation VIII, Progr. Colloid Polym. Sci.131:78-82. C. Wandrey and H. Cölfen, Eds. Springer (2006)

Brookes, E and B. Demeler. Parsimonious Regularization using Genetic Algorithms Applied to the Analysis of Analytical Ultracentrifugation Experiments. GECCO Proceedings ACM 978-1-59593-697-4/07/0007 (2007)

2-D Spectrum Analysis refinement - Example:

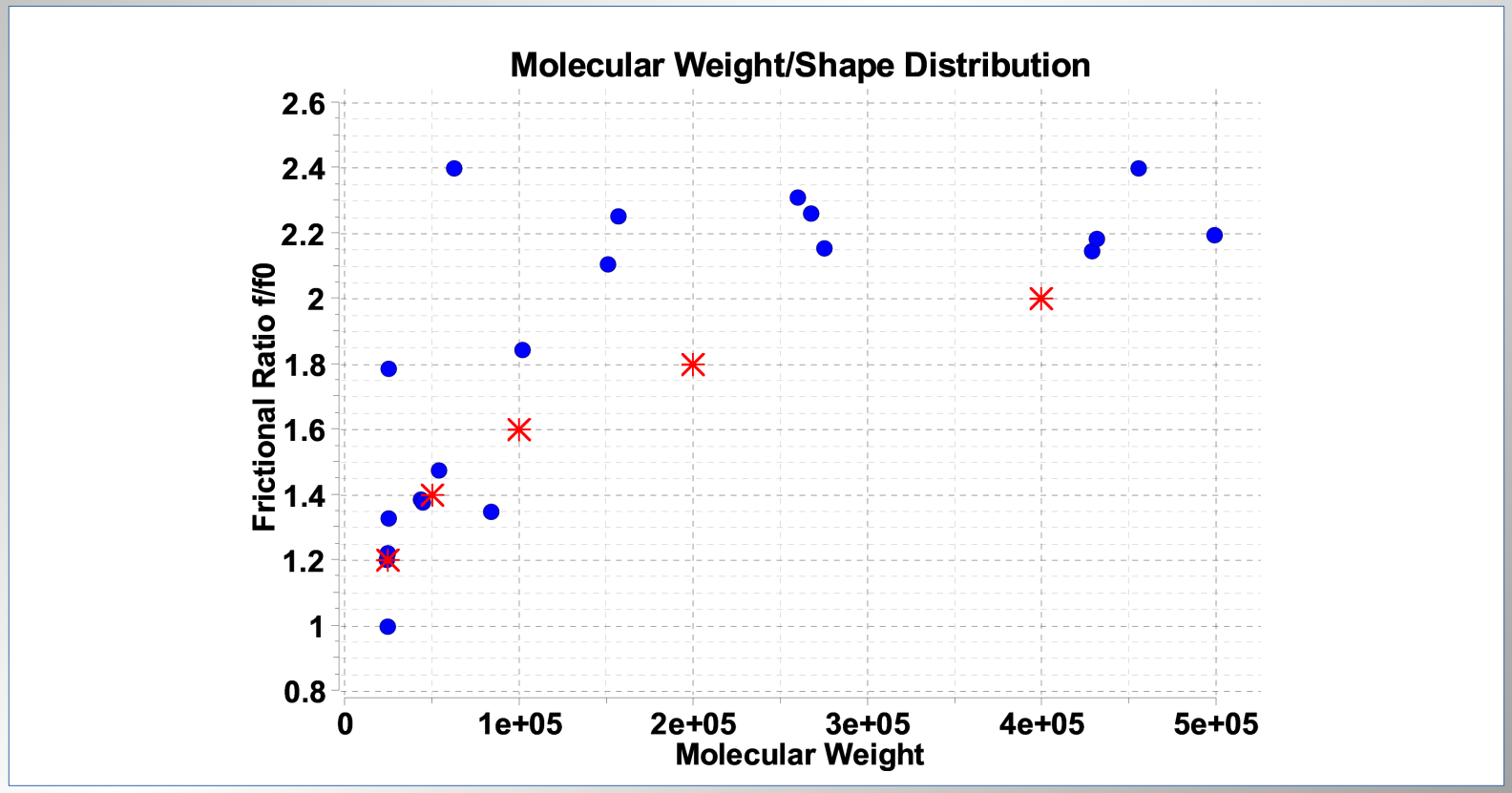
Simulate a 5-component system with heterogeneity in shape and mass
Add stochastic noise equivalent to XLA

MW kD	*s*	*D*	*f/f0*	conc.
25	2.676E-13	9.2736E-7	1.2	0.1
50	3.641E-13	6.3089E-7	1.4	0.1
100	5.058E-13	4.3821E-7	1.6	0.1
200	7.136E-13	3.0912E-7	1.8	0.1
400	1.020E-12	2.2092E-7	2	0.1



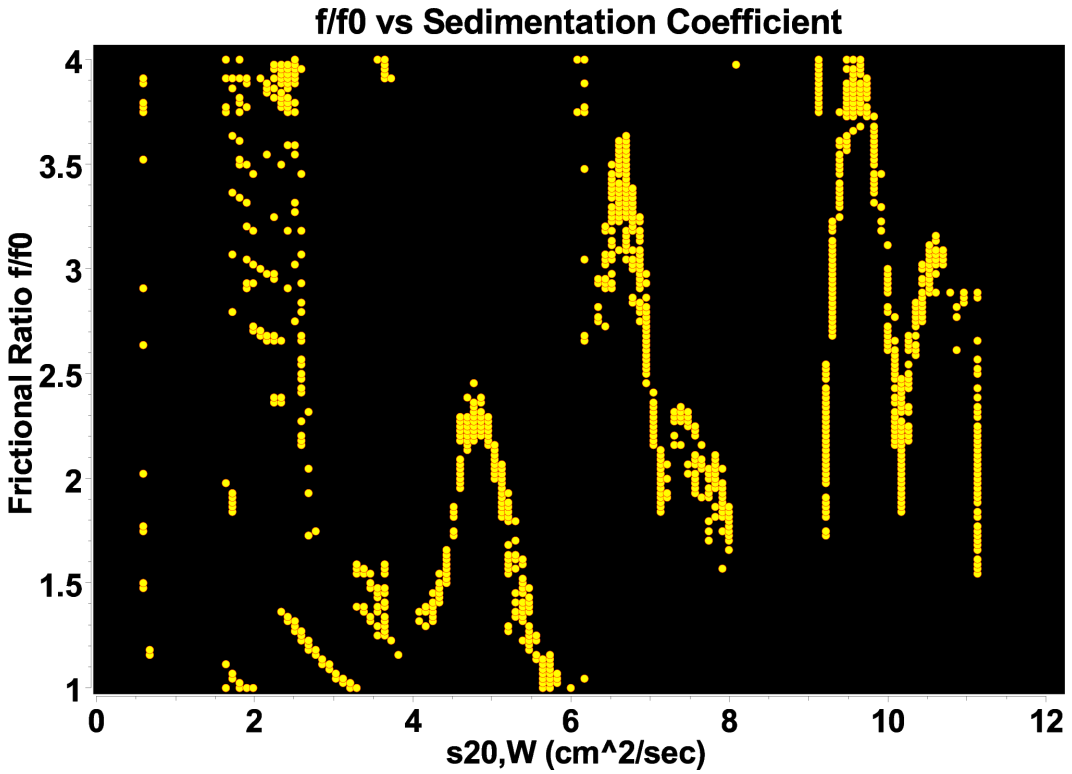
2-D Spectrum Analysis refinement - Example:

Final result is not parsimonious – doesn't satisfy Occam's razor
Solution is over-determined
Noise contributes to false positives



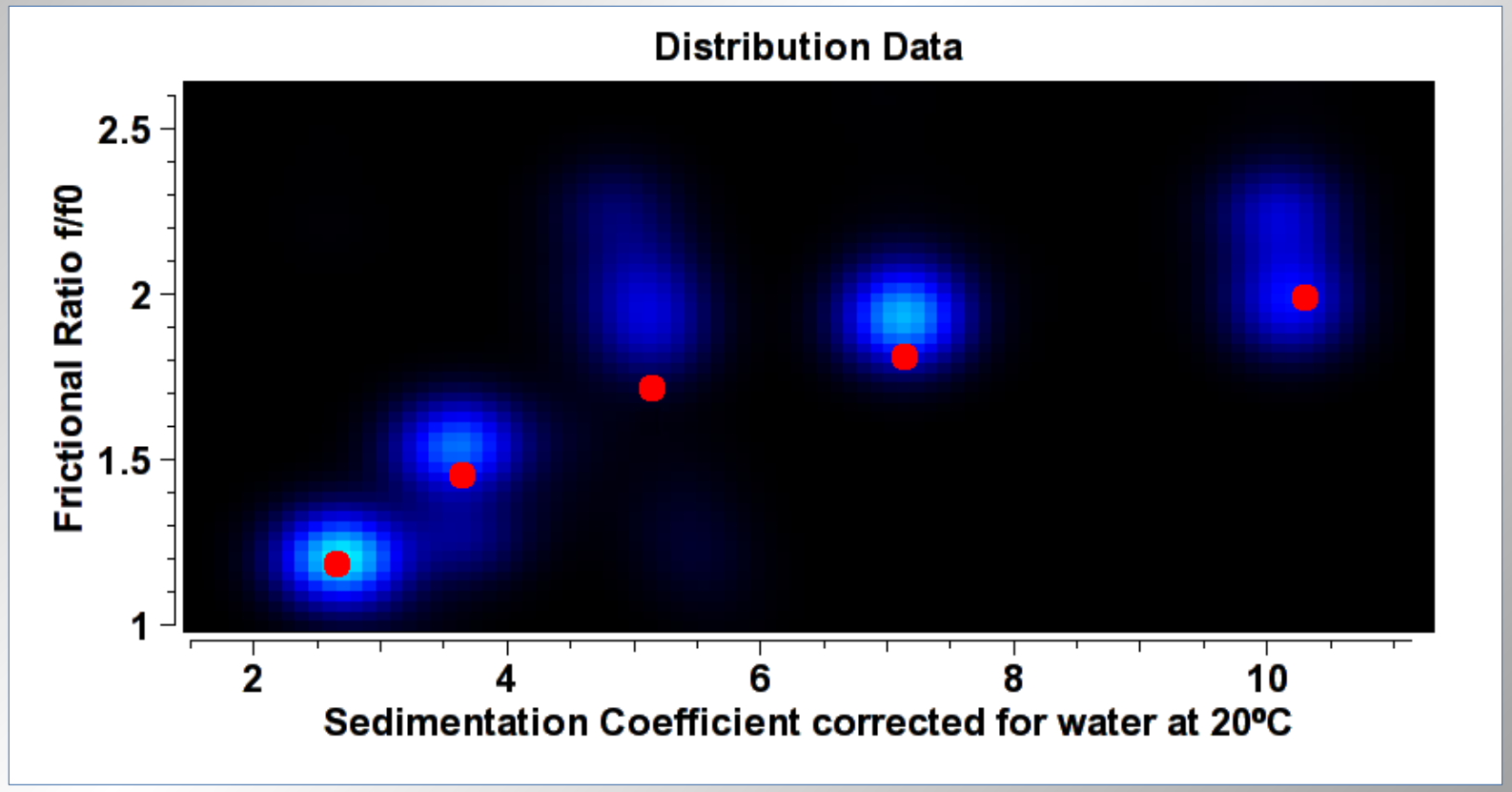
2-D Spectrum Analysis refinement - Example:

Perform 2DSA Monte Carlo analysis to amplify signal linearly
Stochastic noise only amplifies with $\sqrt{2}$



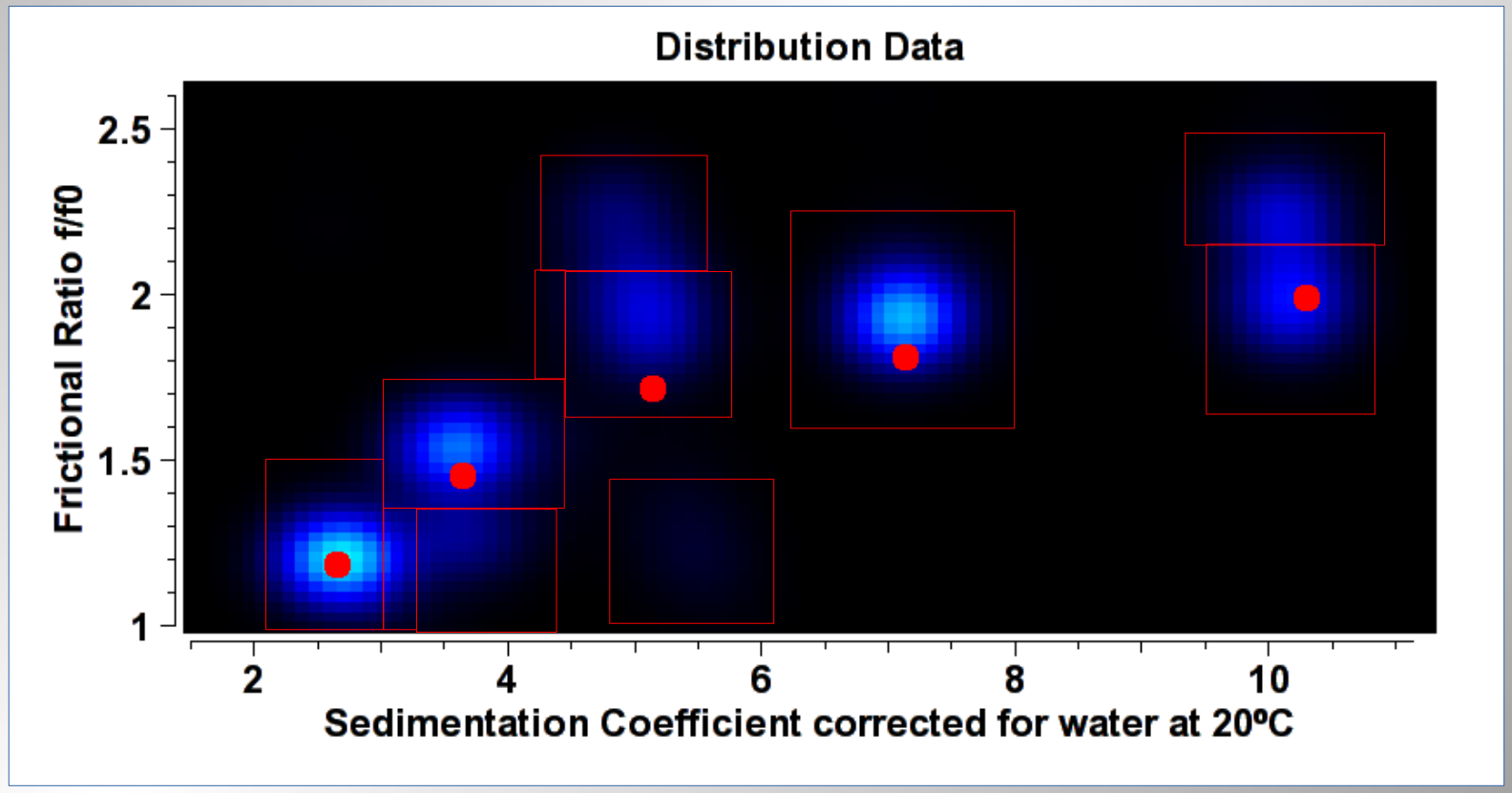
2-D Spectrum Analysis - Refinement:

Stochastic noise signals disappear when frequency is plotted
Sample signal is amplified



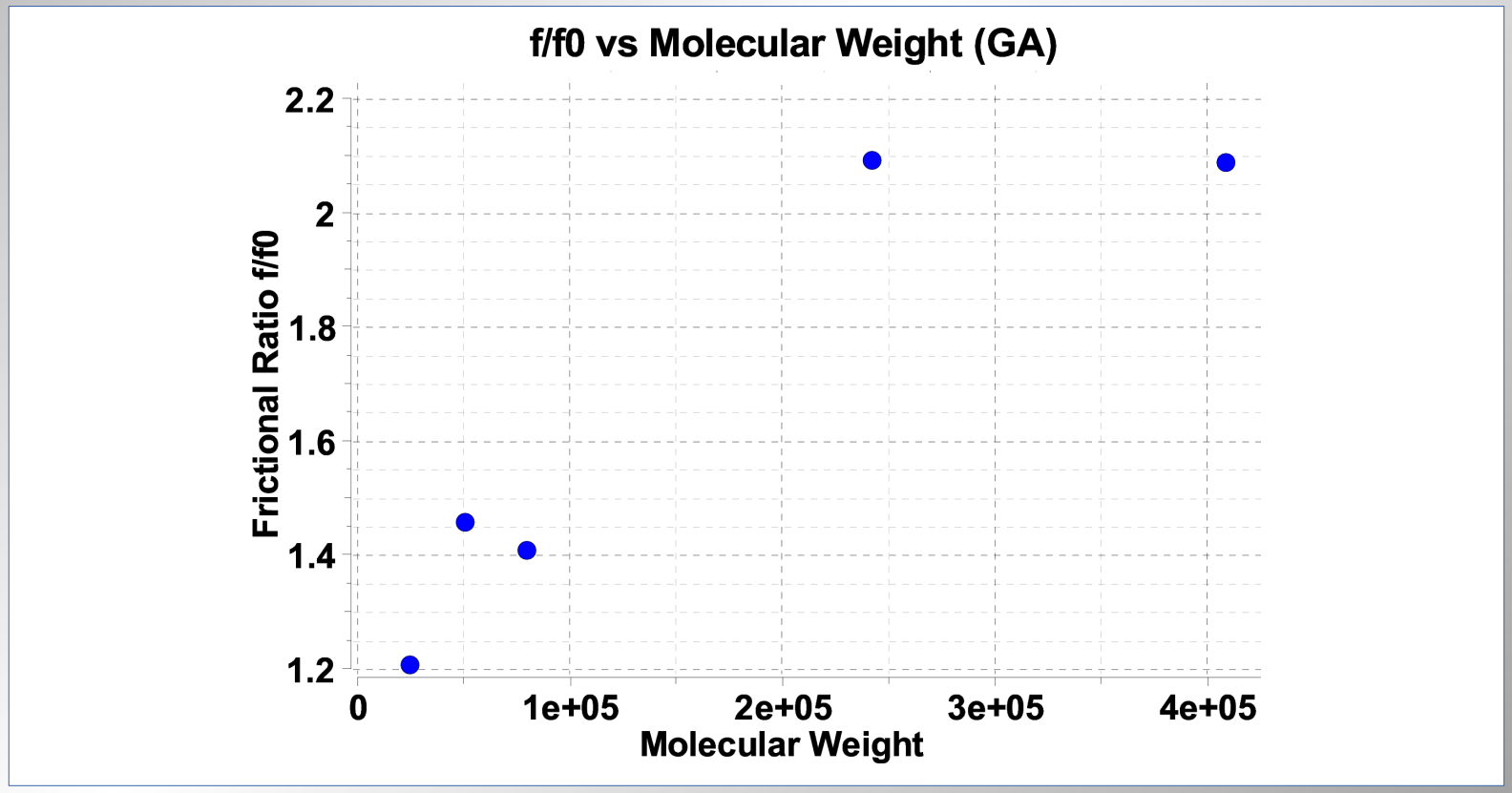
2-D Spectrum Analysis - Refinement:

Identify potential solutes and reduce parameter parameters space using "buckets"
Use buckets to initialize GA analysis



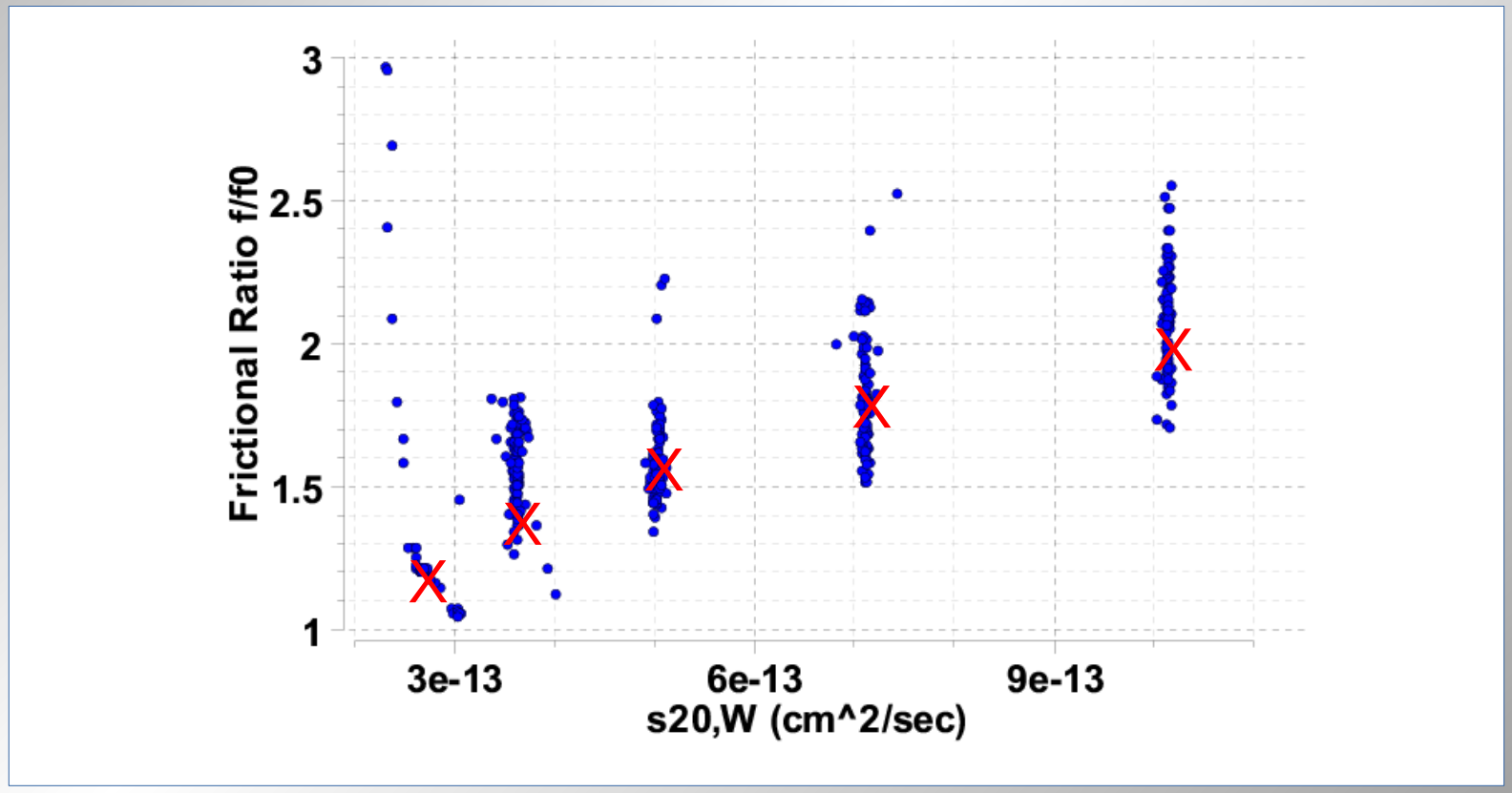
Genetic Algorithm Analysis - Refinement:

Genetic Algorithm produces parsimonious solution
Still affected by stochastic noise



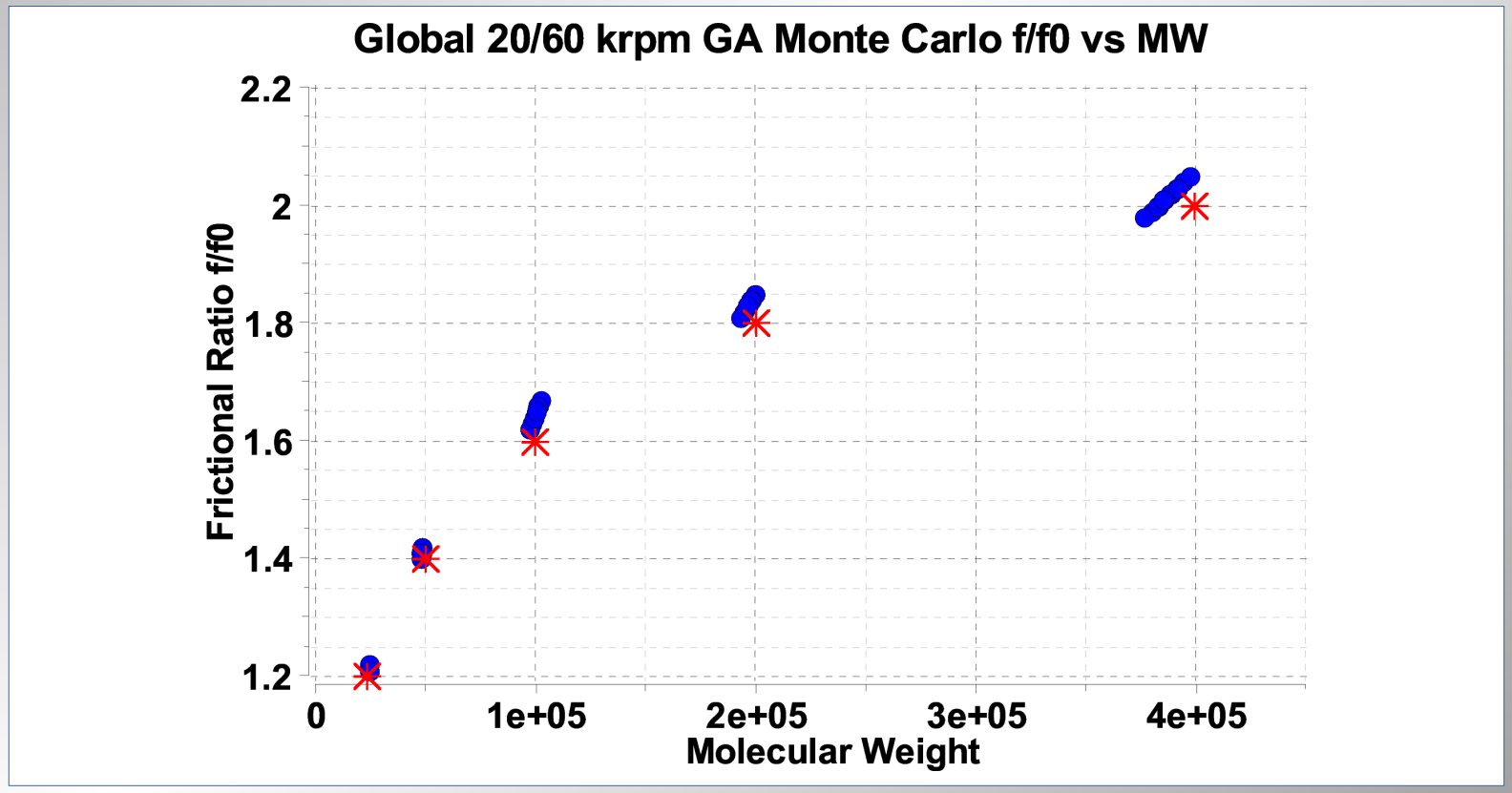
Genetic Algorithm Monte Carlo Analysis

Perform GA Monte Carlo analysis



Global Genetic Algorithm Monte Carlo Analysis

Add low-speed data to emphasize diffusion signal
Perform global GA Monte Carlo analysis



Global 20/60 krpm Monte Carlo Analysis Results

Solute	*Molecular Weight (kD)*	*Partial Concentration*	*Frictional Ratio, f/f_0*
1	24.26 (24.20, 24.33) [25] *	0.0972 (0.0966, 0.0982) [0.1] *	1.21 (1.21, 1.21) [1.2] *
2	48.04 (47.74, 48.46) [50] *	0.102 (0.101, 0.104) [0.1] *	1.41 (1.40, 1.42) [1.4] *
3	100.2 (97.96, 101.8) [100] *	0.0995 (0.0982, 0.101) [0.1] *	1.65 (1.63, 1.67) [1.6] *
4	198.0 (194.2, 200.8) [200] *	0.0996 (0.0989, 0.101) [0.1] *	1.84 (1.82, 1.86) [1.8] *
5	385.3 (380.4, 394.0) [400] *	0.100 (0.100, 0.101) [0.1] *	2.01 (1.99, 2.04) [2.0] *

Monte Carlo Results from a global genetic algorithm optimization using multi-speed data. The results demonstrate remarkable agreement with the original target model. Round brackets: 95% confidence intervals; square brackets: target value. All values rounded off to 3 or 4 significant digits.

* **0% error for 95% confidence interval**

* **< 2% error for 95% confidence interval**

* **< 5% error for 95% confidence interval**

Homework (p1 of 3)

TABLE 4.3 *Diffusion coefficients*[a]

Substance	*Molecular Weight*	$D_{20,w} \times 10^6$ [b]	*Method*[c]
Glycine[d]	75	9.335	G
Sucrose[d]	342	4.586	G
Ribonuclease	13,683	1.068	R
Serum albumin[d] (bovine)	66,500	0.603	R
Tropomyosin	93,000	0.224	S
Fibrinogen[d] (human)	330,000	0.197	R
Myosin[d]	440,000	0.105	S
Tobacco mosaic virus	About 40,000,000	0.053	S
Robbit papilloma virus	About 47,000,000	0.059	S

[a]The diffusion coefficients have been corrected to water at 20°C (see Chapter 5 for the procedure).

[b]Note that D generally decreases with molecular weight but that elongated molecules such as tropomyosin, fibrinogen, myosin, and TMV have unusually low values. The dimensions of D are cm^2/sec.

[c]G, Gouy; R, Rayleigh; S, Schlieren.

[d]Extrapolated to zero concentration.

Homework (p2 of 3)

(A) Given the molecular weights and diffusion coefficients in the table, calculate the following values for Bovine Serum Albumin (vbar = 0.732 ml/g), Ribonuclease (vbar = 0.708 ml/g), Myosin (vbar = 0.731 ml/g), and for a 208 bp double-stranded DNA fragment (vbar = 0.55 ml/g, $D=1.9 \times 10^{-7}$ cm^2/sec, MW=131,000 Da):

1. sedimentation coefficient (10%), 2. frictional ratio (10%), 3. Stokes radius (10%), 4. minimal radius (10%)

5. Which of these molecules is most non-globular? (10%)

6. Using your answer from (3), calculate the amount of water for each protein that would have to be bound to account for the Stokes radius of the molecule in terms of the ratio of grams of water : grams of protein. Is that reasonable? (10%)

Homework (p3 of 3)

(B) Indicate if *s* and *D* increase, decrease or stay the same when the following events occur, and justify your answer:

1. an anisotropic (asymmetric) monomeric protein aggregates into a large globular blob (8 %)

2. a monomeric protein unit elongates through head-to-tail association, forming a fibril shape (8 %)

3. a DNA molecule is dialyzed from a 500 mM NaCl solution into a 5 mM NaCl solution (8 %)

4. a globular, well-folded protein unfolds into a denatured state (8 %)

5. a monomeric, globular protein associates to form a globular hexamer (8 %)

Show ALL work, print legibly!

Homework

a) The anhydrous density of a protein is 1.43 g/ml. When measured in the ultracentrifuge, the partial specific volume was determined to be 0.742 ml/g. How many microliters of water are bound to 1 ml of solvated protein? Assume a water density of 1 g/ml.

b) A researcher studies a 50 kDa protein by sedimentation velocity. At high salt, Genetic algorithm Monte Carlo analysis shows a single species with a mean s-value of 3.5 S and a frictional ratio of about 1.45. As the salt concentration.is decreased, a second species appears with an s-value of about 6.6 S and a frictional ratio of about 1.23. Assume the partial specific volume is constant at all salt concentrations at 0.72 ml/g. Explain a possible model for this observation. What can you conclude about the function of salt with respect to any oligomerization? What can you conclude about the oligomerization state? Draw a model of a molecule that would match these observations (Hint: keep in mind the frictional ratios!)

Homework

c) A researcher has created a 2,500 basepair DNA molecule with 12 nucleosome binding sites and reconstitutes the DNA with histone proteins to create an artificial chromatin molecule. He wants to find out what the effect of adding magnesium is on the reconstituted molecule, and performs velocity experiments in 0, 0.5 and 2 mM $MgCl_2$. He observes a heterogeneous s-value distribution (see next page). To his surprise, an equilibrium experiment when analyzed with an lnC vs. r^2-r0^2 plot revealed a good fit to a straight line. The slope of the linear fit for all salt concentrations was identical and gave a value of 2.092. (assume 20C, 3000 rpm, a viscosity of 0.01poise, a density of 1.0 g/ccm for all salt concentrations, vbar of 0.65 g/ccm)

c1) What do you conclude from the **velocity** results? Explain the different distributions observed and what they indicate. Suggest at least two sample characteristics that could account for the observed **velocity** distributions.

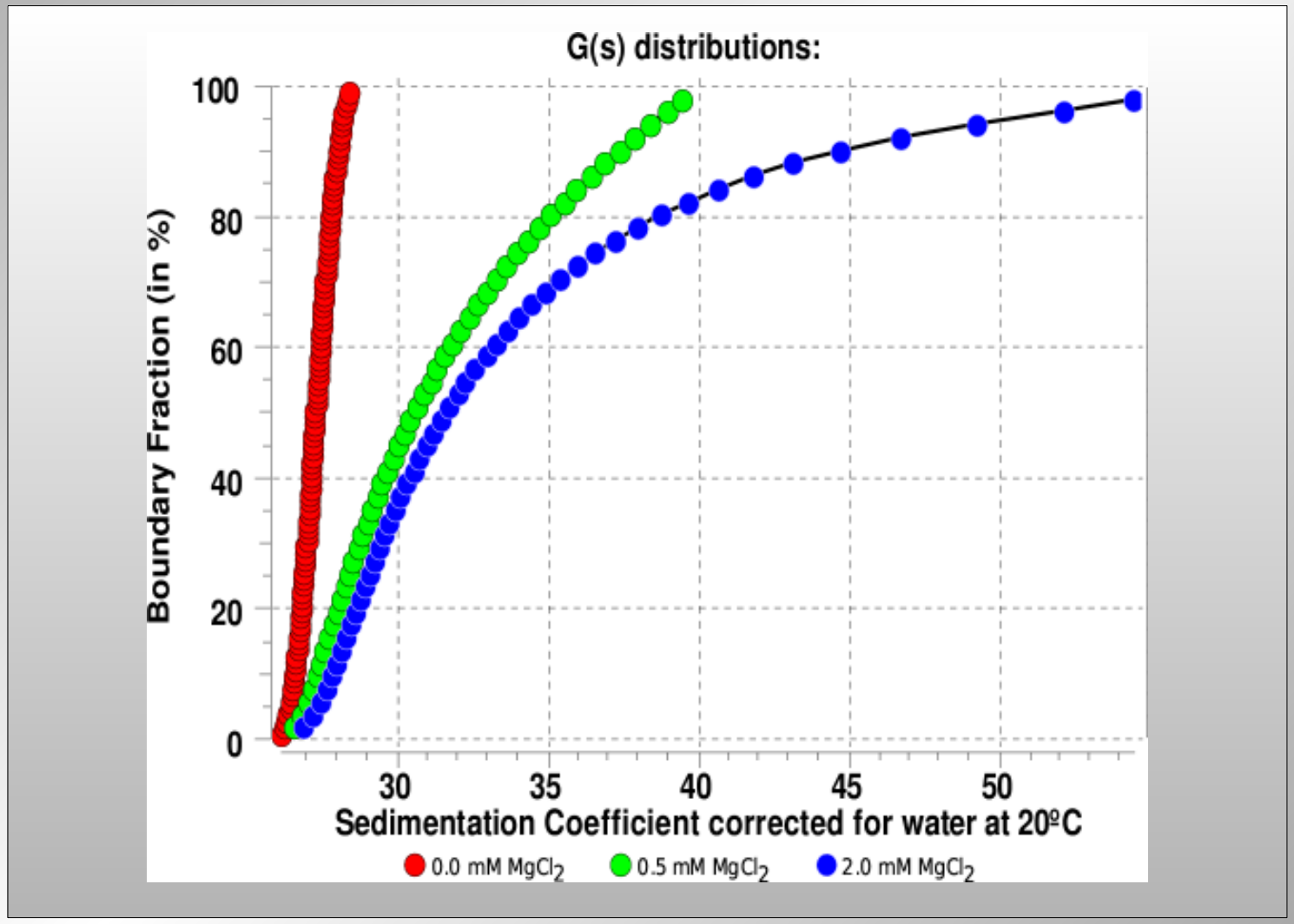
c2) What do you conclude from the **equilibrium** results? Explain.

c3) How can you reconcile the equilibrium results with the velocity results? Explain. Calculate the molecular weight suggested by the slope. What does it say about the chromatin molecule (hint: a histone octamer is about 111.5 kDa, the average DNA molecular weight is ~660 Da/bp.)

c4) Suggest what may have happen to the chromatin molecule as magnesium chloride is added to the buffer. Suggest a model that will account for these results.

c5) calculate the frictional ratio for the 27S and the fastest moving species in each salt conc.

Homework


G(s) distributions:
Boundary Fraction (in %)
100
80
60
40
20
0
30
35
40
45
50
Sedimentation Coefficient corrected for water at 20ºC
0.0 mM MgCl2
0.5 mM MgCl2
2.0 mM MgCl2