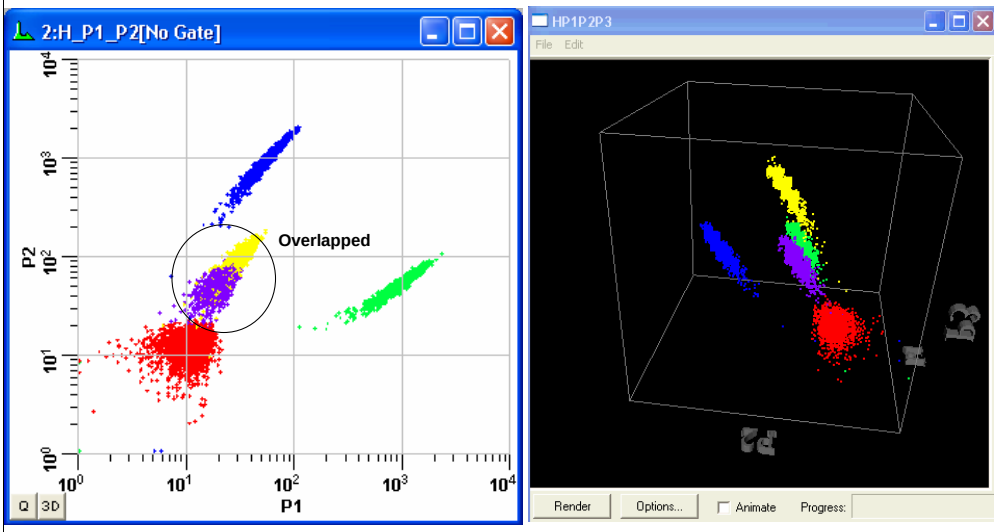
The HyperLog Transformation for Compensated Data

C. Bruce Bagwell MD, Ph.D.
Verity Software House, Inc.

In this lecture I will begin by answering the question of whether we really need to compensate our data or not. I'll then cover some compensation basics. As will be evident in the lecture and demonstrations, the compensation transformation on typical cytometry data often reveals data errors that if not appreciated, can result in operator mistakes in compensating data. Toward the end of the lecture I will discuss new types of display transforms (HyperLog, BiExponential) that properly present compensated data.

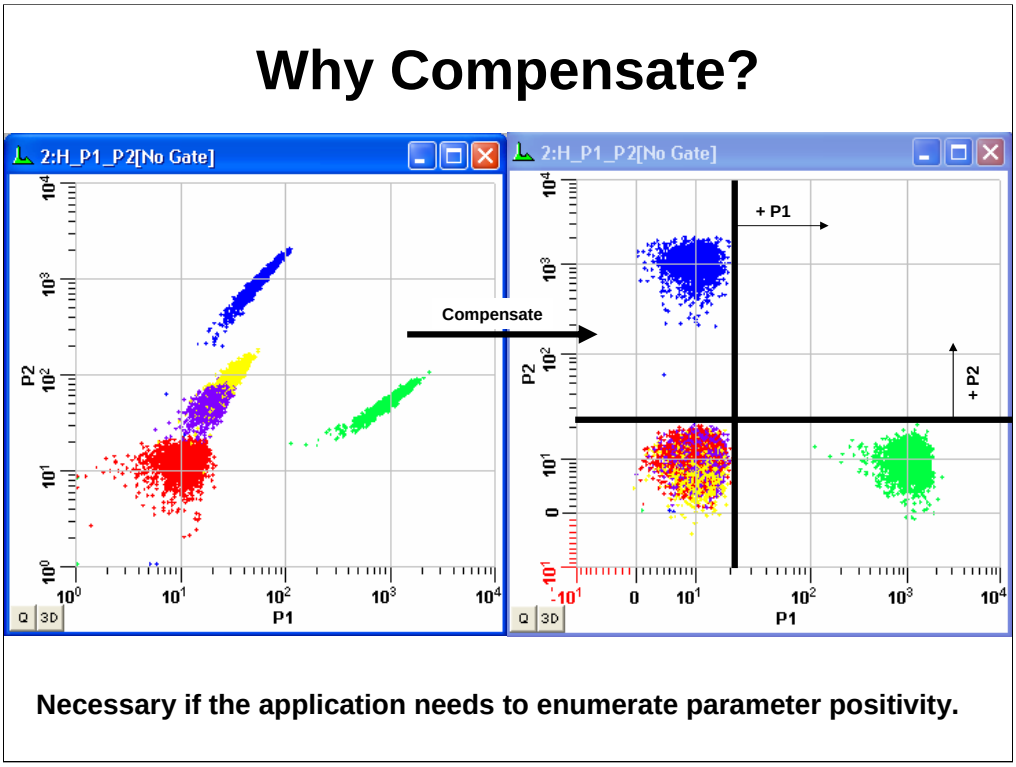
Why Compensate?



Not necessary if the application simply requires population separation.

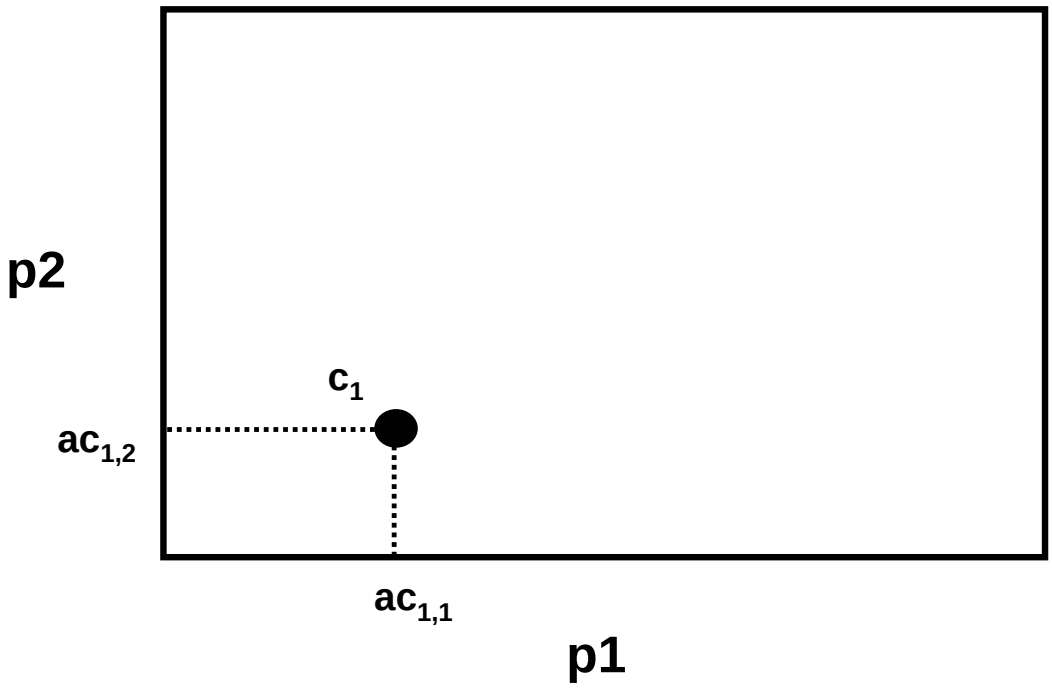
A perfectly logical question to ask is whether it is always necessary to compensate our data for cytometry analysis. The above figure shows some four-color uncompensated data. It doesn't really matter what the parameters are.

The fact that there is some correlation between parameters due to signal crossover does not have any bearing on how separable populations are. The two populations in the above left graph (lavender and yellow) that seem overlapped are easily separated by looking at a different parameter set or adding a new dimension as shown in the animated 3D display. The point is that if you have an application that requires population separation, then compensation is not needed.



On the other hand, if your application requires that you know when a parameter is positive (epitopes, gene products, etc), then you need to compensate to make that evaluation (see right figure).

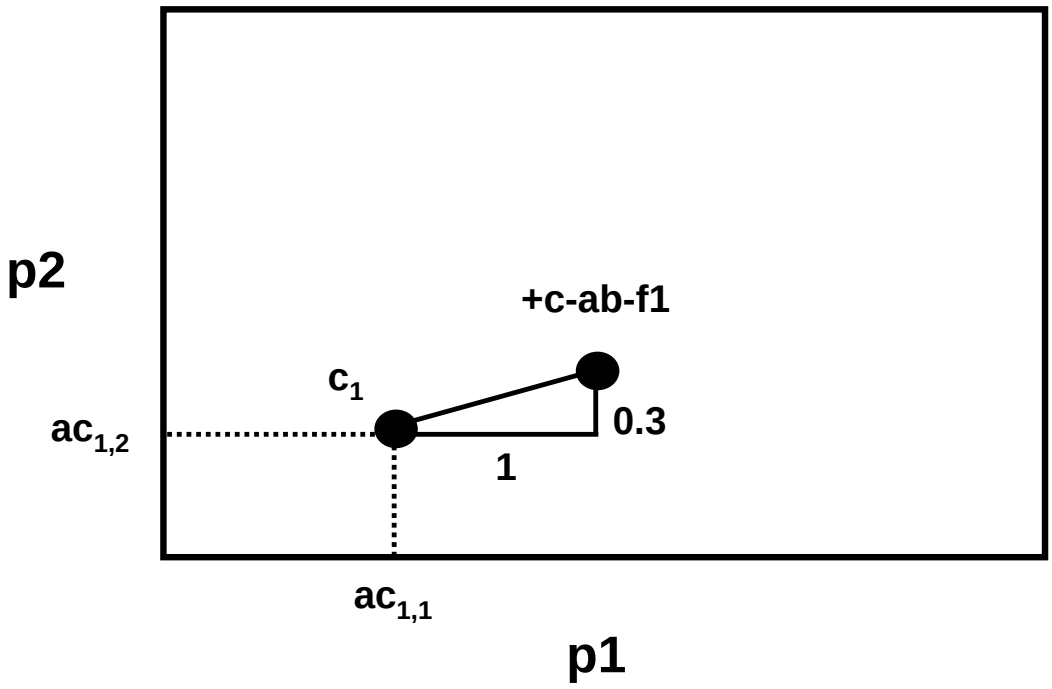
Intrinsic or Autofluorescence



In order to describe how the general process of signal crossover and compensation really works, it's necessary for us to start with a very simple example, adding complexity in a step-wise manner, and developing a rather specific nomenclature as we go. The labels and subscripts were chosen to be just specific enough for our descriptions. It may be helpful for you to substitute these general variables for specific examples. For example, you can think of the cytometer's green detector parameter for p1 and the red detector for p2. For cell c1, think about a single CD4 cell.

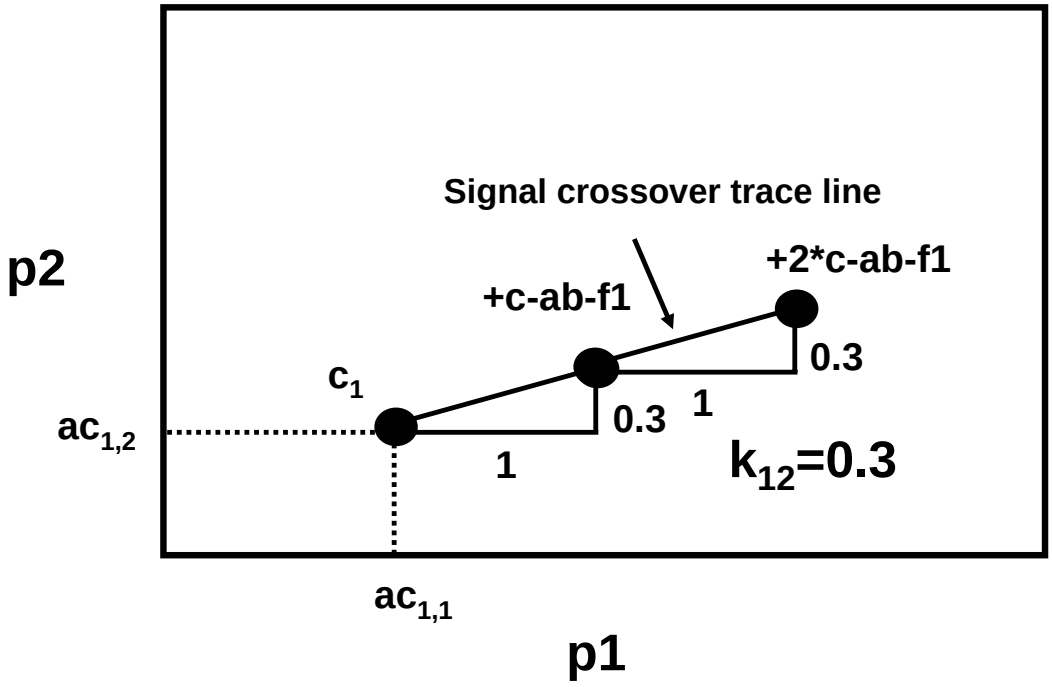
Imagine a single unstained cell, c1, is acquired by a flow cytometer. If we look at two linear fluorescence parameters, p1 and p2, the cell appears as a point located at $\{ac_{1,1},ac_{1,2}\}$. The first subscript in $ac_{1,2}$, for example, identifies the cell, c1, and the second, the detector, p2. The offsets from the origin represent cell c1's intrinsic or auto fluorescence intensity for p1 and p2 parameters.

Signal Crossover



Suppose we attach some antibody, ab, labeled with fluor, f1, to cell of type c. We'll refer to the antibody labeled with f1 that attaches to c as c-ab-f1. The fluor, f1, is predominately detected by the p1 detector. Let's also suppose that for every unit of f1 fluorescence, 30% of this signal crosses over into the p2 detector. As shown above, the cell with the ab-f1 attached, now appears one fluorescence unit to the right and 0.3 units up from the same type of cell with no ab-f1 attached. We are assuming in this hypothetical example that the two axes have identical sensitivities and amplification scales.

Signal Crossover Trace Line

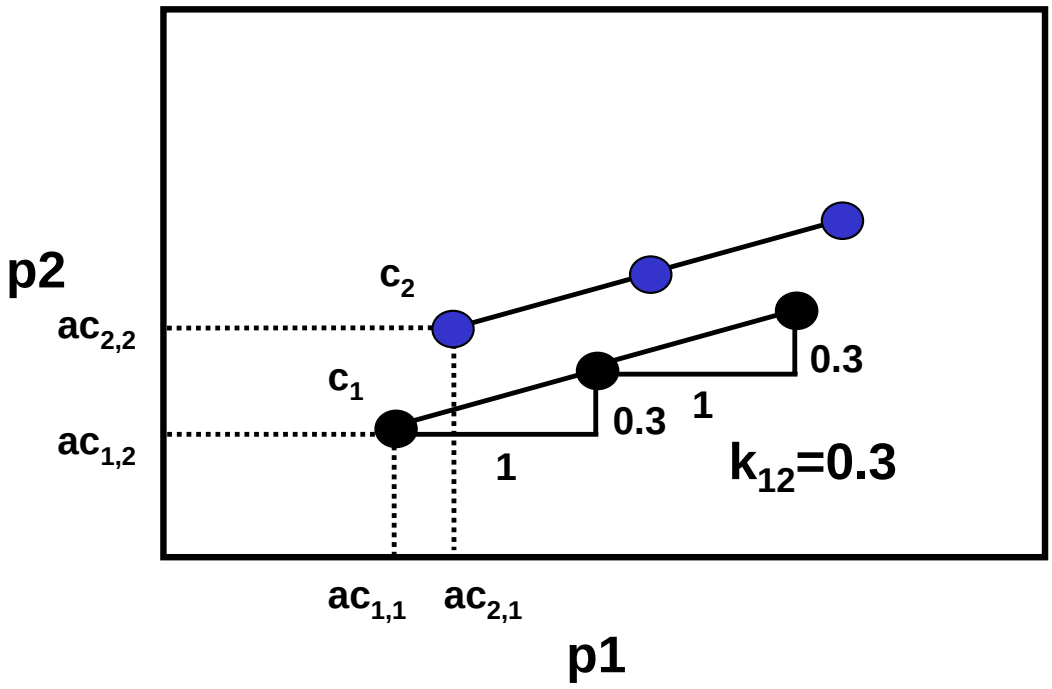


Assumes 1) p1 and p2 are linear parameters over the dynamic range of c-ab-f1 staining and
2) no energy transfer, quenching or other c-ab-f1 concentration effects on or in the cell.

If another c1 cell attaches to two c-ab-f1 molecules, it will be located two fluorescence units to the right and 0.6 units up from the unstained c1 cell. Continuing this process for all possible amounts of c-ab-f1 binding to c1 will theoretically result in a continuous line that starts at $\{ac_{1,1},ac_{1,2}\}$ with slope, k_{12}=0.3. We will refer to this line as the signal crossover trace line.

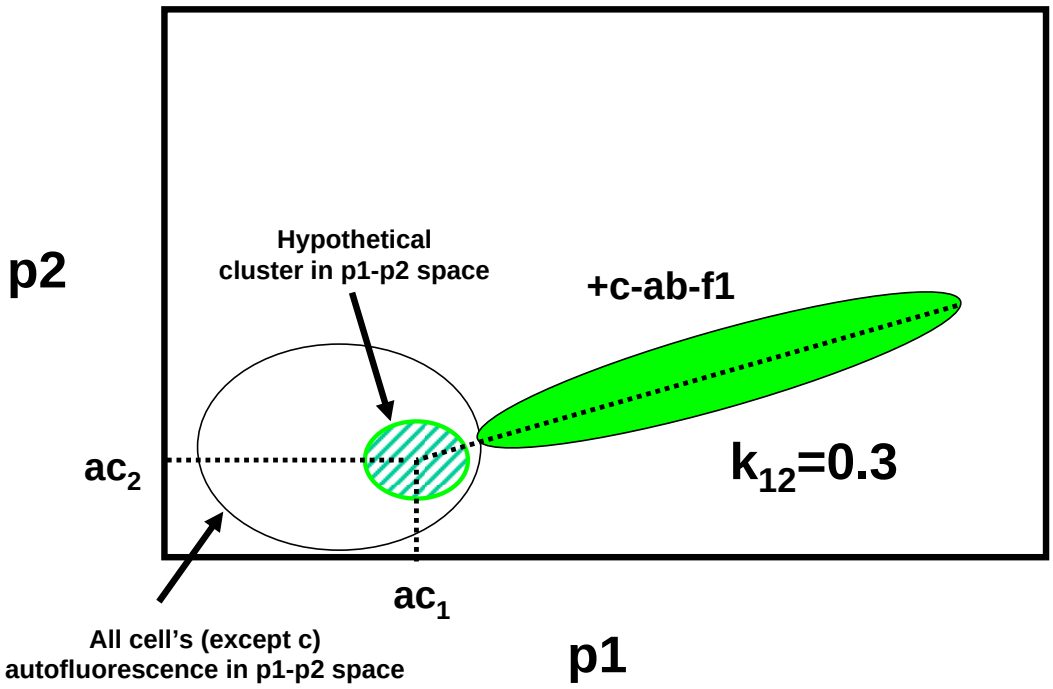
It is important to realize at this point that the signal crossover trace will only be a line if 1) p1 and p2 are linear parameters over the dynamic range of c-ab-f1 staining and 2) there are no energy transfer, quenching or other c-ab-f1 concentration effects on or in the cell.

Two Cell Trace Lines



If we choose another type c cell (c2, blue) with a different intrinsic unstained fluorescence and perform the same experiment with levels of c-ab-f1 staining, we end up with another signal crossover trace line, parallel to the first and originating from the point $\{ac_{2,1},ac_{2,2}\}$.

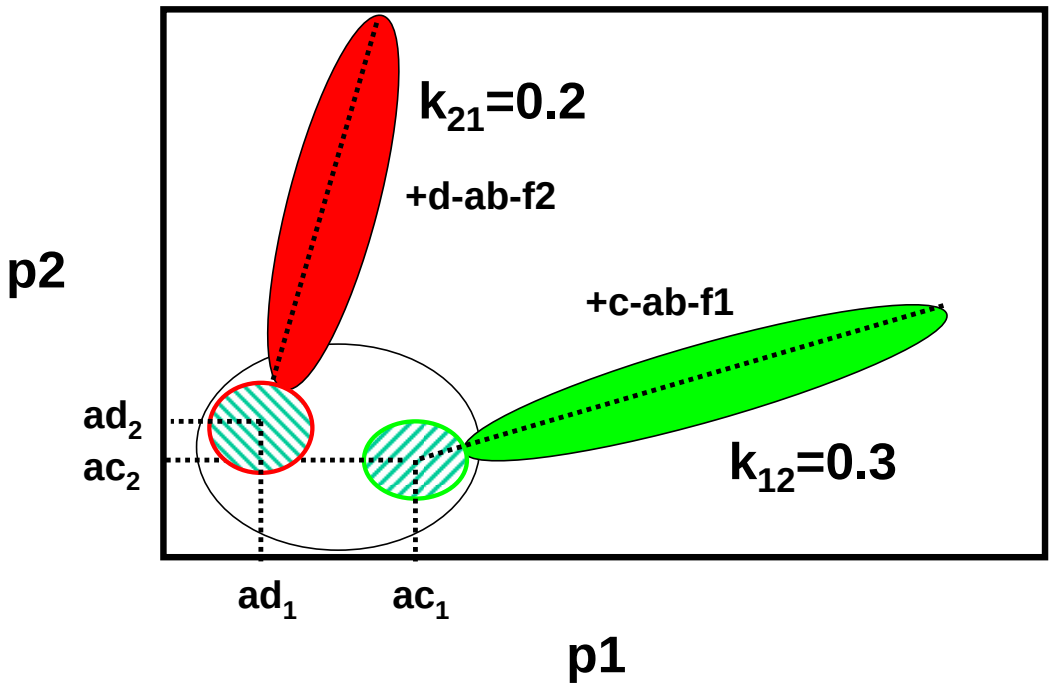
Cell Population Trace Line



Suppose that we performed these measurements on a collection or population of cells, c, that bind different amounts of c-ab-f1. The collection of c cells with no c-ab-f1 becomes a cluster located at $\{ac_1, ac_2\}$ and the c-ab-f1 stained population becomes a diagonal population whose central axis has slope k_{12}=0.3.

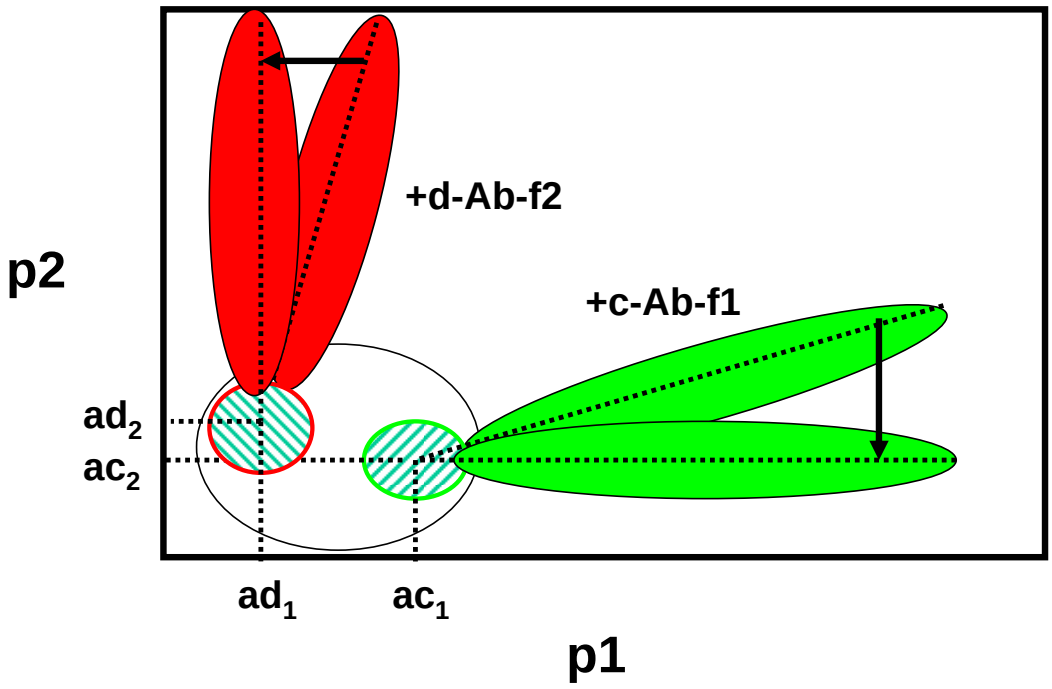
The cluster located at $\{ac_1, ac_2\}$ is a hypothetical cluster since, by definition, in a "c population", all the cells usually stain with some amount of c-ab-f1. In other words, by our trying to measure population c, we cannot measure c's autofluorescence. Another way of looking at this is that before we stained the cells, there was a large population of unstained cells having some defined autofluorescence distribution (see large black circle). After we stain the c cells with c-ab-f1, we eliminate the c cells contribution to this general autofluorescence distribution. We'll come back to this point in a moment when we discuss some common fallacies in performing hardware compensation.

Two Population Trace Lines



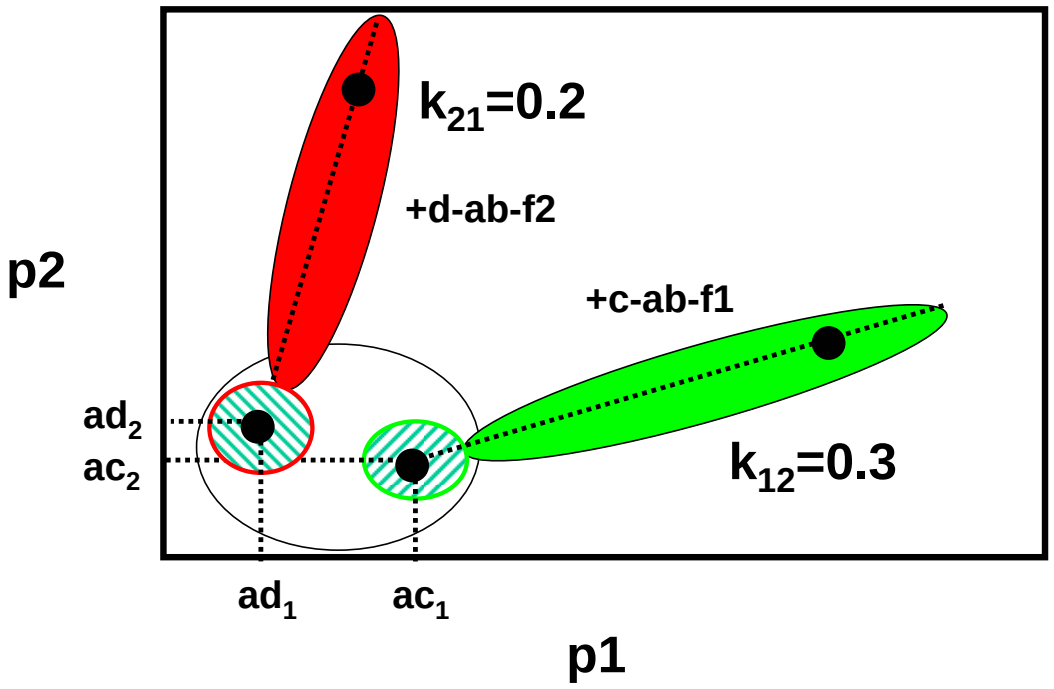
We now stain the cells with a f2 fluor labeled antibody that binds with “d” cells. In this example, 20% of the f2 fluor signal crosses over into the p1 detector. The d population autofluorescence is located at {ad_1, ad_2}. The above figure portrays this more complicated two color situation. Notice that the general autofluorescence population is all the cells that have not reacted to either the c or d antibodies.

Perfect Compensation



If we could perfectly compensate the above data, each population would be rotated about their respective autofluorescence points to an orthogonal position (see above slide). If we knew these autofluorescence positions in space, we could in fact do this bit of mathematical wizardry. Do we know where these points are? Unfortunately, no.

Hardware Compensation



To compensate,

Comp p1 = p1-0.2*p2

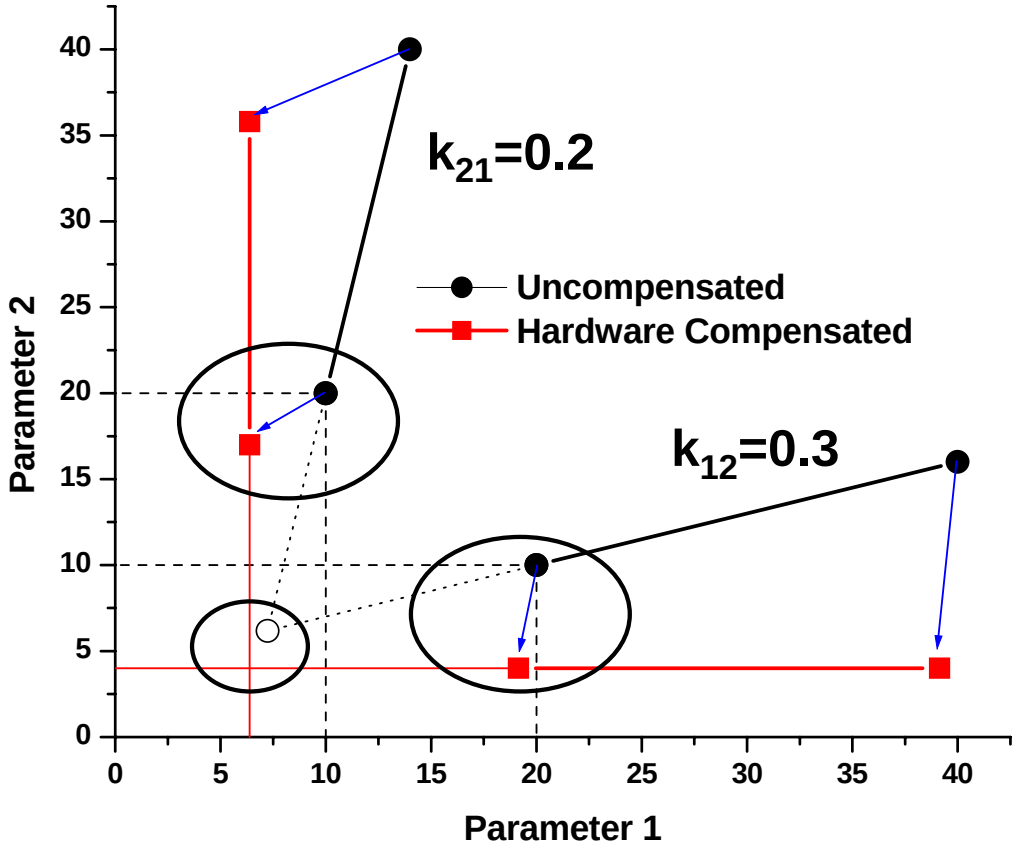
Comp p2 = p2-0.3*p1

To perform hardware compensation, a certain percentage of one signal is subtracted from another. In this particular example, to exactly compensate the above signals we would subtract 20% of p2 from p1 to obtain a compensated p1 signal and 30% of p1 from p2 to compensate p2.

A common mistake is to think that to compensate p1 we have to subtract 30% of p2. The best way of making sure you understand what's really going on is to simplify the system to just include the p1-only (green) data. To compensate this system properly, we must subtract 30% of p1 from p2 (second equation). p1 does not need to be compensated. In a similar manner, the p2-only data is compensated by the first equation, Comp p1 = p1-0.2*p2.

Notice the autofluorescence and high intensity black dots for the green and red only distributions. The next slide will show these points more quantitatively.

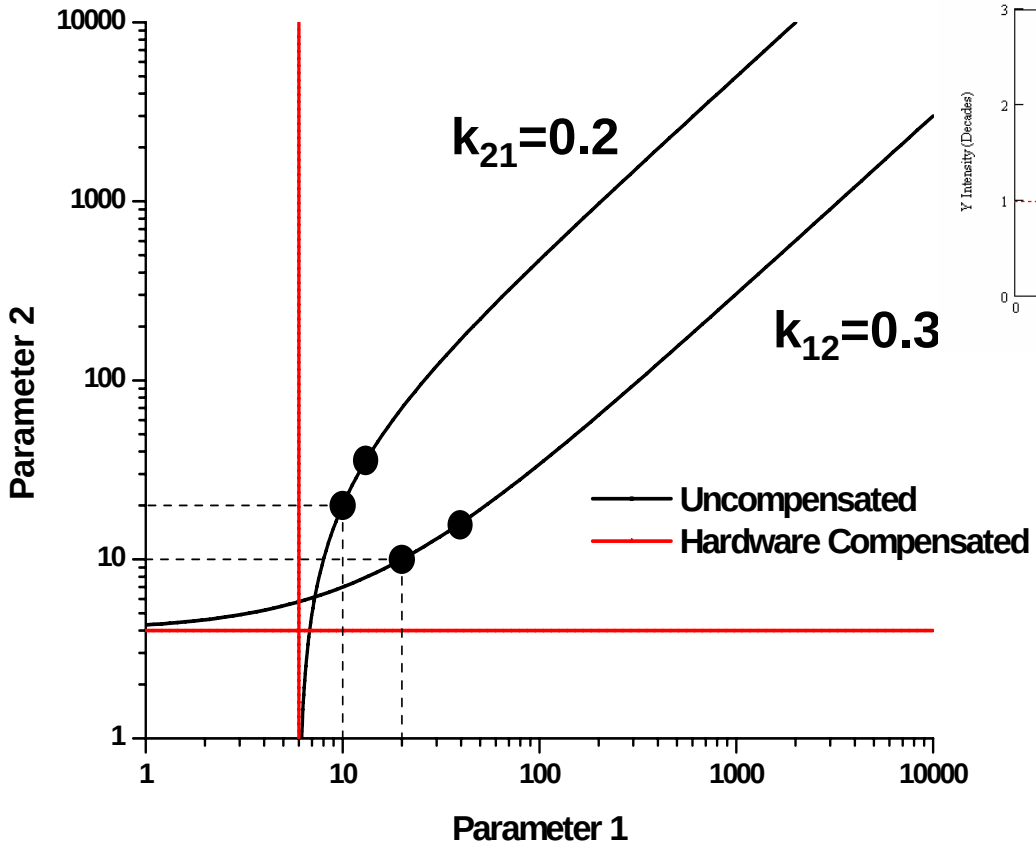
Geometry of Compensation

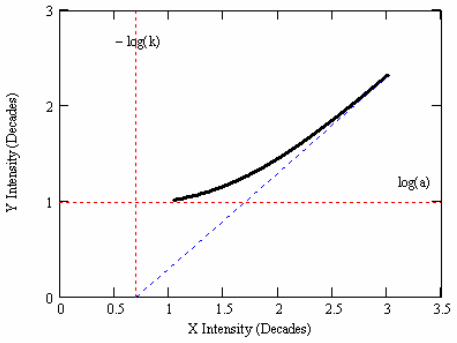


The above figure shows the geometry of our example with and without compensation. The black dots represent the autofluorescence and high uncompensated fluorescence intensities for both single color controls. The red squares are the associated compensated points. Notice that the hardware compensation results in two orthogonal lines, which is the major goal of compensation. Also note that both the autofluorescence points are shifted towards the origin (large circled areas).

A common misconception is that the origins of the compensated and uncompensated trace lines are the same. In other words, the compensation process simply rotates the signal crossover trace lines about a common point in space. In fact, there is no common pivot point if the single color controls have different autofluorescence intensities (see small open circle). The extrapolated lines from the single color trace lines as well as their intersection point have no known importance to cytometrists.

Geometry of Compensation with Log Transformations



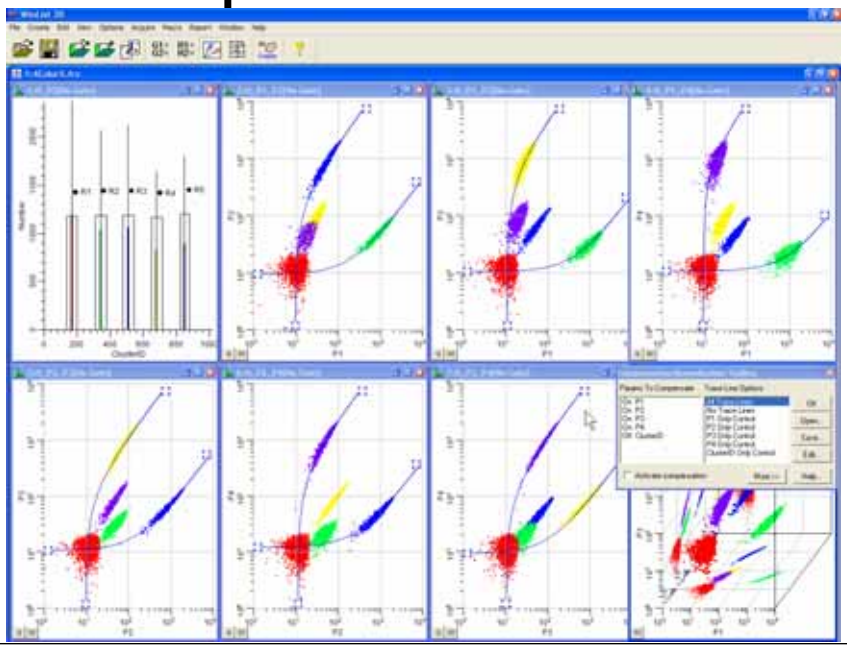


So far, we have only described and viewed the compensation process with linear parameters. What happens when we apply a four-decade log transform to both axes? The above plot is the same example we have been developing, viewed in four-decade space.

Another common misconception is that lines in linear space will appear as lines in logarithmic space. As seen in the above graph, this is generally not the case. The signal crossover traces in log space will only be lines if 1) they are orthogonal to the axes (the compensated red lines) or 2) they have a zero axis intercept, which is usually not the case.

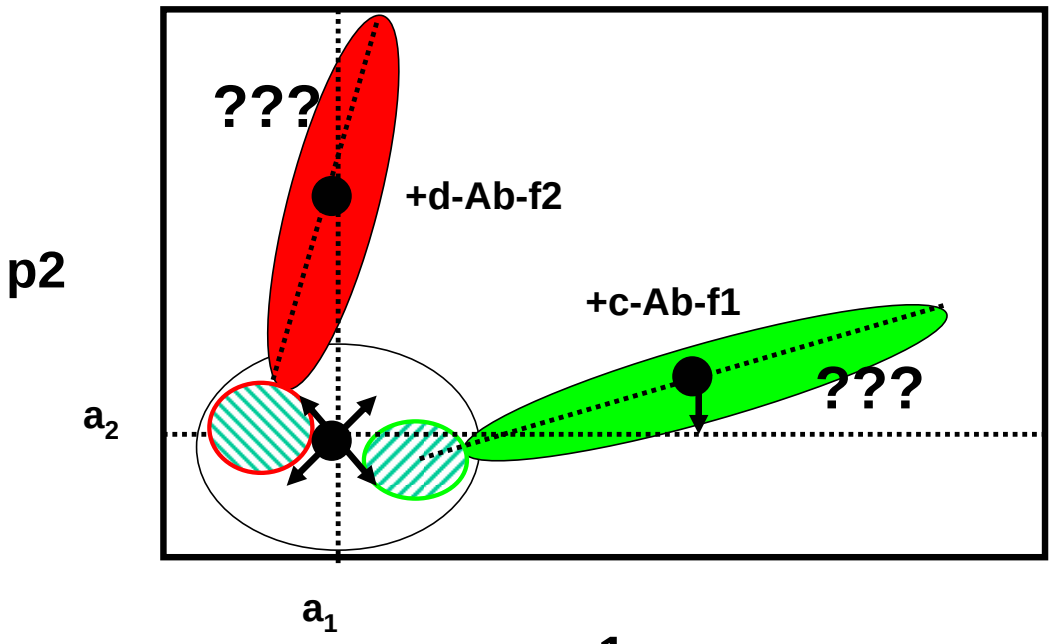
Thus, the signal crossover traces are highly curved in log space, but become orthogonal when proper compensation is applied. Later, we'll describe some other distortions to the data due to the log transform.

Log Transformation and Compensation Trace Lines



With uncompensated cytometry data, the log transformed lines have a characteristic geometry. At high intensity values, they approach 45 degrees and at low intensity values they are generally highly curved, approaching the axis orthogonally. As will be discussed shortly, software compensation systems find the slopes of these lines either automatically or manually and then use them in the compensation mathematics.

Conventional Hardware Compensation



An incorrect approach…

" Find the median position of the autofluorescence population (lower-left black circle) and change the signal subtractors until the medians of the single color populations are the same as the associated median of the autofluorescence population".

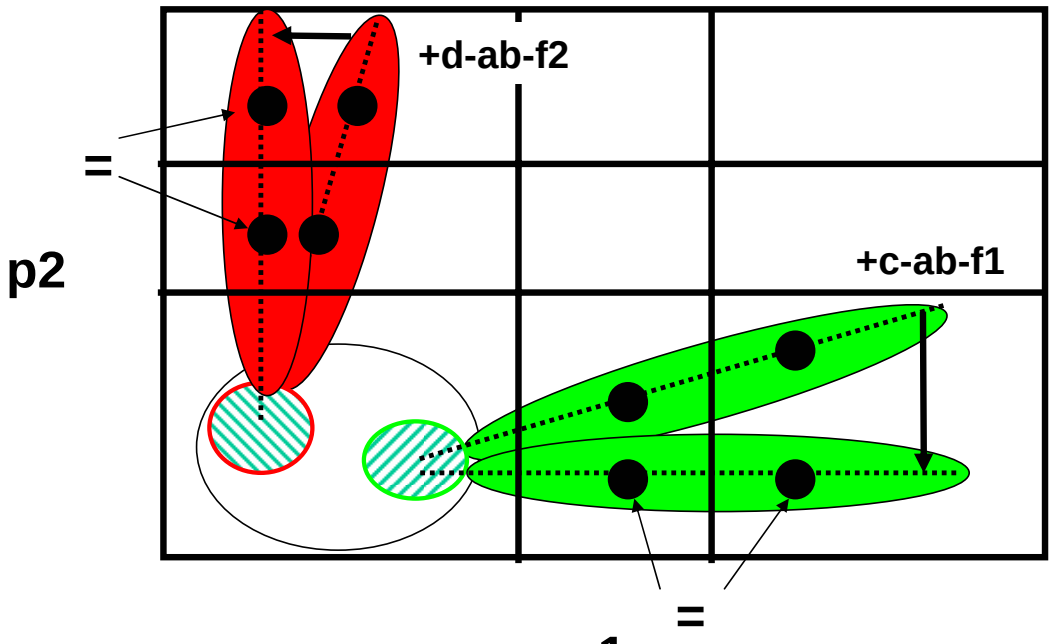
Before examining software compensation in more detail, let's look at the manual hardware compensation approach normally taught to cytometrists, which states, " Find the median position of the autofluorescence population (lower-left black circle) and change the signal subtractors until the medians of the single color populations are the same as the associated median of the autofluorescence population".

As can be seen in the above plot, this algorithm has two major sources of error. First, the center of the autofluorescence population is not the theoretical pivot point for each of the single color trace lines. Even if the two populations stained by c and d antibodies had similar autofluorescence characteristics, the median of the autofluorescence population represents all populations except c and d. As a consequence, depending on where the real pivot points are relative to the center of the autofluorescence population, the system can be over or under compensated after this procedure.

Second, the position of the autofluorescence population is not well defined anyway since it is located in the first decade, which is generally a sea of amplifier noise and log amplifier irregularities. It is usually not a good idea to base a calculation, such as compensation, on such an ill-defined point.

Third, for reasons that will be clear shortly, unless you are very careful on how you calculate these medians, there is a strong tendency to overcompensate due to visual asymmetry to the data error distribution.

High-Low Alternative Approach

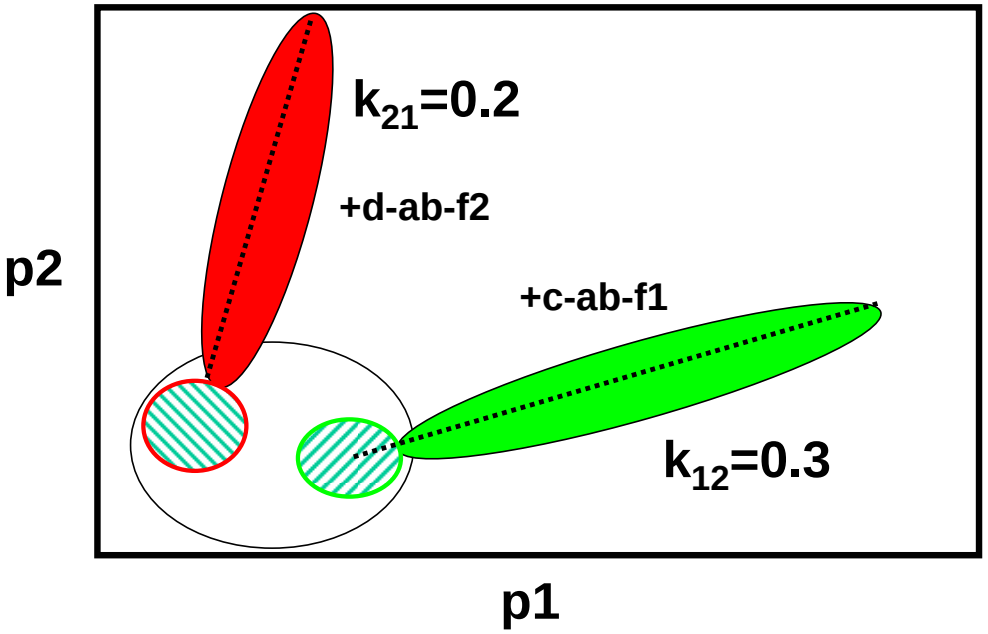


A better approach…

Adjust the signal subtracters such that the low and high medians for both single color populations are identical.

A better approach to hardware compensation is depicted in the above slide. The “high-low” technique divides the single color populations into relative low and high regions that are used to define corresponding low and high medians (see the two black filled circles on the uncompensated populations). The approach is to adjust the signal subtracters such that the low and high medians for both single color populations are identical. Notice how this approach does not involve the autofluorescence population at all.

Software Compensation



$$k = \begin{bmatrix} 1 & k_{12} \\ k_{21} & 1 \end{bmatrix} \qquad k^{-1} \qquad \text{Comp P} = P*k^{-1}$$

Step 1: Create crossover matrix **Step 2: Invert matrix** **Step 3: Matrix multiply**

The process of software compensation is conceptually very similar to the “high-low” technique. Software programs either use a high-low gating method (e.g. FlowJo) or a robust fitting algorithm (e.g. WinList) to estimate the signal crossover slopes. Once the signal crossover slopes are known, they are put into specific elements of a matrix as shown above. A simple inversion of the matrix creates a new matrix called the compensation matrix that can easily convert the raw data to compensated data by a simple matrix multiply operation.

Software Compensation

Reprinted from **Clinical Flow Cytometry**
Volume 677 of the *Annals of the New York Academy of Sciences*
March 20, 1993

Fluorescence Spectral Overlap Compensation for Any Number of Flow Cytometry Parameters

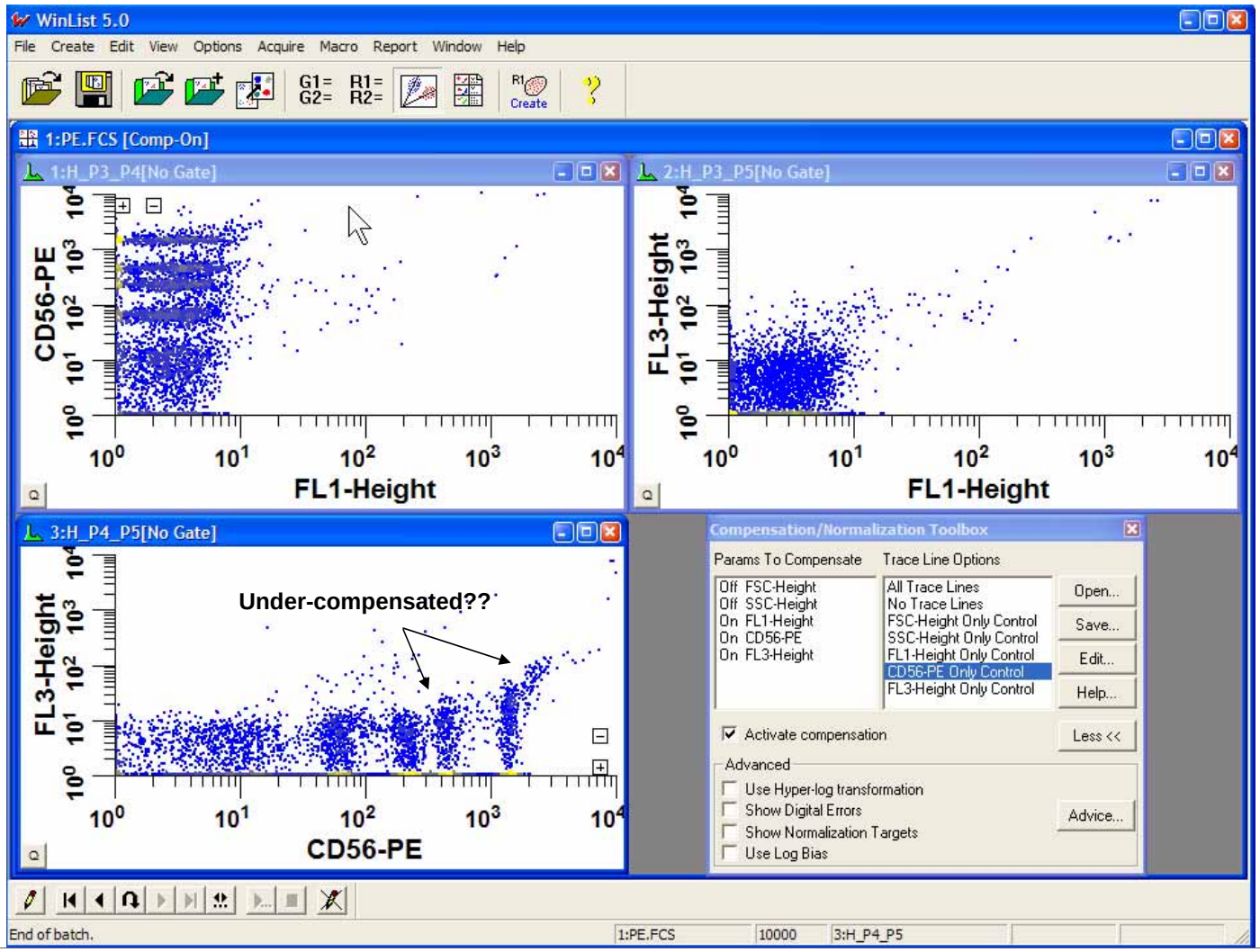
C. BRUCE BAGWELL[a] AND EARL G. ADAMS[b]

[a]*Maine Medical Center Research Institute (MMCRI)*
Maine Medical Center
South Portland, Maine 04106
and
Verity Software House, Incorporated
Topsham, Maine 04086

[b]*Upjohn Laboratories*
Kalamazoo, Michigan 49001

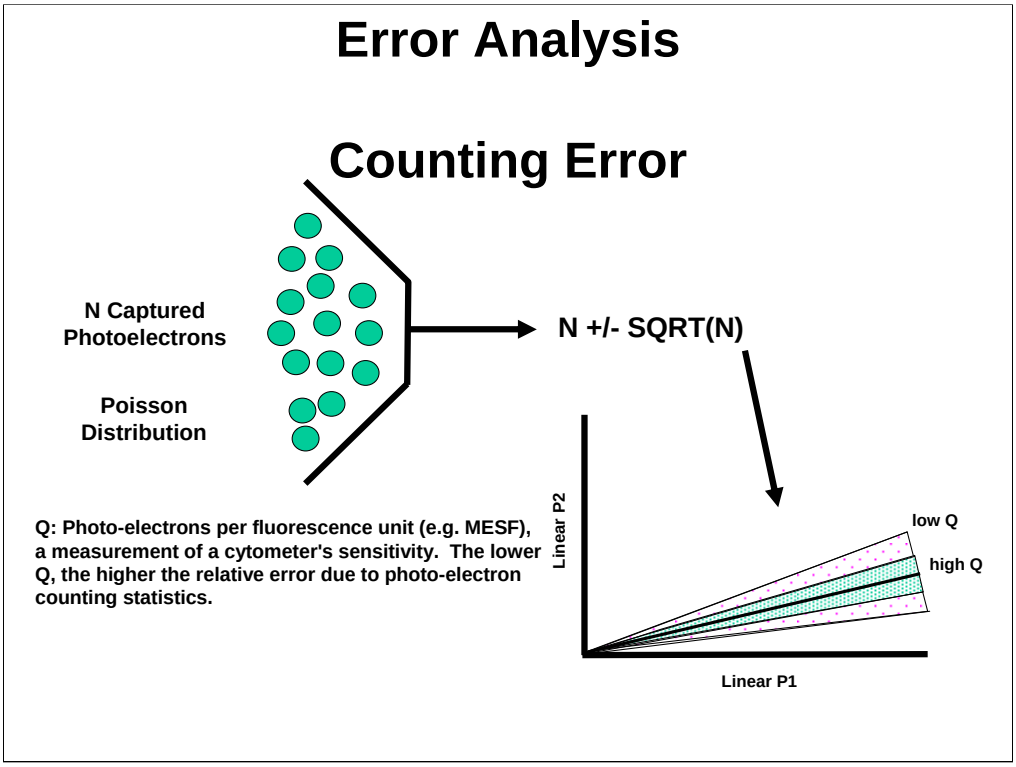
In 1993 Earl Adams and I showed that the entire process of compensation for any number of parameters could easily be solved by these simple matrix operations. The beauty of software compensation is that it can be easily scaled up to involve any number of parameters. It also can be automated. A reprint of the original article is included in your notes.

Overcompensation Bias



The process of software compensation can be quite easy and quick to perform. Each single-color control is loaded into the program and the appropriate signal-crossover slopes are computed. Once the last control is processed, compensation can be turned on for all subsequent analyses. The compensation information can be stored for later retrieval if no adjustments are made either to the fluorochromes or the cytometer's amplification settings.

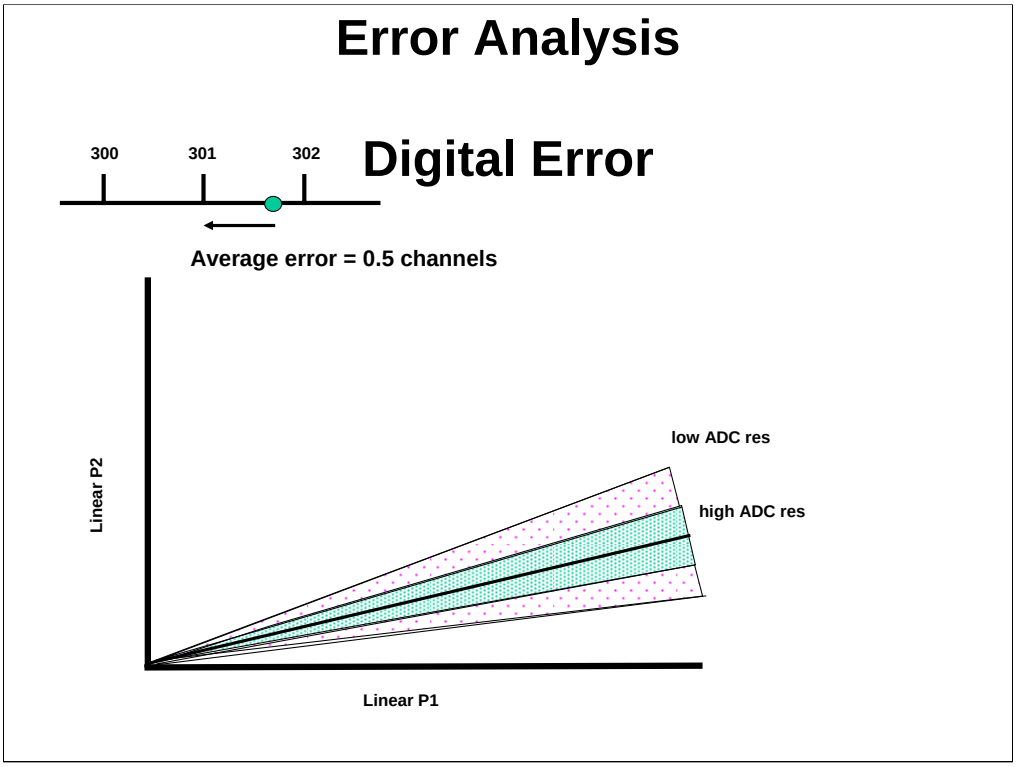
Note the increasing fluorescence intensity for the PE vs Cy5 histogram. Is this an example of under-compensation by the software compensation algorithm? To better understand what's going on, we need to look at some basic types of errors in the cytometer's measurement system and see how they apply to compensated data.



The first error we'll consider is at the level of signal detection. Since the number of photo-electrons ultimately captured by the cytometer's PMT is finite, the signal quantitation is stochastic with a variance governed by the Poisson distribution. The square-root of the number of captured photo-electrons is approximately the standard deviation of the error distribution.

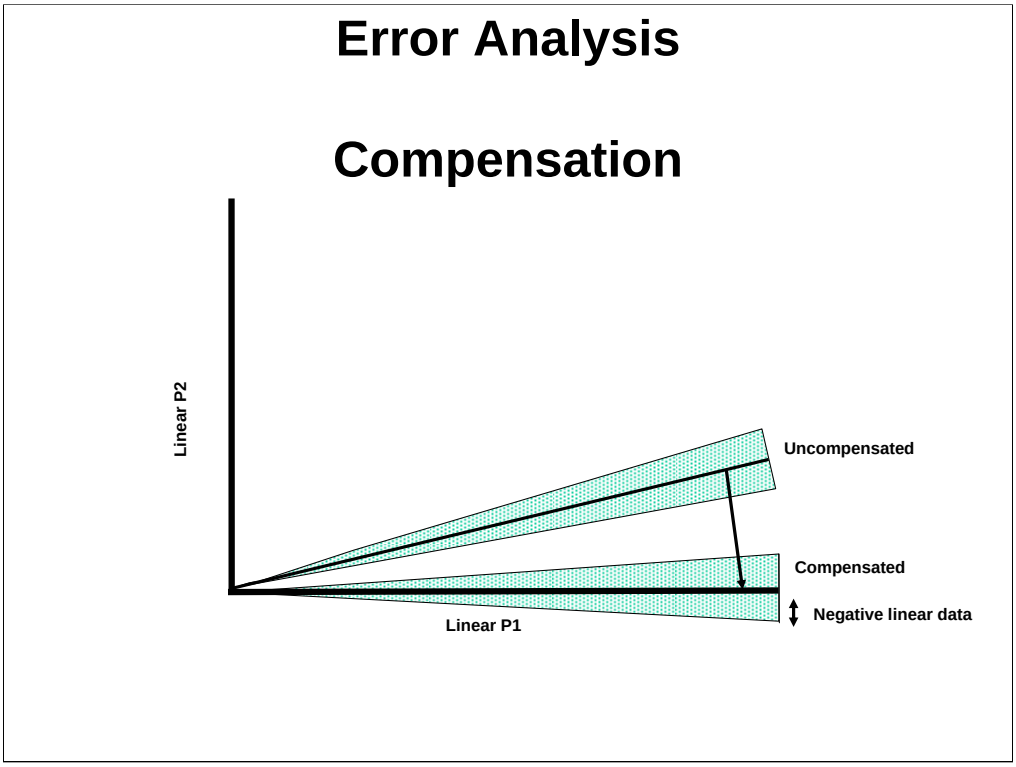
Q is a measurement of the cytometer's efficiency and has units of number of photo-electrons per unit fluorescence intensity (e.g. MESF). The lower the Q value, the higher the amount of relative counting error associated with the signal. The plot in the lower-right depicts this error for a single-color control primarily fluorescing in P1 and crossing-over into P2 with low and high Q values.

In laboratory we'll use a special virtual cytometer to explore this phenomenon in detail.



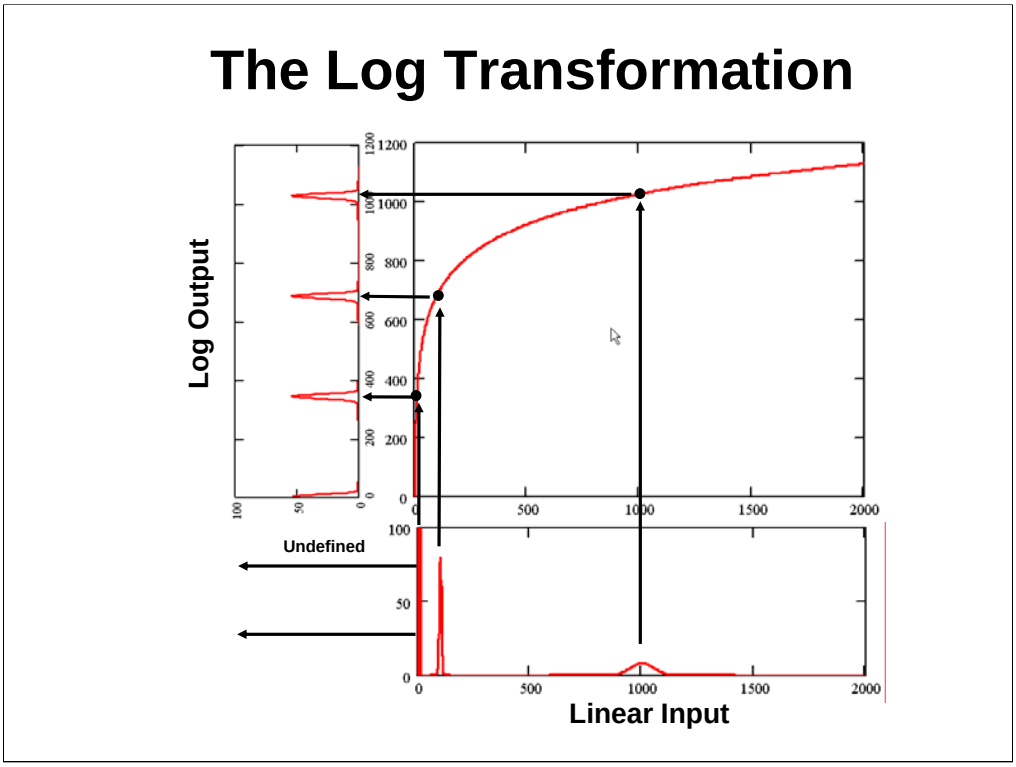
The second error we'll consider is the truncation of the analog log amplified signal to an integer value, digital error. On the average, digital error yields, on average, an error of ½ ADC channel. Unfortunately, this error is augmented by the log transform so that truncations at high ADC values can have a substantial error.

In laboratory we'll use a special virtual cytometer to explore this phenomenon in detail.

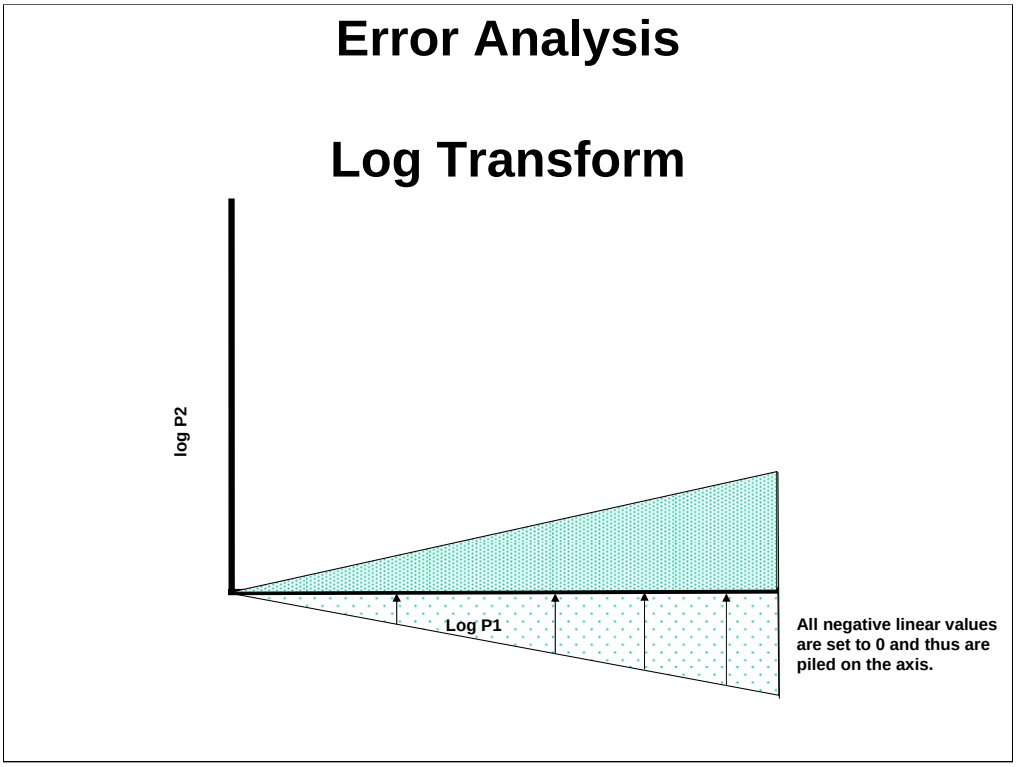


The combination of counting and digital error creates a symmetric error distribution about the single-color trace line. When this data is properly compensated, this distribution becomes symmetric about a horizontal line with some events defined below the axis. These negative values create several problems for us when we transform the axes to log.

Note that this error exists before and after compensation. Compensation does not create this error, it simply exposes it.

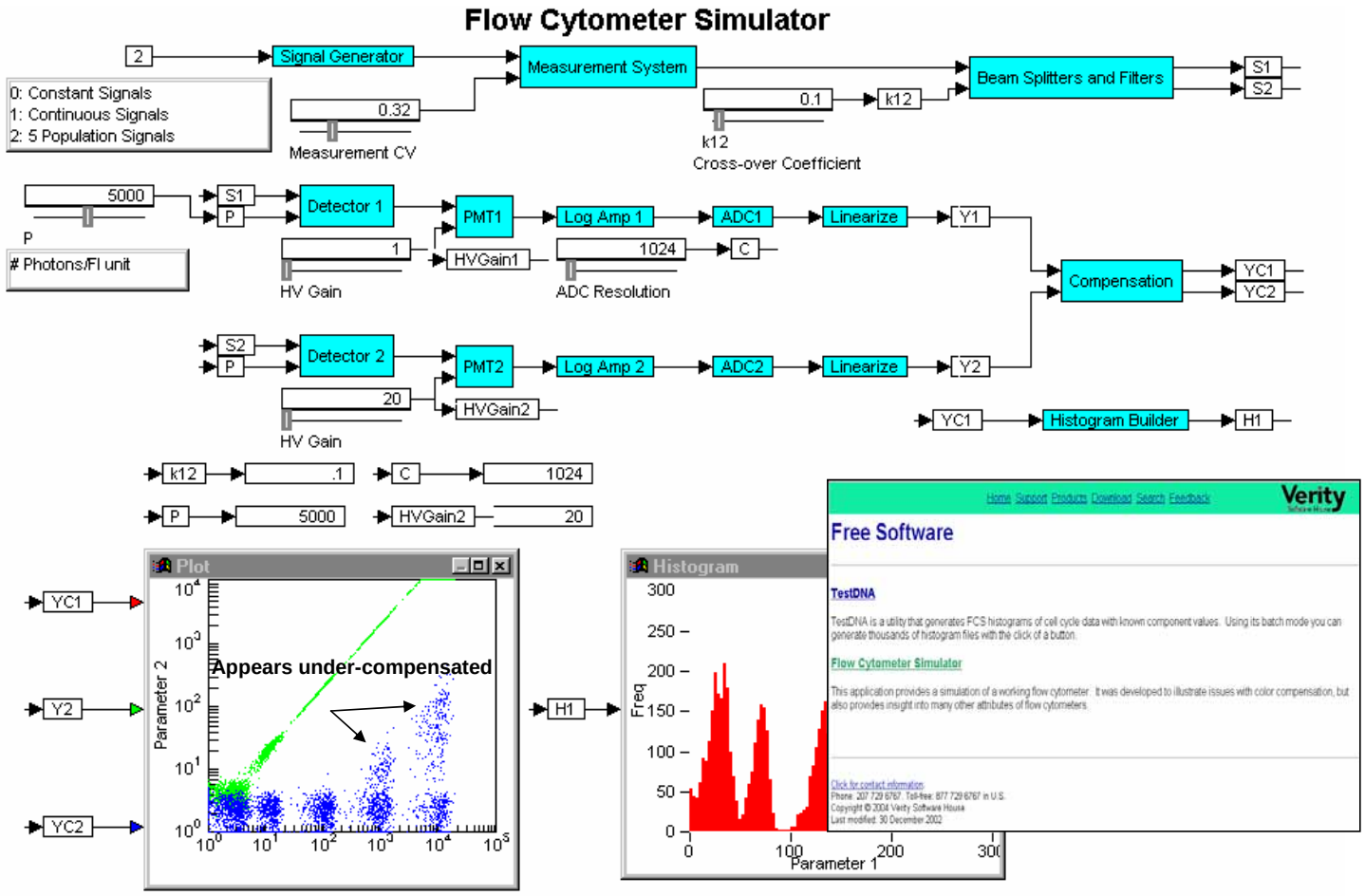


Since the very beginning of cytometry it has been recognized that the log transform is useful for displaying a wide dynamic range of signals. The other important characteristic of the log transform is that it normalizes populations of linearly increasing standard deviations to constant standard deviations. If the log transform didn't have this characteristic, it would be very difficult to appreciate populations separated by more than two decades of intensity. Unfortunately, the log transform is undefined at 0 or negative values (see above), which creates a few problems in the proper interpretation of the data.



Because the log transform is undefined at negative values, software algorithms must set these values to zero, creating a relatively large number of events directly on the axis. The loss of the negative value information creates two display problems. The first is that many events are now piled on the axes, which essentially masks their presence. A worse problem, however, is that from the cytometrists point-of-view, the data looks like it is under-compensated since the symmetric negative error envelope is now gone. This under-compensated appearance is the single most important reason why traditionally listmode data has been over-compensated.

Flow Cytometer Simulator
www.vsh.com

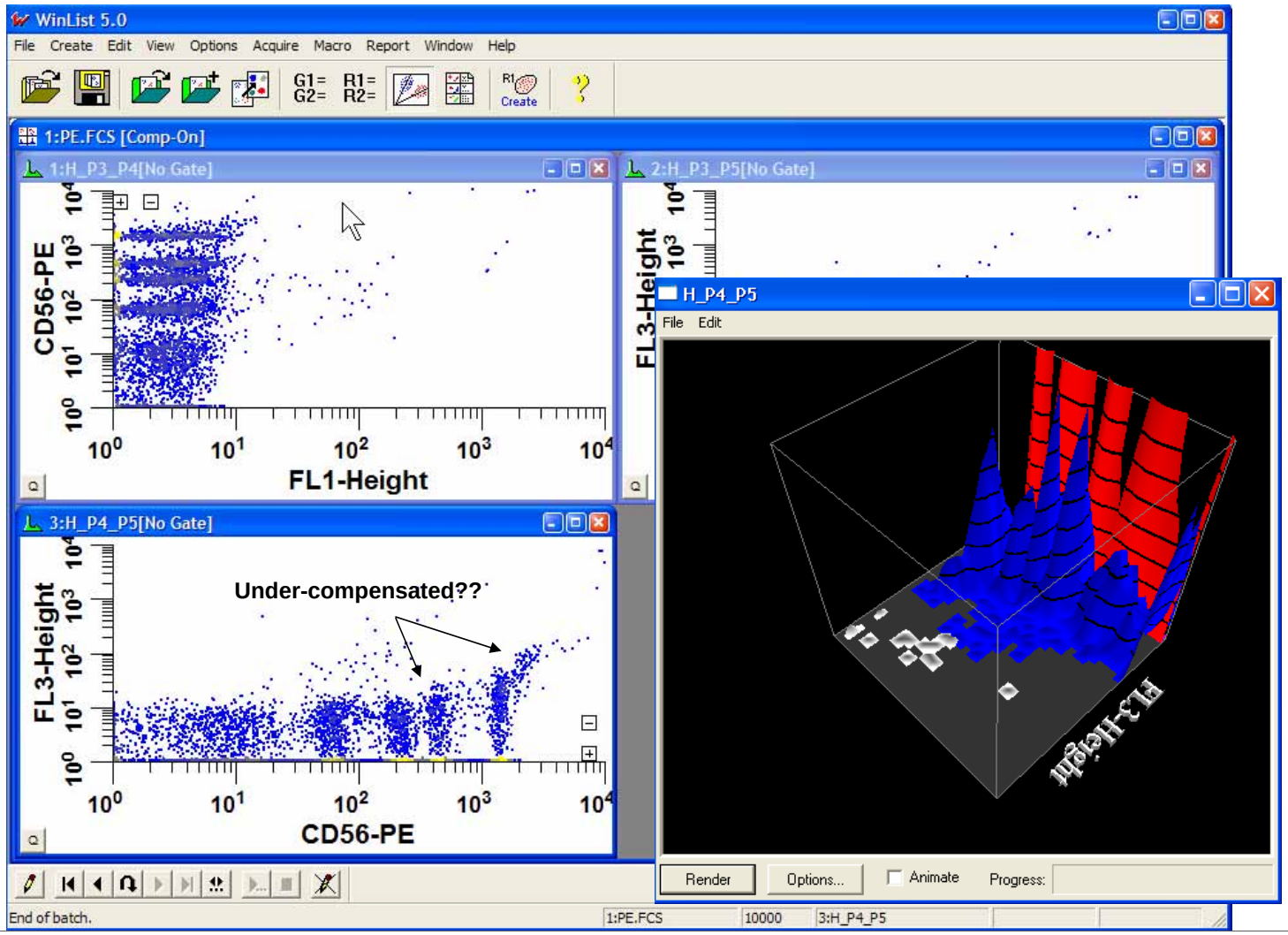


In order to better understand this observation, we will use a flow cytometer simulator in the laboratory. This simulator produces signals of increasing intensity for a single parameter (>100,000 pulses/sec). There are two detectors in the system that detect the primary signal and a crossover signal. The amount of crossover is controlled by k12, which is set to 10% in the above example. The two signals are then routed to PMTs that amplify the signals, controlled by the two HV Gain scrollbars, and then to log amplifiers. The signals are then digitized by the ADCs (Analog to Digital Converters). The system then tries unscramble these two signals by subtracting the exact amount of P2 from P1. The results are graphically shown in the panels at the bottom. The left panel shows the original uncompensated signal (green) superimposed on the compensated signal (blue). The single-parameter histogram for the compensated primary signal is shown in the right panel.

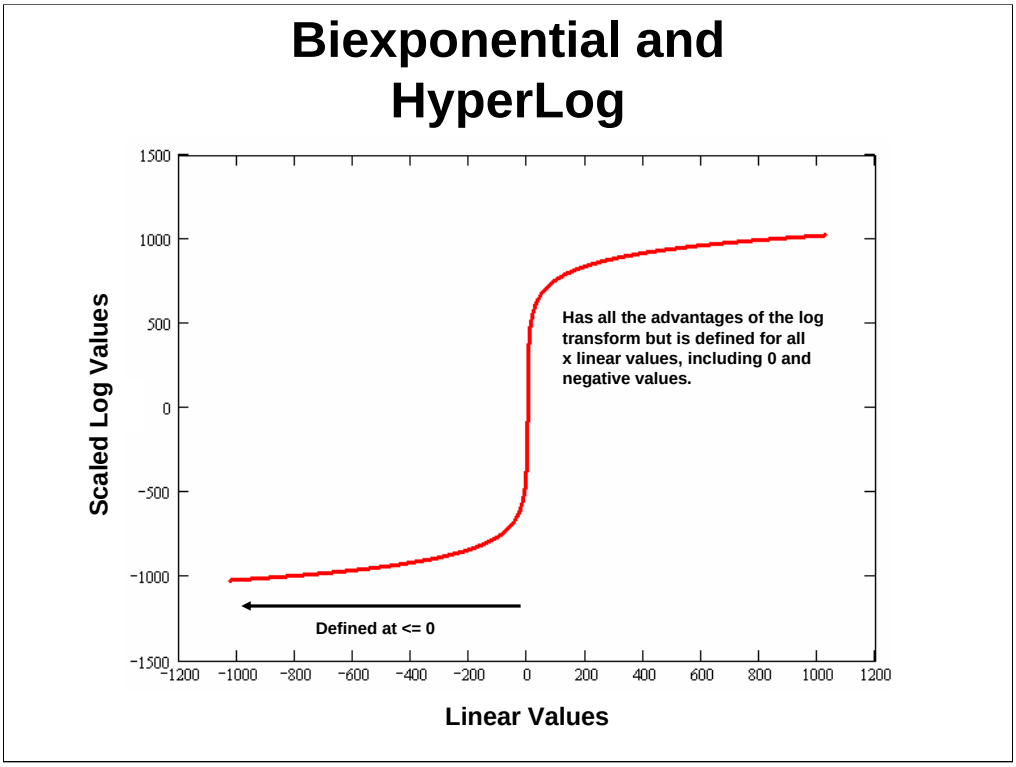
As can be appreciated in the above panel, a perfectly compensated signal can appear to be under-compensated. The demonstration will show that this effect can be due to photon-counting statistics and/or digital error from the ADC. The effect is aggrevated by low light levels, low ADC resolution, high crossover coefficients, and mismatched detector gains.

Note, this simulator can be downloaded for free from the URL, http://www.vsh.com/downloads/Simulator/CytSimulator.htm.

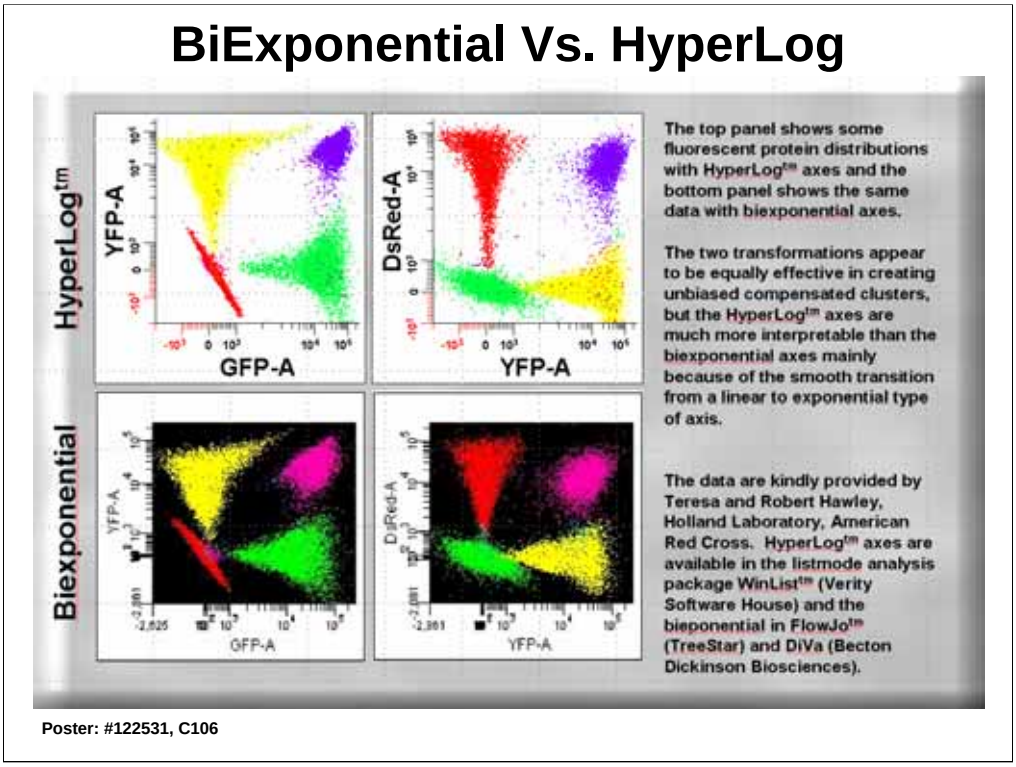
Undercompensated?



Thus, the data that appear to be under-compensated are really not. The upward trend of the upper envelope of the data is due to both counting and digital errors. Is there a better display method that avoids this under-compensated appearance?

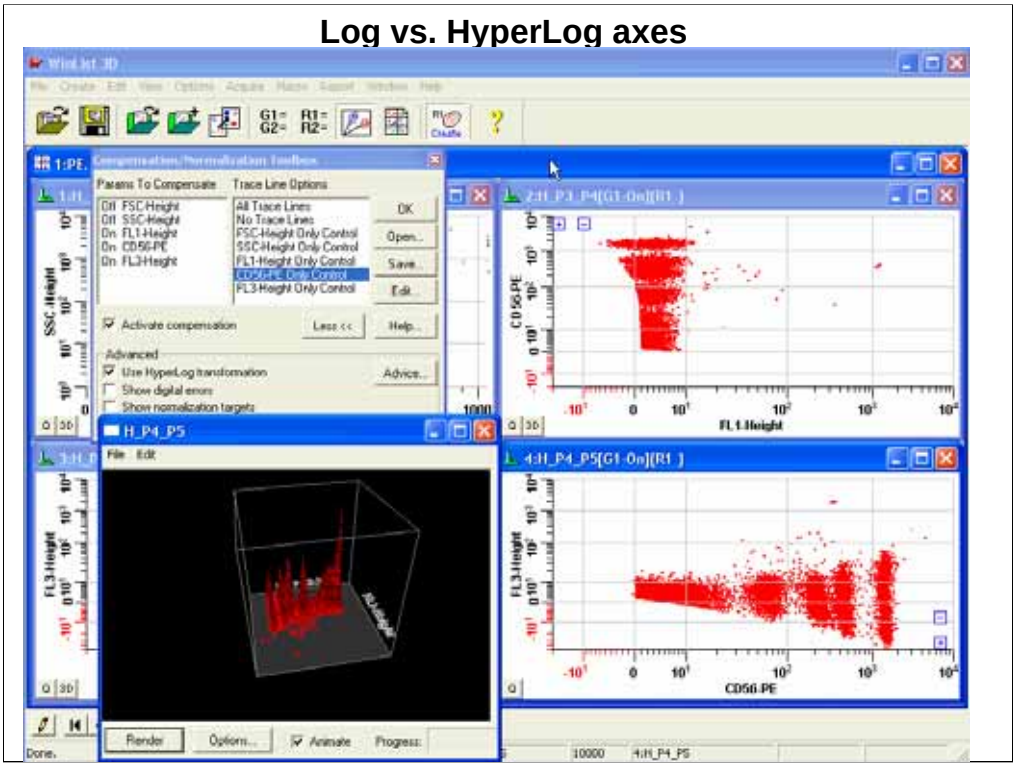


A solution to this cytometry display problem was first proposed by using a generalized form of the hyperbolic sin function, also known as bi-exponential (presented at the 2002 Asilomar Conference, Moore and Parks). More recently, another type of transformation, HyperLog, has been implemented successfully in a listmode analysis software package and published (Bagwell CB: HyperLog – A Flexible Log-like Transform for Negative, Zero, and Positive Valued Data, Cytometry Part A 64A:34-42, 2005). Both of these transforms have similar shape characteristics as shown above. The HyperLog transform was designed to be relatively linear near the origin and to create an intuitively obvious blending of a linear and logarithmic scale. The advantage of these transforms is that the entire domain of linear values can be transformed, preserving the symmetry of the error distribution about our compensated data. For the first time, a compensated dot display does not appear to be under-compensated. These transforms will play an important future role in all listmode acquisition and post-acquisition systems.



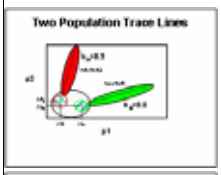
HyperLog axes have an intuitively obvious blending of linear and logarithmic scales which should be well-accepted by other scientific disciplines.

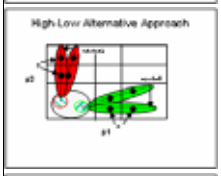
Log vs. HyperLog axes

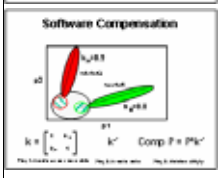


The above animation shows the difference between Log and HyperLog displays. If the display axes are converted to a HyperLog scale, we can see the symmetry of the error distributions and the data appear perfectly compensated. When the display axes are just logarithmic, the data appear to be under-compensated. Also, note that the data are no longer piled up on the axes with the HyperLog axes.

Summary

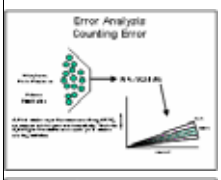




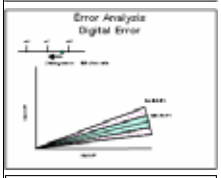


- **Single-color control populations have a diagonal distribution with slopes that quantify the signal crossover between two parameters.**
- **The single-color autofluorescence population is a difficult to measure population that usually does not have the same location as the unstained population. When compensating, do not use the unstained population as a reference.**
- **The best manual procedure for performing compensation is the high-low technique that does not involve the autofluorescence population.**
- **If available, use software to compensate the data since it works well for any number of parameters and is accurate. Software compensation uses matrix operations to perform compensation.**

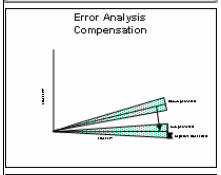
Summary



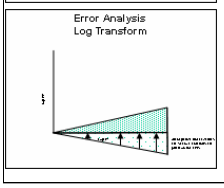
➢Signal detection is stochastic with a variance governed by the Poisson distribution of captured photo-electrons. The error distribution is approximately symmetric and is influenced by the efficiency of the cytometer.



➢The loss of information when a signal is converted to an integer is called digital error. If the signal has been amplified by a logarithmic amplifier, the digital error for the higher ADC values can be large.

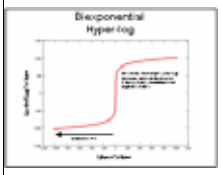


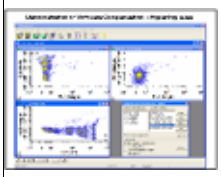
➢The mathematics of compensation rotates the single-color control population so that it is parallel to one of the axes. In many cases the symmetric counting and digital error creates negative linear values. Compensation does not create this error, it exposes it.



➢When the linear data is transformed to log, the negative linear values are set to 0, which creates a large number of events on the axis and makes the data appear to be under-compensated.

Summary





➢New transforms such as the bi-exponential and HyperLog have all the desirable characteristics of the log transform, but also transform 0 and negative values allowing the visualization of the true shape of the compensated population.

➢With these new transforms, it is now possible to appreciate properly compensated parameters in listmode analysis display systems.

The Stewart Summary

There really is no reason to be terrified about multicolor compensation.

The Stewart Summary

Yes, if you are a novice, you can create problems for others.

The Stewart Summary

And yes, there are some distortions that you need to be aware of.

The Stewart Summary

But, the point is...

The Stewart Summary

…if you are not afraid to look for the answers…

The Stewart Summary

…you will find them and control your destiny. Live long and compensate!