Effects of Resolution Reduction on Data Analysis

Authors:
Hunsberger B, Verity Software House, Inc.
Bagwell CB, Verity Software House, Inc.
Herbert D, Verity Software House, Inc.
Bray C, Verity Software House, Inc.
Langweiler M, Dartmouth-Hitchcock Medical Center

Contact Info:
Benjamin C. Hunsberger
Verity Software House
PO Box 247, Topsham, ME 04086 U.S.A.
207 729 6767
bch@vsh.com

Abstract and Key terms:

Background:
There is often a need in flow cytometry to display and analyze histograms at resolutions lower than those native to the data. It is common, for example, to analyze DNA histograms at 256-channel resolution, even though the data were acquired at 1024-channels or more. The most common method for reducing resolution, referred to as the Consecutive Summation method (CS), can introduce distortions in the shape of histograms. Peaks that were symmetric in the original data can become skewed in the reduced-resolution histogram. Data analysis can be negatively affected by the distortions produced by reducing the histogram resolution.

An alternative technique for reducing histogram resolution, the Unbiased Summation method (US), minimizes shape distortion. This paper describes the US method and examines the benefits it provides in the analysis of DNA histograms.

Methods:
Reduced Chi-square (RCS) was used to measure the response to three experimental variables in the least-squares analysis of simulated DNA histograms. For each of the variables (%CV, number of events, and mean position of the G1 distribution), a test dataset of 1000 histograms was generated at 1024-channel resolution. Histogram resolutions were reduced using each method, and then analyzed with ModFit LT cell-cycle analysis software (Verity Software House, Topsham, Maine, USA). S-phase error and processor computation time of each method were also evaluated. A Monte Carlo experiment was performed to compare CS and US methods to theoretically correct reductions.

Results:
CS method analysis results were negatively affected by changes in %CV, number of events, and G1 peak position. The US method produced consistently lower RCS values (more accurate results) within the tested ranges. The US method eliminated bias in S-phase error and had negligible impact on analysis processing speed. It improved RCS values 44.50% on average ($p < .0002$) with actual DNA histograms. While the CS method became less accurate (Chi-square test) as the amount of reduction increased, the US method was unaffected, producing consistently better results.

Conclusions:
The Unbiased Summation method is recommended for reducing histogram resolution in modeling applications such as DNA cell-cycle analysis. It may have implications in other areas of flow cytometry data analysis.

Key terms:
Algorithms, Flow Cytometry/*methods, Software, DNA Content Analysis, Binning.

Introduction:

This paper arises from the following observation: when a DNA histogram with a small CV is analyzed at two different channel resolutions by modeling software, the goodness-of-fit measured by the Reduced Chi-Square (RCS) (1) is lower for the higher resolution histogram, indicating a better fit of the data. This observation was unexpected; examination of the formula for RCS would suggest that histogram resolution should not have a significant impact on the analysis. Our task here is to explain the observation, and to offer a solution that will eliminate the effect.

To better understand the problem, we first created simple histograms, each with a single, normally-distributed population (2). We then reduced the resolution of the histograms by a factor of four and examined the resulting distributions. The computer algorithm that performed the resolution reduction adds the frequencies of four consecutive channels and stores the result in one channel of the low-resolution histogram.

Reducing the resolution produced histograms with distortions in shape. Peaks that were symmetric in the original histogram became asymmetric when the resolution was reduced. We hypothesized that the observed difference in RCS values was at least in part due to the change in symmetry of the peaks in the histograms.

With a hypothesis for the cause of the problem, we considered two corrective actions. First, we could suggest that all DNA histograms be acquired and analyzed at 1024 channels. While this solution might have a positive impact on the problem at hand, it creates other, more significant problems. High-resolution histograms for DNA would require laboratories to run four-times as many cells through the cytometer in order to achieve frequencies at which accurate S-phase estimates can be made (3). Published guidelines for DNA acquisition and analysis suggest standardizing on the use of 256-channel histograms in order to obtain consistent inter-laboratory results (4). These guidelines have been used successfully to obtain clinically-significant prognoses, and weigh heavily in support of using 256-channel analysis resolution.

An alternative solution would be to find a replacement algorithm for reducing histogram resolution, one that preserves the original shape of the distributions more accurately. We present and evaluate an alternative in this paper: the Unbiased Summation (US) method.

While this paper focuses on the impact on DNA analysis, resolution reduction is commonplace in many aspects of flow cytometric analysis. The implications, therefore, pertain to virtually every aspect of data analysis of high-resolution digitized data. Appendix A examines the theory of digital data reduction from a more general point of view.

Materials and Methods:

We designed a set of analysis experiments using simulated DNA histograms. For each experiment, 1000 histograms were generated with 1024-channels of resolution using DNA simulation software (TestDNA, Verity Software House, Topsham, ME, USA, freely available utility on www.vsh.com). One experimental variable, (%CV, G1 mean position, or number of events), was randomly varied in each experiment to evaluate the RCS of each reduction method.

For Experiment 1, the histograms were generated with %CV ranging from 1% to 8%. Experiment 2 varied the G1 peak position between channel 40 and 200 in the high-resolution histogram. This range was selected based on recommended practices for typical DNA analysis. In Experiment 3, we varied the number of events (cells) in the histograms between 1000 and 100000. For each experiment, each of the other parameters was held constant: %CV at 1%, peak position at channel 80, and number of events at 40000. Table 1 summarizes the DNA histograms generated for each of the experiments.

The histogram resolution was reduced to 256 channels and histograms were automatically analyzed with ModFit LT cell-cycle analysis software (Verity Software House, Topsham, Maine, USA). This software performs a non-linear least squares analysis to fit a model to the histogram data. In the automatic mode used in these experiments, the software selects and applies a model to the histogram without user intervention. Version 3.0 was used for the CS method, and a modified version was created to implement the US method. The modified version differed from Version 3.0 only in the reduction algorithm used.

Reduced Chi-Square and S-phase error were used to measure the response of each method in the experiments. RCS is a measure of how well a fitted model matches a histogram, with lower values suggesting a better match. S-phase error is computed as the difference between the S-phase percentage generated by the simulation software, considered the *true* S-phase for these experiments, and the percentage computed by the cell-cycle analysis software

A Monte-Carlo experiment was designed to compare the accuracy of the two methods over a range of reduction factors, in order to determine whether the distortion effects were related to the magnitude of the reduction factor. In this experiment, a Gaussian distribution was created with a mean at channel 8, an area of 1000 events, and a standard deviation of 1.0. From this distribution, 200 *increased*-resolution Gaussians were generated by randomly selecting a factor between 2 and 100 and scaling the original population upwards. The increased-resolution distributions were then analyzed with the CS and US methods of reduction. A Chi-square analysis was performed to compare the resulting reduced-resolution distributions with the original, low-resolution Gaussian.

Results:

We refer to the typical algorithm used to perform resolution reduction as the Consecutive Summation (CS) method. To reduce the resolution by some factor, *i*, the algorithm iterates through all of the high-resolution histogram channels. Let *n* range from 0 to (maximum channel – 1). The frequencies of *i* consecutive channels are summed and stored in the *n/i* channel of the low-resolution histogram

For example, to reduce the resolution of a 1024-channel histogram by a factor of four to 256-channels, the frequencies stored in channels 0, 1, 2, and 3 of the 1024-channel histogram are added together and stored in channel 0 of the 256-channel histogram. Frequencies in channels 4, 5, 6, and 7 of the 1024-channel histogram are summed and stored in channel 1 of the 256-channel histogram, etc.

To understand the characteristics of different data reduction methods, we first examine the theoretically ideal behavior. Panel A of Figure 1 shows a simple example Gaussian distribution with a mean of 16. The distribution is also shown at two, reduced resolutions: the ideal x/2 distribution centered on channel 8 and the ideal x/4 distribution centered on channel 4. The important point to observe about the reduced resolution distributions is that they are symmetrical. See Appendix A for discussion of how these ideal reductions are formulated.

Panel B of Figure 1 graphically illustrates the results of the CS method on the original example distribution. The solid gray lines depict the ideal distributions from Panel A. The line with filled squares shows the population reduced by a factor of two with the CS method, and the line with triangles shows the CS factor of four reduction. As shown in this example, there can be distortion in the shape with the CS method. Depending on where the peak is centered, it may be skewed toward lower channels, toward higher channels, or not at all in the reduced resolution histogram. The worst-case skewing is shown in the figure. Figure 2, Panel A, graphically describes how the CS method combines channels from the original resolution to form a reduced resolution histogram.

The Unbiased Summation (US) method is designed to minimize skewing of distributions in the reduced histogram. Channels from the high-resolution source histogram are distributed in the low-resolution histogram so that the shapes of peaks are preserved. The US method dampens the shape distortion effects of resolution reduction, regardless of where the peak is located.

For example, in a four-fold reduction, frequencies in channels 0, 1 and half of 2 from the source histogram are added and stored in channel 0 of the reduced histogram. Half of the frequency of channel 2 is added with frequencies of channels 3, 4, 5, and half of 6, and stored in channel 1 of the reduced histogram. This pattern is repeated for the entire histogram. Figure 2, Panel B, illustrates this process for a four-fold reduction example, and Figure 8 provides an example C++ implementation of the algorithm.

In Panel C of Figure 1, the US method is applied to the same example distribution shown in Panel A. The original distribution and ideal reductions are shown with solid gray lines. The line with squares shows the factor of two reduction using the US method, and the line with triangles depicts the reduction by a factor of four. The shape of the original distribution is maintained in the reduced histogram, and there is no apparent skewing. Note that the US reduction closely approximates the ideal, reduced-resolution Gaussian distribution.

In Experiment 1, depicted in Figure 3, the %CV of the G1 distribution was varied between 1% and 8%. The CS method of data reduction (squares) yielded increasing RCS values as %CV decreased. RCS was greater than 5.0 for %CV values below 3%. This RCS threshold has been well-documented as an important acceptance criterion for DNA content analysis (4,6). It is important to note that this experiment held the G1 position constant at channel 80, which creates the worst-case skewing in the CS method. The US method (triangles) produced lower RCS values for the entire %CV range tested, and RCS values were all below 5.0.

In Experiment 2, depicted in Figure 4, the G1 peak position was varied between channels 40 and 200. The CS method (squares) produced increasingly variable results as G1 channel position was reduced. The zoomed view inset of Figure 4 shows a repeating pattern of the CS method, related to the reduction factor of four and the position of the G1 peak. The US method (triangles, close to the axis) was robust for all tested G1 channel positions, producing consistently lower RCS values.

In Experiment 3, depicted in Figure 5, the number of events in the histograms varied between 1,000 and 100,000. Using the CS method (squares), the RCS increased as the number of events in the histograms increased. The US method (triangles, close to the axis) was unaffected by the number of events in the histograms tested. It produced consistently lower RCS values compared to the CS method. As with Experiment 1, it is important to note that the G1 position was held constant at channel 80, which creates the worst-case skewing in the CS method.

The accuracy of the S-phase fraction estimate of the analysis software was evaluated for each of the three experimental data sets. S-phase error was computed as the difference between the generated S-phase percentage and the percentage computed by the cell-cycle analysis software. Figure 6 shows the S-phase error comparison for Experiment 1 where %CV was varied from 1% to 8%. The CS method, shown in squares, showed a slight bias toward underestimating S-phase by approximately 0.6% on average. The US method, shown in triangles, showed no bias. Figure 7 shows the S-phase error comparison for the G1 position experiment. The CS method, shown in squares, is again biased toward underestimation of S-phase, and the variability increases as the G1 position is reduced. This finding has been previously reported (3). The US method shows no bias in the S-phase estimate, though it also begins to increase in variability as G1 position is reduced. Similar results, not shown, were observed in Experiment 3.

The computation speed of the two methods was compared to evaluate the performance of each. A computer subroutine executed each resolution reduction algorithm 1,000,000 times. Microsoft C++ compiler was used to compile the test code, and the test was performed on an Intel Pentium III processor running at 1 GHz. Using profiling software to compute the average time in microseconds for a single iteration of each method, the US method was on average 2.2% slower than the CS method.

Seven DNA histograms from human samples were analyzed to compare RCS using CS and US methods. The DNA histograms were from bone marrow cells prepared according to published methods (5) and run on a FACSCalibur Multipurpose Cytometer from BD Immunocytometry Systems. Table 2 summarizes the results of this comparison. The US method produced lower RCS values for all tested histograms, and showed less variability for the test data set. The change in RCS averaged 44.50%. Using a Chi-Square test of statistical significance, the change in results was found to be significant at $p < .0002$. S-Phase percentage (not shown) was less than 1% different between the two methods on average.

A Monte-Carlo experiment was designed to determine whether the distortion effects seen in the CS method were related to the magnitude of the reduction factor. Figure 9 shows the results of Chi-Square tests for 200 reduction factors ranging from 2 to 100. The Chi-square value compares the histogram resulting from the reduction with the original, low-resolution histogram, which is considered to be “truth” for this experiment. The error of the CS method (squares) increases dramatically as reduction factors increase from 2 through 10, and continue to increase with reduction factors from 10 through 100. The US method (triangles) produces consistently low Chi-Square results across the tested range of reduction factors.

Discussion:

The method typically used to reduce histogram resolution, CS, often distorts the shape of histograms, in some cases skewing distributions. This observation may explain why high-resolution histogram analysis with DNA modeling software yields lower Reduced Chi-Square, compared to low-resolution analysis. Since the modeling software uses a Gaussian function to mathematically fit peaks, simple histograms that have symmetric peaks can be modeled more accurately than those with skewed peaks. The CS method can introduce asymmetry in peaks, with the effect of degrading the accuracy of the analysis. The proposed US method minimizes the distortion of shape and produces lower RCS values in DNA modeling analysis as a result.

RCS is elevated when CS is used to reduce histogram resolution for DNA cell-cycle analysis. As the %CV of a G1 distribution decreases, the CS method produces increased RCS values. The CS method causes RCS to rise as the G1 peak position moves toward lower channels on a DNA histogram. The effects of the CS method become more pronounced as the number of events in the histograms increase. S-phase error in cell-cycle analysis is biased low when CS is used.

Conversely, when using the US method in DNA cell-cycle analysis, RCS is relatively unaffected by %CV, G1 peak position, or number of events. The US method produces lower and less variable RCS values in all tests performed. In addition, S-phase error is not biased with the US method.

The very small impact on processing speed using US method is mitigated by its benefit to the analysis of DNA histograms with cell-cycle analysis software.

Further study is recommended to determine the effects of resolution reduction methods on more complex DNA histogram forms, e.g. those with debris. It would also be informative to examine the effects of these methods on other types of flow cytometry data analysis.

The use of the proposed method (US) results in a statistically significant improvement in RCS. There were no negative effects on S-phase percentages observed in the tests performed.

Perhaps the most surprising result of this study was that the US method was unaffected by the magnitude of the reduction factor, as shown in the Monte-Carlo experiment. While the CS method was greatly compromised by increasing reductions, the US method was remarkably robust across the full range of tested factors. This unexpected finding has significant implications for some of the newer, high-resolution digital flow cytometers, which are capable of millions of channels of resolution. If the conventional CS method is used to reduce resolutions to common analysis resolutions of one thousand channels or less, the reduction process can introduce significant changes in the distributions. The US method provides an alternative that is unaffected by the size of the reduction factor.

Appendix A: Resolution Reduction Theory

Digital measurement systems, like flow cytometers, convert continuous signals to digitized values. Normally, the conversion is done by binning signals into uniform intervals called channels. The size of these intervals and the measured signal's dynamic range determine the resolution of the digitized parameter. Typical parameter resolutions are 1024 channels in flow cytometry, but newer digital cytometers have resolutions often exceeding 1 million.

In order to appreciate the distribution of the measurement parameters, the digital data is often rebinned into lower resolutions and represented as histograms. We refer to this process of rebinning into a lower resolution as resolution reduction. In this section, we will not discuss non-uniform binning as found in log transformations, although the binning biases described below are also applicable to non-uniform binning as well.

Resolution reduction can be performed on data stored as a histogram or as digitized listmode data. Suppose we wanted to reduce the resolution by a factor of four. If the data were stored as a histogram, we might sum up every four consecutive channels and associate this sum with a single lower resolution channel (Figure 1, Panel B). For listmode data, we would divide the data by four or equivalently, bit shift by two, and store the resultant values in a histogram.

Unfortunately, both these approaches create the same biases in the reduced resolution histograms, and under some circumstances, these biases can be detrimental to other analysis methods such as the one presented in this paper.

In order to appreciate these biases, we will construct an example and then show how the above commonly used methods create biases in the resultant resolution reduced distributions.

Let our continuous measurement parameter be given by a Gaussian,

$$\frac{A}{\sqrt{2\cdot\pi}\cdot\sigma}\cdot e^{\frac{-(x-\mu)^2}{2\cdot\sigma^2}}\cdot dx \qquad \text{Eq. 1}$$

where,

A: area of the Gaussian,
σ: standard deviation,
μ: mean,
x: continuous independent variable.

If we want to bin this distribution into channel *c* and reduce the resolution by some factor *k*, we use the integral. In this expression, *k* is equivalent to the reciprocal of fold of the reduction, e.g. *k*=1/4 for a four-fold reduction.

$$\mathrm{GausBin}(c,\mu,\sigma,A,k) := \int_{c-\frac{1}{2}}^{c+\frac{1}{2}} \frac{A}{\sqrt{2\cdot\pi}\cdot k\cdot\sigma}\cdot e^{\frac{-(x-k\cdot\mu)^2}{2\cdot(k\cdot\sigma)^2}}\,dx \qquad \text{Eq. 2}$$

Figure 1, Panel A, examines a specific example of a Gaussian distribution reduced by a factor of two and four using Eq. 2, and demonstrates that they remain symmetric after resolution reduction and the means

of the distributions are at their expected locations. These are the theoretically correct reduced resolution distributions, with means at channel 8 and channel 4 respectively.

If we apply the conventional consecutive summation (CS) technique to reduce the resolution by factors of two and four, we can appreciate the biases in the CS method. As depicted in Figure 1, Panel B, the distributions are no longer symmetric and their means shift to the left. The same distributions would be produced by dividing the listmode parameter values by two and four respectively and storing the results in histograms (data not shown).

The complete mathematical proof for this observation is outside the scope of this paper, but the problem is caused by using a non-symmetric resolution reduction method. The Unbiased Summation (US) method described by Figure 2, Panel B, is symmetric and can be applied to histogram data to reduce the resolution without as much bias. An implementation of the US method in C++ is shown in Figure 8.

The Chi-square statistic (1) is a useful way of comparing the reduced resolution distributions to the theoretically correct distributions. Columns A and B in Table 3 summarize how well the new US method works compared to the CS method for this sample data. The Chi-square values demonstrate that the US method produces a distribution that is much closer to the exact, theoretical reduced resolution distributions.

In order to compare these two methods for a larger variety of conditions, we use a Monte Carlo simulation examining thousands of combinations (n=2000) of Gaussian means (16-48), standard deviations (2-7), and reduction factors (2-8).

The US method results in far lower Chi-square values than the conventional CS method (Table 3, Column C), with a smaller standard deviation of results as well (Table 3, Column D).

References:

1. Bevington, PR. Data reduction and error analysis for the physical sciences. New York: McGraw-Hill Book Company; 1969. p 204-246.

2. Van Dilla MA, Dean PN, Laerum OD, Melamed MR. Flow Cytometry: Instrumentation and Data Analysis. Academic Press Inc. (London) Ltd; 1985. p195-218.

3. Herbert DJ, Bagwell CB, Munson ME, Hunsberger BC. Examination of Some DNA Guidelines Using Simulation Data. ISAC Poster presentation. Cytometry 1998; Supplement 9:96.

4. Bagwell CB, Clark GM, Spyratos F, Chassevent A, Bendahl P, Stål O, Killander D, Jourdan M, Romain S, Hunsberger B, Baldetorp B. Optimizing Flow Cytometric DNA Ploidy and S-Phase Fraction as Independent Prognostic Markers for Node-Negative Breast Cancer Specimens. Cytometry 2001, 46:121-135.

5. Weaver DL, Bagwell CB, Hitchcox SA, Whetstone SD, Baker DR, Herbert DJ, Jones MA. Improved flow cytometric determination of proliferative activity (S-phase fraction) from paraffin-embedded tissue. Am J Clin Pathol 1990 Nov;94(5):576-84.

6. Bagwell CB, Clark GM, Spyratos F, Chassevent A, Bendahl P, Stål O, Killander D, Jourdan M, Romain S, Hunsberger B, Wright S, Baldetorp B. DNA and Cell Cycle Analysis as Prognostic Indicators in Breast Tumors Revisited. Clinics in Laboratory Medicine 2001, 21:4, 875-895.

Figures:

Figure 1: Ideal, CS, and US Method Examples

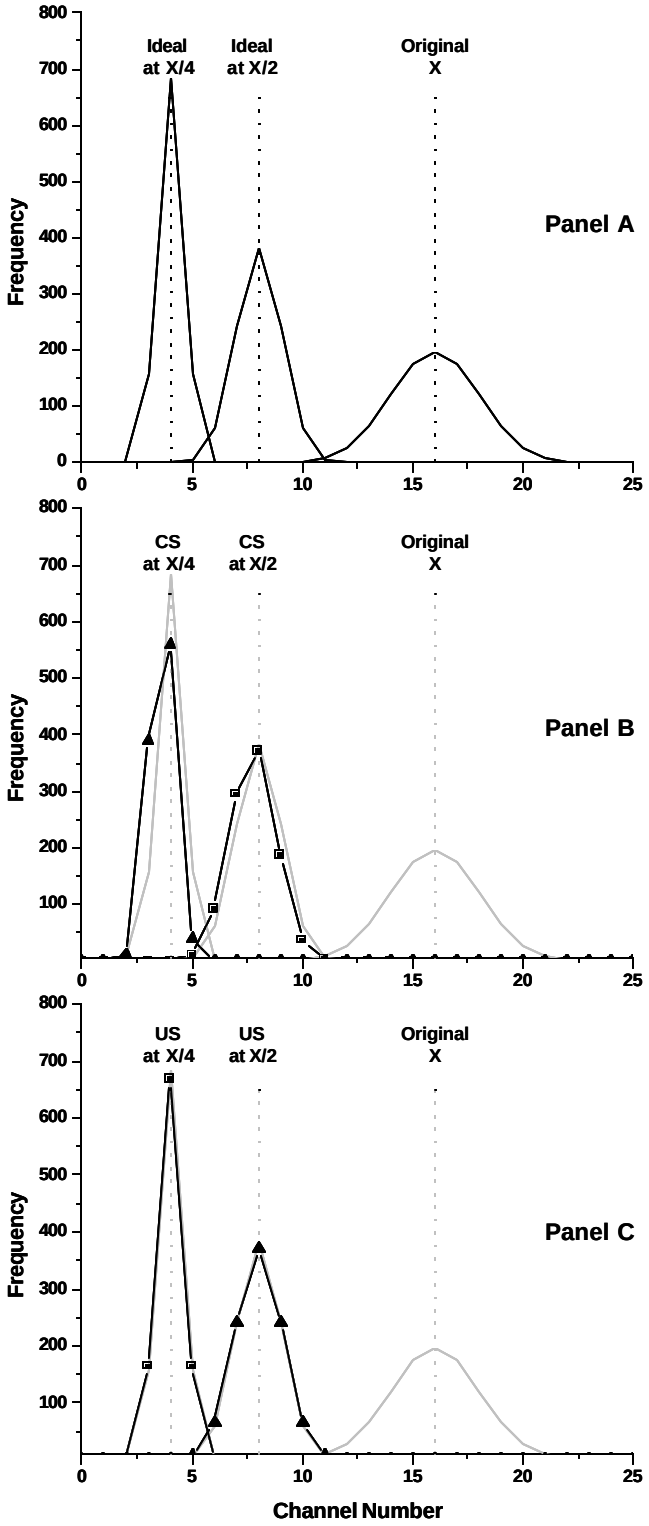


Figure 2: Method Comparison

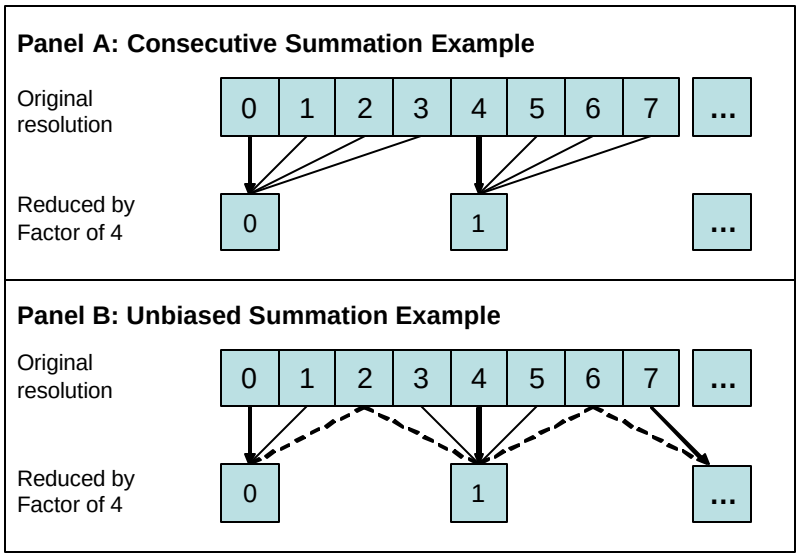


Figure 3: RCS Response to %CV in Generated Histograms

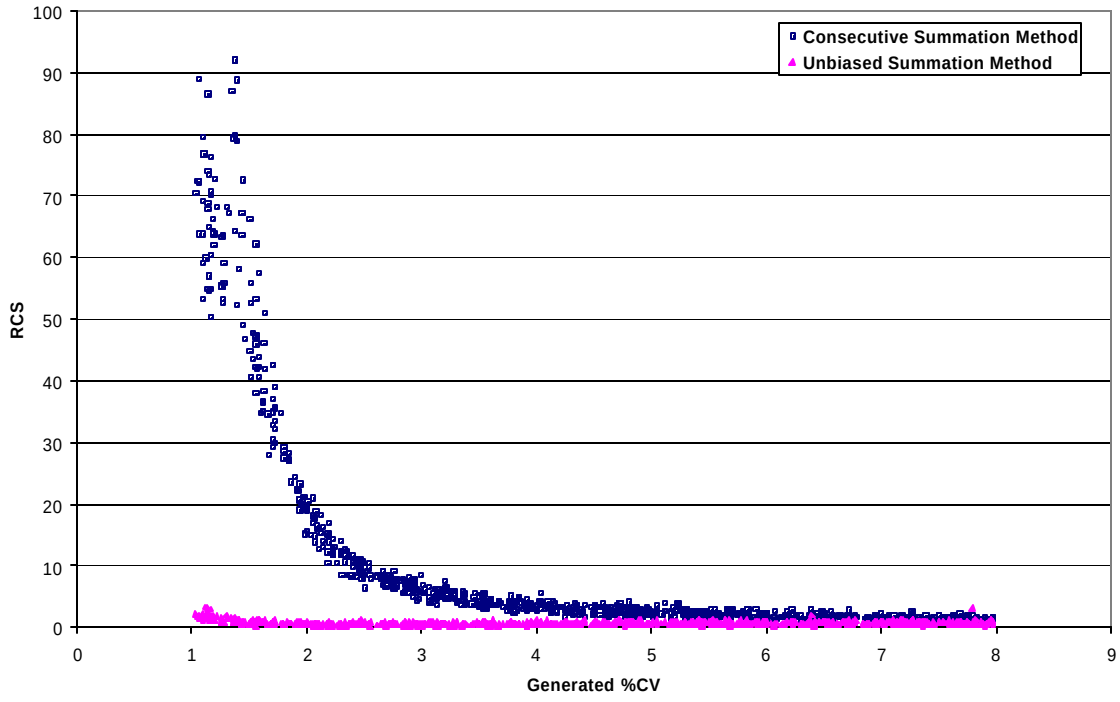


Figure 4: RCS Response to G1 Peak Position in Generated Histograms

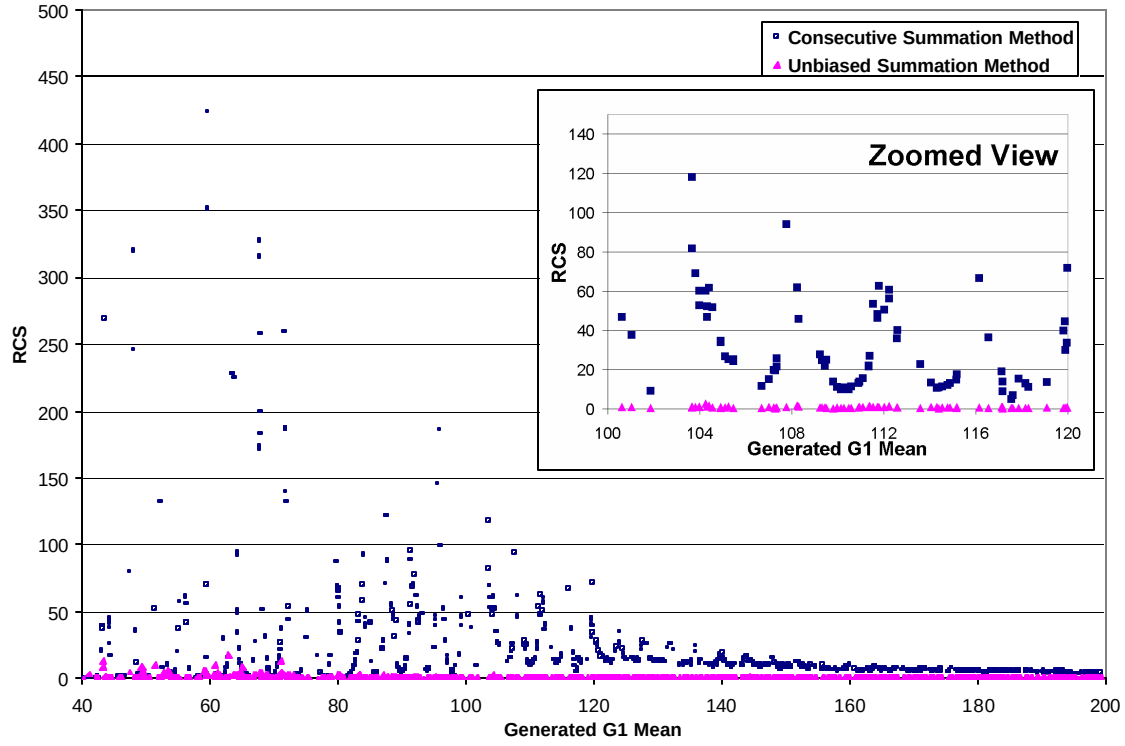


Figure 5: RCS Response to Number of Events in Generated Histograms

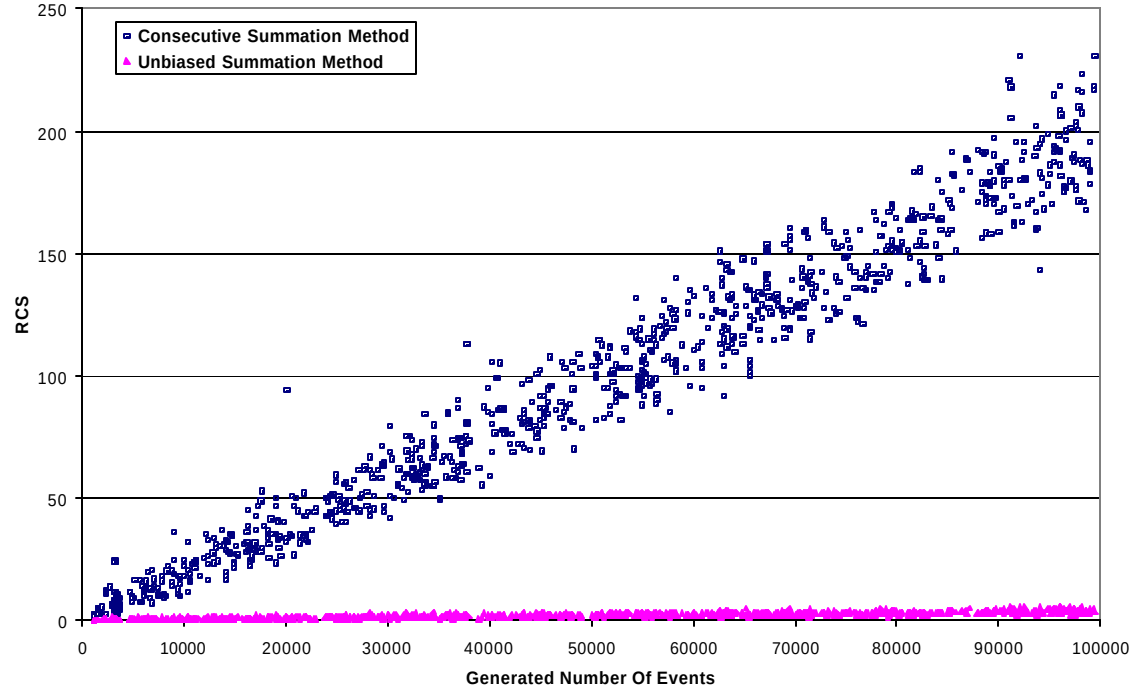


Figure 6: S-phase Error Comparison in CV Experiment

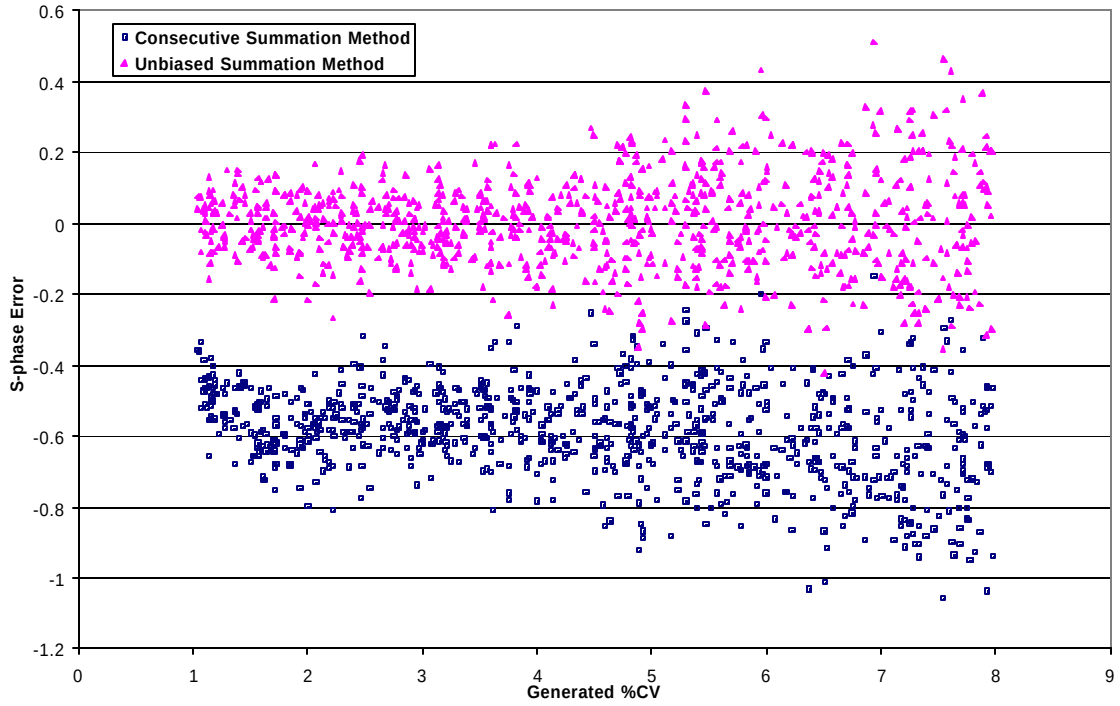


Figure 7: S-phase Error Comparison in G1 Position Experiment

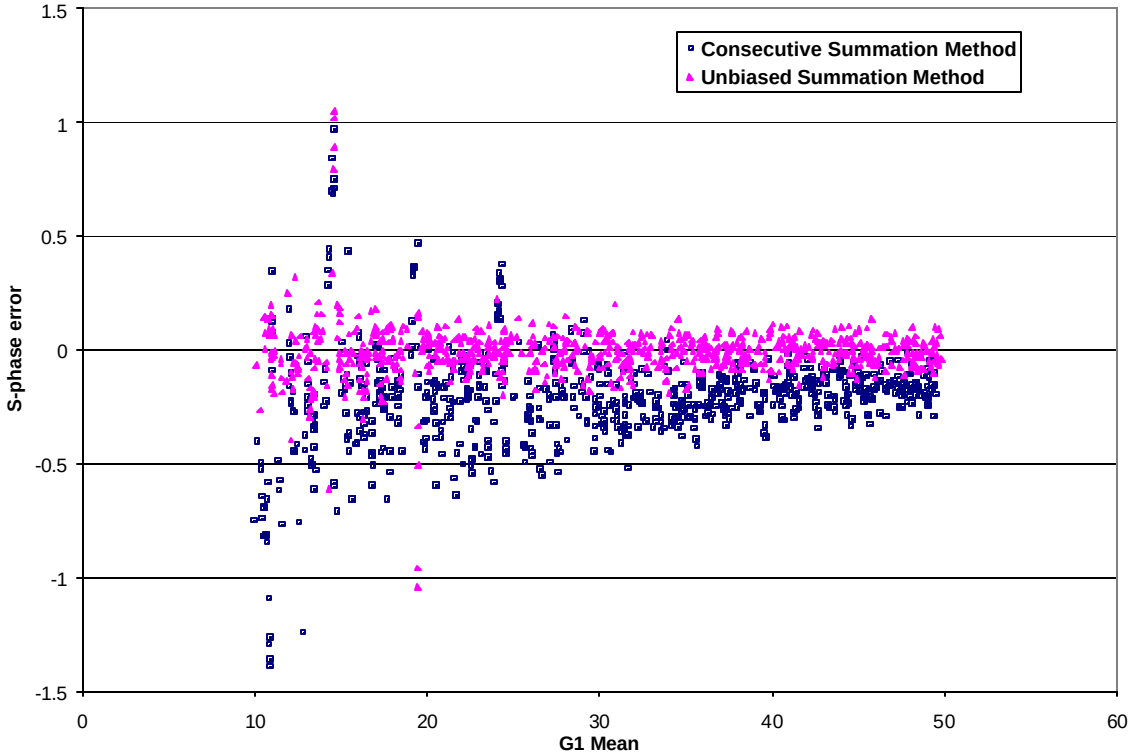
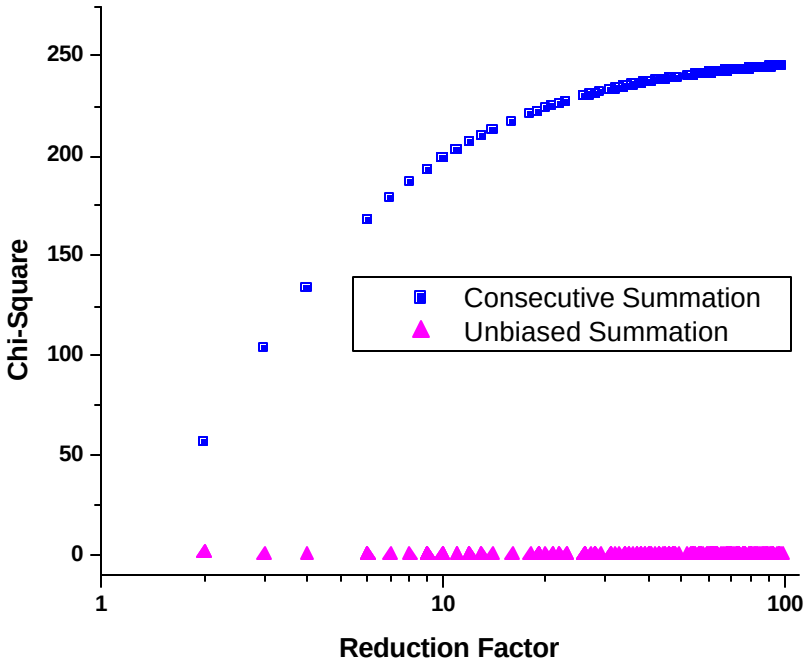


Figure 8: C++ Computer Algorithm for Unbiased Summation

```
void UnbiasedSummationReduction(int sourceRes, int destRes, DWORD *pData)
// Perform an inplace unbiased summation reduction of resolution
// from sourceRes to destRes.
// pData is assumed to be an array of sourceRes elements.
{
 int groupChannels = sourceRes/destRes; // Channels to group
 int midChan = groupChannels/2;      // Channel that needs to be split.
 int odd = groupChannels % 2;        // Odd or even number of channels
 int j = 1;                          // Channel-in-group index
 int k = 0;                          // Reduced-res channel index

 for(int i=1; i<sourceRes; i++)      // Iterate through the hi-res data
  {
   if(j != midChan)
    pData[k]+= pData[i];
   else
    {
     if(odd == 0)                    // even number channels to reduce
      {                              // spread over two low-res channels
       int half = pData[i]/2;
       pData[k] += half;

       if(pData[i]%2)                // prevent loss of a count
        pData[k] +=1;

       if(k<destRes-1)               // increment the low-res channel
        k++;

       pData[k] += half;
      }
     else                           // odd number channels to reduce
      {
       pData[k] += pData[i];
       if(k<destRes-1)
        k++;
      }
    }
   pData[i] = 0;                    // zero out the hi-res channel
   j++;
   if(j>=groupChannels)
    j=0;
  }
}
```

Figure 9: C++ Computer Algorithm for Unbiased Summation



Tables:

Table 1: Summary of experiments

Experiment	Variable	Range	Important Constants
1.	**%G1 CV**	**1%-8%**	**G1 channel = 80, Number of events = 40000**
2.	**G1 channel**	**40-200**	**%G1 CV = 1%, Number of events = 40000**
3.	**Number of events**	**1000-100000**	**%G1 CV = 1%, G1 channel = 80**

Table 2: Comparison with Real DNA Histograms

Data File	A. CS Method RCS	B. US Method RCS	C. Change % (CS-US)*100/CS
01-1256.002	10.583	6.392	39.60%
1363.002	12.378	5.818	53.00%
1378.001	6.132	3.674	40.08%
1404.001	15.205	5.925	61.03%
1412.001	6.782	5.250	22.59%
1441.001	1.296	1.230	5.09%
1498.001	2.430	2.126	12.51%
Mean	7.829	4.345	44.50%
Std. Deviation	5.140	2.031	

Table 3: Chi-square comparison of theoretical distributions

	Example		Monte Carlo	
	A. Chi-square for reduction of x/2	B. Chi-square for reduction of x/4	C. Mean Chi-square	D. Standard Deviation
CS method	**58.8**	**540.2**	**255.3**	**346.2**
US method	**1.6**	**1.2**	**0.1**	**0.4**

Legend:

Figure 1: Panel A depicts a Gaussian distribution at channel 16, and the ideal reductions at 1/2 and 1/4 scale at channels 8 and 4 respectively. Panel B depicts the original distribution reduced with the CS method. Shape distortion and shifting of the means are apparent. Panel C depicts the original distribution reduced with the US method, which preserves the shape more accurately.

Figure 2: This figure compares the two methods for a 4-fold reduction example. The Consecutive Summation method, Panel A, sums 4-consecutive channels of the original resolution and stores the result in a single channel at the reduced resolution. The Unbiased Summation method, Panel B, distributes the original data more symmetrically. Dotted lines indicate that events in the original resolution channel are divided between two channels in the reduced resolution data.

Figure 3: Experiment 1 examined generated histograms in which the %CV of the G1 distribution was varied between 1% and 8%. The CS method (squares) yielded increasing RCS values as %CV decreased. RCS was greater than 5.0 for %CV values below 3%. The US method (triangles) produced lower RCS values for the entire %CV range tested, with RCS values all below 5.0.

Figure 4: Experiment 2 examined the G1 peak position in generated histograms, which was varied between channels 40 and 200. The CS method (squares) produced increasingly variable results as G1 channel position was reduced. The zoomed view inset highlights the repeating pattern of the CS method, related to the reduction factor of 4 and the position of the G1 peak. The US method (triangles, close to the axis) was robust for all tested G1 channel positions, producing consistently lower RCS values.

Figure 5: Experiment 3 examined the effect of number of events in generated histograms, which was varied between 1,000 and 100,000. Using the CS method (squares), the RCS increased as the number of events in the histograms increased. The US method (triangles, close to the axis) was unaffected by the number of events in the histograms tested. It produced consistently lower RCS values.

Figure 6: The S-phase error was compared for Experiment 1 data, where %CV was varied from 1% to 8%. The CS method (squares) showed a slight bias toward underestimating S-phase by approximately 0.6% on average. The US method (triangles) showed no bias.

Figure 7: S-phase error was compared for data in the G1 position experiment. The CS method (squares) was biased toward underestimation of S-phase, and the variability increases as the G1 position is reduced. The US method (triangles) showed no bias, though variability also increased as G1 position is reduced.

Figure 8: The US method can be implemented on a histogram using a C++ algorithm. The implementation shown is written to be readable, and is not necessarily the most efficient structure.

Figure 9: A Monte-Carlo experiment tested the accuracy of the two methods over a range of reduction factors. The graph plots the results of Chi-Square tests for 200 reduction factors ranging from 2 to 100. The CS method (squares) becomes less accurate as the reduction factor increases. The US method (triangles) produces consistently low Chi-Square results across the tested range.

Table 1: Three primary experiments were constructed to compare and test the CS and US methods. Each experiment constructed 1000 simulated DNA histograms and varied one experimental variable. The first experiment varied the %CV, the second varied the G1 peak position, and the third varied the number of events in the histogram.

Table 2: Seven DNA histograms from human samples were analyzed to compare RCS using CS and US methods. The US method produced lower RCS values for all cases, and showed less variability for the test data set. The improvement in RCS was 44.50% on average.

Table 3: To evaluate the methods statistically, a Chi-square test was used to compare each method with the ideal reduction. For the example histogram (columns A, B), the US method substantially outperformed the CS method in reductions by factors of 2 and 4. A Monte Carlo comparison of 2000

histograms found that the US method yields significantly lower Chi-square results, with less variability than the CS method (columns C, D).

Erratum Pertaining to Manuscript 02-099-.R1:
B. Hunsberger, C. B. Bagwell, D. Herbert, C. Bray, and M. Langweiler:
Effects of Resolution Reduction on Data Analysis.
Cytometry Part A 53A:103-111 (2003)

The example C++ routine, Figure 8, in our original manuscript contains a coding error. The error affects the second channel in factor-of-two reductions, and would result in a loss of events in that channel.

The error does not exist in the implementation used by the authors in the ModFit LT application code, and therefore does not affect the results published in the article. When the example code was written for the article, the actual code was revised to make it more readable and the error was introduced. The authors regret the error, and gratefully thank Wen-Tang (Mike) Shen and Peter S. Rabinovitch of the Dept. of Pathology, University of Washington, Seattle, for identifying the problem and calling it to our attention.

The following figure shows the corrected example C++ routine to replace Figure 8 in the original manuscript:

```
void UnbiasedReduction(int sourceRes, int destRes, DWORD *pData)
// Perform an inplace unbiased summation reduction of resolution from sourceRes to destRes
// pData is assumed to be an array of sourceRes elements.
{
 int groupChannels = sourceRes/destRes;  // Nnumber of channels to group into one channel.
 int midChan = groupChannels/2;          // The channel to split between two channels.
 int odd = groupChannels % 2;            // Do we have an odd number of channels
 int j = 1;
 int k = 0;

 for(int i=1; i<sourceRes; i++)          // Iterate through the hi-res data
  {
   if(j != midChan)
     pData[k]+= pData[i];
   else
    {
     if(odd == 0)                        // even number of channels in the reduction
      {                                  // spread this channel over two low-res channels
       int half = pData[i]/2;
       pData[k] += half;

       if(pData[i]%2)                    // prevent loss of a count
        pData[k] +=1;

       if(k<destRes-1)                   // increment the low-res channel
        k++;

       // pData[k] += half;              // error in original manuscript

       // Shen-Rabinovitch Correction 1  // CORRECTION 1
       if(i>k)
        pData[k] += half;
       else
        pData[k] = half;
       // End Shen-Rabinovitch Correction 1
      }
     else                                // odd number of channels in the reduction
      {
       pData[k] += pData[i];
       if(k<destRes-1)
        k++;
      }
    }
   // pData[i] = 0;                      // error in original manuscript

   // Shen-Rabinovitch Correction 2      // CORRECTION 2
   if (i>k)
    pData[i] = 0;
   // End Shen-Rabinovitch Correction 2

   j++;
   if(j>=groupChannels)
    j=0;
  }
}
```