13.8 INTELLIGENT THINNING ALGORITHM FOR EARTH SYSTEM NUMERICAL MODEL RESEARCH AND APPLICATION

Rahul Ramachandran*, Xiang Li, Sunil Movva, Sara Graves
University of Alabama in Huntsville

Steve Greco, Dave Emmitt
Simpson Weather Associates

Joe Terry
Science Applications International Corporation

Robert Atlas
Goddard Space Flight Center, NASA

1. INTRODUCTION

As the number of observation platforms and numerical models on all scales continues to rapidly increase, the large amounts of data that they generate threaten to hamper the progress in scientific research and forecast improvement, or even overwhelm (both in time and data-space) the researchers and modelers that use them. The NASA Earth Science Enterprise (ESE) strategy and mission documents specifically note that one of the important challenges facing NASA is that of transforming vast quantities of data and information into products that can be beneficial to users, especially for economic and policy decision making. Pertinent examples of these products would be model-generated forecasts such as hurricane landfall, air quality, 3-day and 7-day weather forecasts. While the assimilation of more and better observations is necessary for forecast improvement, today's global models and data assimilation systems cannot ingest and utilize all of the data available to them due to extremely large computational costs and constrained network bandwidth. This is not only a result of the data volume, but also due to issues in dealing with the potential impact of each additional observation type, as well as differences between model grid size and the data grid/density. For example, typical observational data has a horizontal resolution of 25km or better. In addition, in regions where there is an overlap of orbital paths from different satellites the combined data density is much higher. The crudeness of current techniques used to thin such data to manageable densities indicates the lack of viable methods for dealing with the complex problem of extracting the information-dense data that provides the best representation of the atmosphere.

* *Corresponding Author Address*: Rahul Ramachandran, Information Technology and Systems Center, University of Alabama in Huntsville, Huntsville, AL 35899; email: rramachandran@itsc.uah.edu

This paper describes the development and testing of an automated Intelligent Data Thinning (IDT) algorithm to facilitate improved data assimilation schemes and forecast accuracies by preserving information-dense regions while removing redundant data points. The development of this algorithm is a collaborative effort involving a team of data mining experts at the University of Alabama in Huntsville (UAH), numerical modelers at Goddard Space Flight Center (GSFC) and Simpson Weather Associates (SWA) .

2. BACKGROUND

As space-based observing systems generate ever increasing volumes of data, there arises the need to better discriminate between useful data points and data points which are simply redundant. Data Assimilation Systems (DAS) processing times can increase by as much as the square of the number of observations, depending on the assimilation method used. To circumvent this problem, most operational centers must resort to using very crude thinning methods to reduce data volume. Some of the existing data thinning techniques used in the Earth science modeling and research community include superobing, using a single observation per grid box, using observations closest to the model first guess, and using observations furthest from the model first guess. Superobing is a regional averaging technique using a simple weighting scheme. Data thinning using a single observation per grid box is similar, with the observation closest to the center of each box of a user-defined global grid kept for assimilation. In the data thinning approach using observations closest to the model first guess, only the observation minus model first guess values that are smallest and within a prescribed threshold are kept for assimilation. In other words, more weight is given to the model first guess and less to the observations. With data thinning approaches using observations furthest from the model first guess, only the observation minus model first guess values that are largest and within a prescribed threshold are kept for assimilation. With each of these techniques, the number of observations can be reduced significantly but at the cost of considerable loss of atmospheric structure and information. Therefore, these methods eliminate large amounts of data, some useful,

while retaining a considerable amount of useless information. An example is shown in Fig. 1, where 1A shows all available QuikScat wind observations in a region surrounding Typhoon Beni. A circulation center is clearly evident near 15S latitude and 161E longitude, the center of the typhoon. Fig. 1B shows a much reduced field of QuikScat wind observations to which a current data thinning technique was applied. The rejected data are at nadir, edge, and rain-flagged points. Unlike in the previous figure, no clear circulation is evident in this wind field.

3. IDT ALGORITHM DESCRIPTION

Data thinning in the computer graphics domain, referred to as decimation or simplification, is performed to reduce the number of polygons to be rendered so that there is minimal loss in image quantity. These thinning algorithms use a recursive and greedy removal of "least significant" points from the data sets. These algorithms are adaptive and support a user-specified image quality metric, allowing the algorithm to be controlled to meet an accuracy level. These thinning algorithms are conceptually very simple. The data is viewed as a triangulated mesh, which can be decomposed into hierarchical blocks with different details at every level. Two adjacent triangles are simplified or thinned if the objective measure, in this case the maximum perceived geometric error, is smaller than the user specified threshold. This process is applied recursively until no further simplification is possible [1]. The advantage of these algorithms is that the thinning is not limited to a specific feature and can fully utilize the data. However, the challenge in designing these algorithms is in determining the right objective measure.

The IDT algorithm is based on the algorithmic concepts used in data decimation or simplification [1,2,3,4] and the pseudocode is given in Figure 2. The algorithm searches for regions with large variances and keeps all the data points within such regions. For regions with low variances, the algorithm subsamples the region to select a representative point. There are two preprocessing steps performed before the thinning. The data is first normalized to values ranging from 0 to 1 and then a global mean for the entire data is calculated. The IDT algorithm is computationally efficient, using quad-tree decomposition to recursively divide the data into four quadrants. For each quadrant or region, the algorithm calculates an objective measure. If the objective measure is greater than the cutoff threshold, the algorithm continues dividing this quadrant into four sub-quadrants and repeats the procedure for each of these sub-quadrants. If the objective measure is less than the cutoff threshold, then the algorithm terminates that recursive path and the center data point of the quadrant is used as the representative thinned value. The other termination condition for the algorithm is when the recursion reaches the lowest level where the quadrant contains four points. For this condition, the algorithm saves all four points.

The cutoff threshold used by the IDT to perform recursive splitting is based on an *acceptable standard deviation (aStdDev)* which is calculated by multiplying the user threshold and the global mean. This measure describes the cutoff value for the variance within a region. For a given quadrant, a statistical F-test is performed. The F-test is generally used to test the hypothesis that two sample have different variances by rejecting the null hypothesis that their variances are actually consistent. The F-test in IDT is performed between the quadrant (sample 1) and the hypothetical sample 2 that is represented by *aStdDev*. The null hypothesis in this test is that:

variance of sample 1 (var1) <= variance of sample 2 (var2)

Since the IDT algorithm is searching for only the regions with high variances, the test is performed in two steps. Then *var1* is less than *var2*, signifies that the region has lower variances than the prescribed cutoff, consequently the algorithm stops the recursion and subsamples the region to find the representative data value. If *var1* is greater than or equal to *var2*, then the algorithm calculates the F-Test probability using the size of the data as degrees of freedom. If the F-test probability is within the acceptable limit, then the null hypothesis holds true implying that samples have similar variances. Otherwise the samples have different variances and the region is further split into four quadrants.

4. INITIAL RESULTS

The IDT algorithm was tested on wind field data from a model output. The initial results from the IDT algorithm are extremely encouraging. In the initial experiments, the algorithm was used on U component of the wind data, followed by the V component and the results combined at the end. Two examples of the application of IDT on model output can be seen in Fig. 3 and 4. Figs. 3A and 4A are the original wind field and Figs. 3B, 3C and 4B, 4C are the thinned data. To visually analyze the performance of the IDT algorithm, the thinned data is overlaid with confluence values in 3B and 4B to identify regions of convergence and with vorticity values in 3C and 4C to identify regions of circulation. Dark regions in Figs 3B and 4B represent regions of convergence and similarly light colored regions in Figs 3C and 4C signify regions of circulation. These figures clearly show that the IDT does indeed select regions of interest i.e., regions with high confluence and vorticity values. In both cases, the algorithm thinned the data by a substantial amount (> 85%) while retaining regions of interest.

5. SUMMARY AND FUTURE WORK

An Intelligent Data Thinning algorithm was developed to address the needs of global data assimilation systems that

are unable to handle the enormous volume of observational data due to the prohibitively large computational costs. This recursive simplification algorithm is based on the concepts of data decimation or simplification used in computer graphics. The algorithm uses a quad-tree decomposition to recursively divide the data and calculate an objective measure for the partitioned data. The algorithm searches for regions with large variances and keeps all the data points within such regions. For regions with low variances, the algorithm subsamples the region to select a representative point. The initial analysis of the thinned data produced by the IDT have been very encouraging and the algorithm is currently being tested against other current assimilation methods.

A series of short regional experiments using a variational analysis method (VAM) are currently being conducted to evaluate the data thinning results of the IDT algorithm. The VAM is a two-dimensional wind analysis scheme, developed by Hoffman [5], that has been used extensively within the GSFC to create a variety of ocean surface wind products. Both the intelligently-thinned data fields and randomly-generated wind fields, consisting of an equal number of data points, will be processed by the VAM. The resulting analyzed fields will be compared against the "truth" field from which the thinned data was derived. Multiple analysis background fields and time periods will be tested. There are plans to further test the IDT results using one or more GSFC global data assimilation systems.

6. REFERENCES

[1] Lindstrom, P., D. Koller, W. Ribarsky, L. F. Hodges, N. Faust, and G. A. Turner, 1996: Real-Time, Continous Level of Detail Rendering of Height Fields. *Proceeding of SIGGRAPHS 96, Computer Graphics Proceeding, Annual Conference Series*, New Orleans, Louisana, ACM SIGRAPH, 109-118.

[2] Dyn, N., M. S. Floater, and A. Iske, 2002: Adaptive thinning for bivariate scattered data. *Journal of Computational and Applied Mathematics*, 145, 505-517.

[3] Dyn, N., M. S. Floater, and A. Iske, 2001: Univariate Adaptive Thinning, Report TUM M0106.

[4] Iske, A., 2000: Hierarchical Scattered Data Filtering for Multilevel Interpolation Schemes, TUM-M0007.

[5] Hoffman, R. N., 1984: SASS wind ambiquity removal by direct minimization. Part II: Use of smoothness and dynamical constraints. *Mon. Wea. Rev.,* 112, 1829-1852.

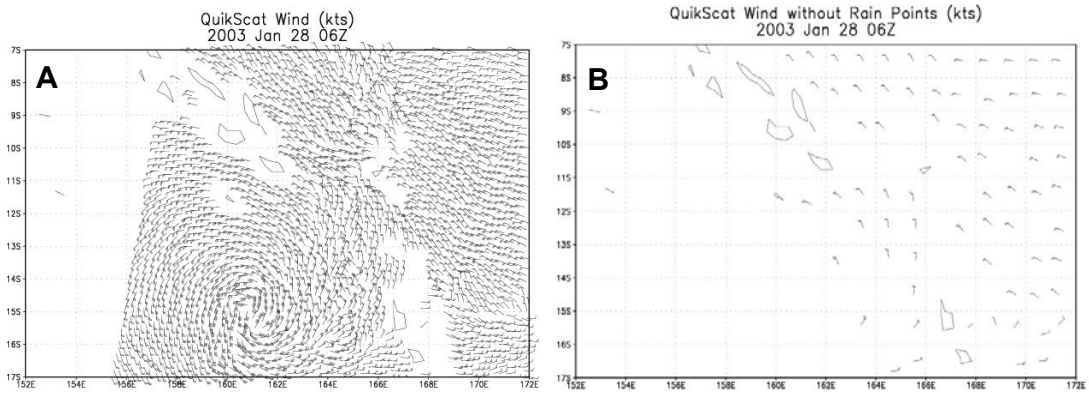

Figure 1: Example of original QuikScat wind data and the result from a current thinning algorithm

```
;Variables:
;data- original data
; ndata – thinned data
; threshold – user specified threshold
; endpoints – Max(X,Y) , min(X,Y)

function main (data , threshold)
{
        Data = Normalize(data)
        globalMean = Mean(data)
        thinnedData = IDT(data, ndata, globalMean, threshold, endPts)
        return thinnedData
}

function IDT (data,ndata, globalMean, threshold, Bounds)
{
        If ( size(data) == 4)
        {
                ndata = data
        }
        else
        {
                Q1,Q2,Q3,Q4= Divide(data) //returns quadrant end points
                Quadrant(data,ndata,globalMean,threshold,Q1)
                Quadrant(data,ndata,globalMean,threshold,Q2)
                Quadrant(data,ndata,globalMean,threshold,Q3)
                Quadrant(data,ndata,globalMean,threshold,Q4)
        }
}

function Quadrant ( data, ndata,globalMean, threshold, Q)
{
        aStdDev = globalMean*threshold
        varSample = Calc_ObjectiveMeasure(data, Q)
        if (varSample > (aStdDev*aStdDev))
                IDT(data,ndata,globalMean, threshold,Q)
        else
                ndata[midPointX,midPointY] = data[midPointX, midPointY]
}
```

Fig 2: Pseudocode for the Intelligent Data Thinning Algorithm

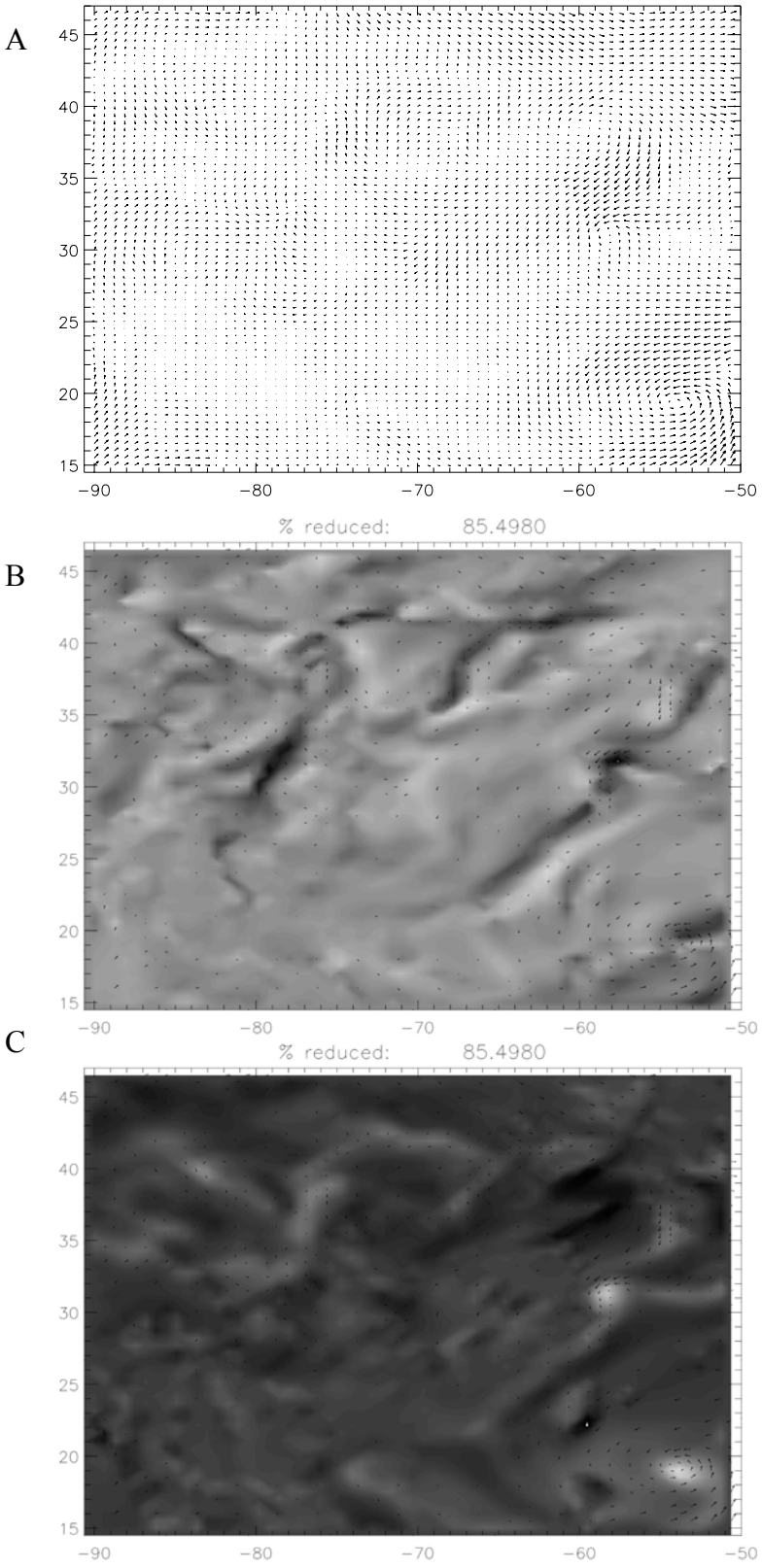

Figure 3: A. Original Data B. Thinned data overlaid with confluences C. Thinned data overlaid with vorticity

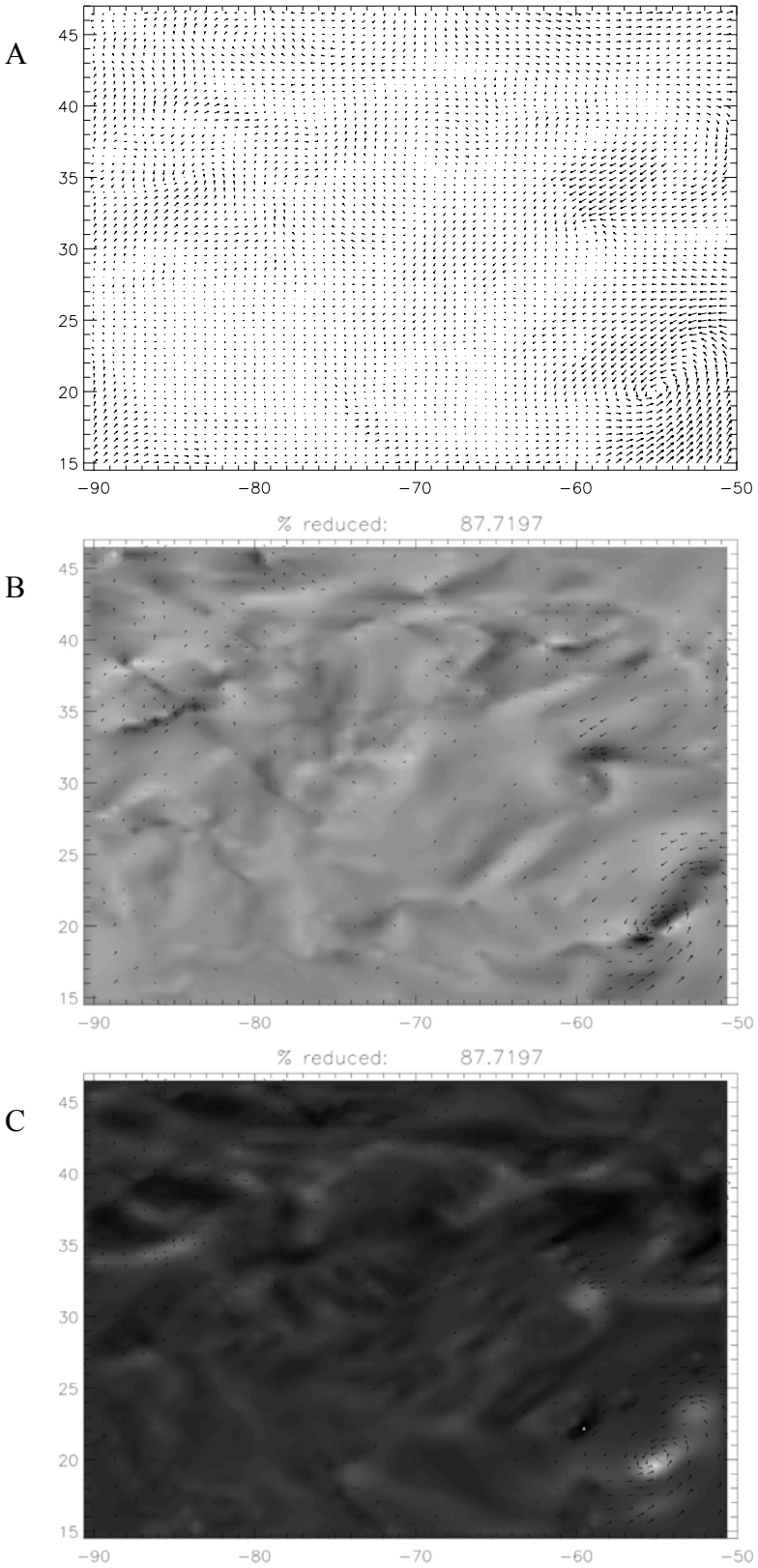

Figure 4: A. Original Data B. Thinned data overlaid with confluences C. Thinned data overlaid with vorticity