A Focus+Context Approach for Visualizing Multi-Dimensional Functions

Balaprasuna Chennupati **Ashwini Pande** **Ravikiran Vatrapu** **Vijay Hanumolu** **Chris North**

Department of Computer Science
Virginia Tech
Blacksburg, Virginia, 24060, USA
{prasuna, aspande, rvatrapu, vhanumol, north}@vt.edu

ABSTRACT

In this paper, we present a focus+context information visualization approach for multi-dimensional functions. We introduce a new interaction metaphor for this type of data, which we call "focal point with a view": a user chosen focal point presents a view of the function with respect to this focal point. The data is viewed from this focal point to see patterns along rays that proceed directly away from the focal point in all directions. The rays can be re-ordered to prioritize on a particular dimension and view the data accordingly. Hence, the behavior of the function can be visualized in a single view not only with respect to a specific point in its domain but also with respect to a specific dimension. While the approach can scale to many dimensions, the current implemented tool is best at about 7 dimensions.

Keywords

Information visualization, multi-dimensional visualization, focus+context.

INTRODUCTION

The data to be represented graphically is a multi-dimensional single valued function. The function is uniformly discretely sampled in a bounded region of the n-dimensional parameter space. Hence, the data can be thought of as an n-dimensional image.

More rigorously, a multidimensional single-valued function can be represented mathematically as $\mathbf{y = f(X)}$ defined over a hyper-rectangular domain. That is, the argument $\mathbf{X} \in \mathbf{R^n}$ satisfies $\mathbf{2n}$ inequalities defined for all of its $\mathbf{n}$ components: $X_i^{lo} <= X_i <= X_i^{hi}$, $i = 1..n$. The lower limit values X_i^{lo} and the upper limit values X_i^{hi} define vertices of the hyper rectangle. A discretization grid is defined on the hyper rectangle, with each dimension discretized uniformly. Each of the dimensions is divided into N_i intervals of length $\Delta_i = (X_i^{hi} - X_i^{lo}) / N_i$ resulting in N_i+1 discretization points $X_i[K_i] = X_i^{lo} + K_i \bullet \Delta_i$. The endpoints of the interval are $X_i[0] = X_i^{lo}$ and $X_i[N_i] = X_i^{hi}$. Conceptually, all discretization points can be organized into a n-dimensional array $X[K_1][K_2]\ldots[K_n]$. The values of the function are known at the discretization grid points. They form a n-dimensional array of scalar values of the function, $Y[K_1][K_2]\ldots[K_n] = f(X[K_1][K_2]\ldots[K_n])$. In general, monotone increasing or decreasing behavior of the function can be assumed between any two adjacent grid points.

An example application where such data is used is the field of engineering design. A set of input system parameters determines a set of system performances. The designer needs to analyze the patterns between the system performances and input parameters to make valid conclusions and arrive at the right design decisions [6]. Such a system parameter space can be viewed as an n-dimensional bounded function. An example of such data is system error measurements, which typically result from multiple system parameters. Often, the parameter space is systematically sampled on the discretization grid by simulations. The multi-dimensional aspect of the input space makes it difficult to understand where and why the system performs poorly and to generate theories or rules of thumb for improving it.

It is simple to visualize and understand such data when it is restricted to 2 dimensions. The data can be graphically represented as a colored 2D plane or as a height field. At 3 dimensions it becomes more difficult, perhaps represented as a colored 3D volume. Greater than 3 dimensions becomes very difficult. In a 3 dimensional world, it is hard to conceptualize what 5, 6, or 7 dimensional space "looks" like, making it difficult to design visualizations of such a space. Higher dimensionality data can be represented easily by visualizing it with respect to 2 or 3 dimensions at a time. However, this does not help in visualizing the data with the effect of all dimensions simultaneously. We present a method of visualization that eliminates this problem of loss of dimensionality.

Related Work in Multi-Dimensional Visualization

For visualizing multi-dimensional functions, *Worlds within worlds* [2] and Mihalisin's work [4] use a technique of nested axes. Mihalisin uses 1D and 2D representations, while Worlds uses 3D representations. Worlds represents multiple dimensions by nesting one co-ordinate system within another. Hence, the axes are overloaded. Other tools use a more explicit form of interactive slicing to view 2D planar sections of the data space [5]. The major disadvantages of these approaches are lack of overview context, occlusion (in the 3D case). The nesting and slicing make it difficult to visualize the data trends across all dimensions uniformly. Trends can only be viewed on 2 or 3 dimensions at a time, and never an overview of the trend over all dimensions.

Approaches for multi-dimensional data (not function) visualization such as [1][3][6][7] are not well suited to visualize fully sampled spaces.

SOLUTION

We have designed a new interaction metaphor, called "focal point with a view" that uses the **focus+context** approach to visualizing data. The main idea behind this concept is to concentrate greater detail on the data near the focal point and give less detail to the data away from the focal point. This concept can be seen in **figure 1** where all the data points near the focal point (0,0) are shown but point like (2,3) is not shown because it is further away from the focal point. In addition, our solution represents all dimensions simultaneously without breaking dimensions apart.

The solution provides a way of visualizing the data with respect to a focal point. Rays directed outward from the focal point represent the points neighboring this focal point in the parameter space. Each ray is given a name depending upon the relation between the sampled data points that the ray represents and the focal point. The focal point lies somewhere in the bounded n-dimensional data space. All data patterns on rays from this focal point that proceed outward are visualized.

The focal point is represented in our visualization as a horizontal line called the focal line. The rays that proceed outward from this focal point in the data space are arranged vertically on the focal line in our visualization with the data points color-coded based on their Y-values. By stretching the focal point into a line in our representation, we increase the amount of area available to represent the focus area of the data, at some expense to the context area. This is based on the focus+context approach.

The rays can be prioritized according to a particular dimension. This prioritization arranges the rays in such a manner that the values of the prioritized dimension increase as we move down from the focal line and decrease as we move up from the focal line. Assigning priority to the dimension results in ordering the vertical rays such that as one moves from left to right along the screen the importance of that prioritized dimension decreases, at the same time the importance of the other dimensions increase.

We explain this concept in the simple case of a 2D parameter space where the data for both the dimensions is in the range [–3, 3].

Figure 1 shows an example of the visualization of a bounded 2D data space with the origin as the focal point and explains the concept of rays. The ray [0 1] represents the points whose value of the first dimension is constant and the second dimension increases with respect to the focal point. Other rays are also interpreted in the similar manner. The 0 in [0 1] represents a constant value, 1 represents an increasing value and a –1 represents a decreasing value of the corresponding dimension as we move away in the ray from the focal point. The colored dots represent the data points to be plotted on the respective rays and their color indicates some specific value of the function.

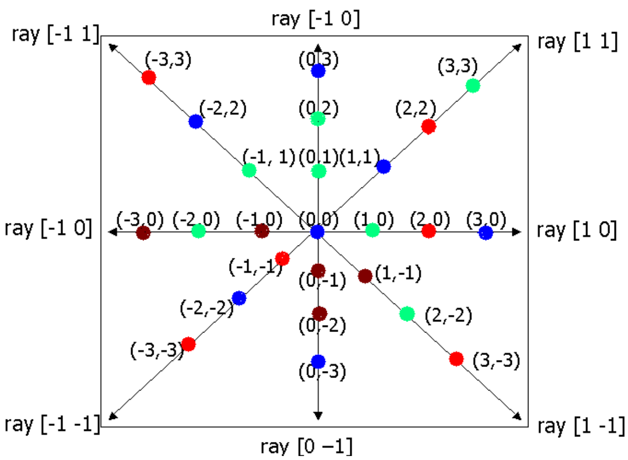

Figure 1 Origin as Focal Point in a 2D data space X1,X2.

Figure 2 expands the earlier concept of a Focal point and rays in bounded 2D space. The focal point in this case is not the origin but a point (2,2). This explains how we can change the focal point dynamically in the parameter space. The color of the data points represents their respective value in the legend.

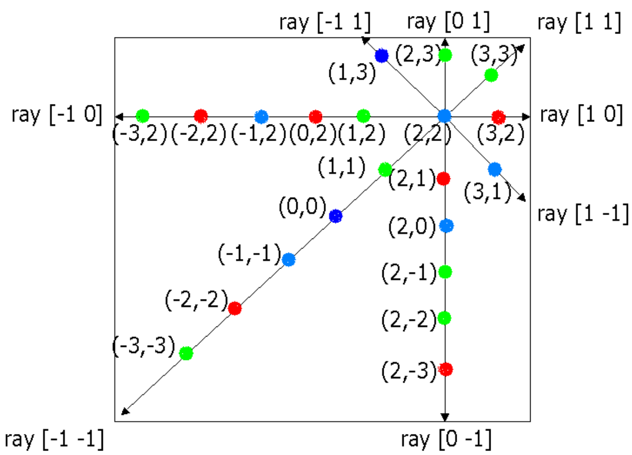

Figure 2 point (2,2) as the Focal Point in a 2D data space.

When the focal point is in the middle of the parameter space all the rays will represent equal number of data points (Refer **Figure 1**). As we move the focal point towards the edges of the parameter space, some rays increase in length and others decrease in length (see **Figure 2**).

The concept of rays emerging out of the focal point can be extended for a 3D image, by imagining a sphere made up of rays coming out from all directions from the center of the sphere where the center of the sphere is the focal point. This sphere is a 3D image bounded by a periphery. Our technique modifies this concept by flattening this 3D sphere and placing on a 2D plane. A similar flattening is performed in case of multi-dimensional functions with any number of dimensions.

Flattening such a function poses a problem of being able to arrange them in a manner such that it is possible to make out which ray from the multi-dimensional parameter space has been placed in which location in the 2D space. The solution to this problem is explained in **Figure 3** and **Figure 4.** The focal point is now represented a horizontal line called the Focal Line. All the rays emitting from the focal point are plotted as vertical rays emitting from this focal line. These rays are then ordered based on the priority of the dimensions. In this case X1 has greater priority than X2 so first ray has X1 values changing where as X2 values are remaining constant. Both these figures help in visualization of the concept of change in the ray length when the focal point moves closer to the boundaries of the parameter space.

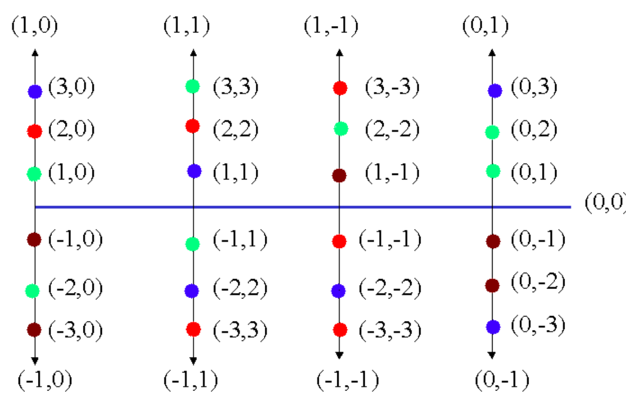

Figure 3: Visualization for data shown in Figure 1. Vertical rays have the priority of the dimensions for X1 has a greater priority over X2 in this case. The focal point is origin (0,0).

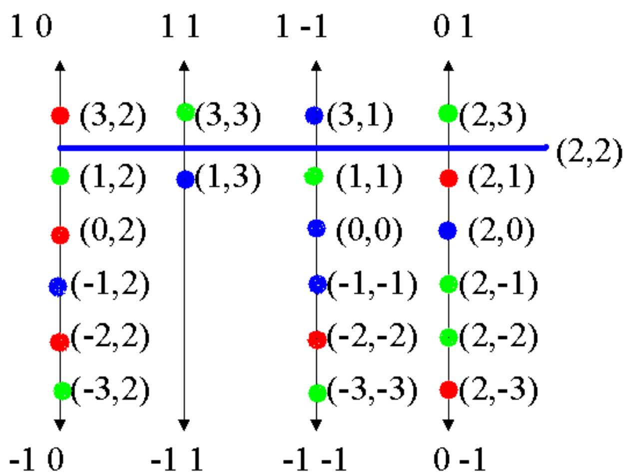

Figure 4 Visualization for data shown in Figure 2. The focal point is now (2,2).

NAVIGATION

In this visualization, the default focal point is the center of the bounded n-dimensional space (median of the range of the values for each dimensions). The focal point can be changed dynamically during the visualization. The function to be visualized is displayed at the top on the screen. (For simplicity we are showing mathematically defined functions. The original sampled engineering data provided by supporting company cannot be shown due to proprietary nature). The legend for the color codes of the Y values is shown at the bottom. Clicking on any point on the vertical rays makes this new point as a new focal point and gives the view of the entire data with respect to this Focal Point. After selecting a Focal Point, the dimensions can be prioritized to have different views of the data from that particular focal point. Right clicking on the screen and selecting the dimension from the drop down menu can prioritize a different dimension. A reset option on the drop down menu rests the visualization to the default focal point keeping the prioritized dimension as it is. This allows for improved exploration as the user can concentrate on one dimension and still visualize the effect of all other dimensions. Figure 5 represents the color legend used for the Y values. Each dot the screen represents one Y value depending on the color.

EXAMPLE

We have extended this concept to visualize a 5 dimensional parameter space where the value of each dimension ranges between [0, 40]. The example we discuss is a single valued function of five dimensions. The function used in this case study is

$$\mathbf{Y = X_1^2 + 5*X_2 - X_3 - X_4^2 + X_5}$$

This Y value is represented in the visualization using specific colors depending on the range in which it falls. **Figure 5** shows the legend for the colors.

Figure 5: The color legend for the visualization.

The initial visualization with the default Focal point is shown in **Figure 6**. The default focal point is the median of the range of the values of each of the dimensions, i.e. (20,20,20,20,20) in this case and the default prioritization is on dimension X1. The colors of the points represent the range in which the Y value belongs. We can see how the Y value changes as we move away from the focal point. **A**ll the points surrounding the focal point have the white and yellow color indicating that their Y values are in the range [0 200]. Some data points in the top corner of the visualization that show red and pink color signifies that these points have an extreme Y value in the range Y <= -750 and –750 < Y < 0.

Figure 7 shows this where one of those extreme points shown in red color in Figure 6 is now made the focal point by clicking on that point. The focal point in this case is (20,37,3,37,20) and the prioritized dimension is X4. This figure also shows the drop down menu that is used for selecting the dimension to be prioritized. It can be seen that the length of each of the rays in this case is different as opposed to the case where the focal point was in the middle of the parameter space. This effect is observed because some of the dimensions have values closer to the bounds of the parameter space. (Refer to the explanation for **Figure 2**).

Now lets see how this can be useful for analysis of some real time system that operates on multiple inputs and we have to visualize the effect of these inputs on the system output simultaneously. Suppose the Y value represents system output and if a negative Y value represents bad system performance. We have used the color red for extremely negative Y values and pink for slightly negative Y values. Further, we have used different colors to represent the range of positive Y values as shown in the legend in **Figure 5**. Now referring to **Figure 6** we can see that in order for the system to give the minimum acceptable performance we need to keep the input value somewhere closer to the default focal point which is (20,20,20,20,20) as lot of white and yellow color data points can be observed around this focal line. Since we know that the system will perform badly if the Y value is negative then we can navigate the visualization to see what inputs take us to a non-acceptable Y values. **Figure 7** shows that the system will perform very badly if some of the input dimensions have a value closer to the bounds of the parameter space represented by the focal point (20,37,3,37,20). Thus we can see how the system performs on different inputs by changing the focal points and visualizing the data with respect to those points. (refer **Figure 8**). This is very useful for a system analyst to have a control over the system by determining which inputs to give to make the system perform in the best possible manner. The concept of prioritization of rays will in turn be helpful if we need to find the effect of a particular input on the system output.

Thus we can navigate the high dimensional parameter space by changing the focal point. This navigation is useful in getting a detailed view of the space in regions with extreme function values. We can also use this for knowing the range of the dimension values that caused the generation of these extreme Y values.

ANALYSIS

This concept for visualizing multi-dimensional data is a generic one and can be extended to any number of dimensions. However, since the number of rays increase exponentially with the number of dimensions given by the formula $R = 3^N$, where R gives the number of rays and N gives the number of dimensions. It can be seen that the number of rays for a 5D parameter space is 243 in which half of them are directed vertically downwards and the other half of the rays are directed vertically upwards. In case of a 7D the number of rays is 2187. Hence, for a screen size of 1000 by 1000 pixels, we can show 1000 rays upwards and 1000 downwards, thus limiting the dimensionality to 7 with the current approach. The approach can be extended by eliminating some rays (some low priority combinations of dimensions) for very high dimensional data, and adding more rays for low dimensional data.

Another issue is the loss of data points. This results because we do not represent all the rays that exist in the parameter space. Considering the 2D example from Figure 1, we can see that we are representing only 8 rays and do not represent the other rays. For example, we represent a ray [1 1] (where both the dimensions increase at the same rate) but ignore the ray [2 1] (where the first dimension increases twice as fast as the second). This ray would fall somewhere in between the ray [1 1] and [0 1]. A potential solution is to use aggregation of data points onto rays. That is, all points can be represented by aggregating them into the nearest ray. This would be more consistent with the focus+context approach.

This loss of data points is proportional to the sampling rate and this is inversely proportional to the number of dimensions being visualized. This is because as the number of dimensions is reduced, the number of rays considered in each plane (8 in case of Figure 3) can be increased thus increasing the sampling rate. Hence, there is a tradeoff between the number of dimensions and the number of data points lost. This of course is caused because of the limited and constant screen space.

The main advantage of our approach is that we can visualize the multi-dimensional parameter space data with respect to a focal point without loss of dimensionality. This visualization of the simultaneous effect of all the dimensions on the data is very helpful for analyzing data.

Further, half the data points that are visualized belong to the neighborhood of the focal point and hence are relatively more important as compared to the other points that are far away in the parameter space.

CONCLUSION AND FUTURE WORK

We believe that the *focal point with a view* approach holds great promise for the visualization of multi-dimensional functions. Our approach scales up well to 7 dimensions, but can be modified to scale very high by eliminating low priority rays.

Further research needs to be done to prevent the loss of data points. A methodology of aggregation of lost data points with the currently represented points can be done to see the effect of these points in the parameter space. In the future, we would like to add the ability to enable viewing the data from multiple focal points in multiple frames. In addition, the ability to allow the allocation of the similar priority to more than one dimension at the same time is desired. These two features will allow more complete analysis of the data. Another feature to be added is to label the primary rays and show the exact values data values on moving the mouse over a particular point in the visualization. It is also desirable to have an option where the user can manually enter a desired focal point. Finally, we need to implement a gray scale coloring scheme to show continuous gradients of data along rays.

REFERENCES

[1] Ahlberg, C., Shneiderman, B., "Visual Information Seeking: Tight Coupling of Dynamic Query Filters with Starfield Displays", *Proc. ACM CHI'94 Conference - Human Factors in Computing Systems*, pp. 313-317, (1994).

[2] Feiner, S., and Beshers, C. Worlds within Worlds: Metaphors for Exploring n-Dimensional Virtual Worlds. *Proceedings USIT'90*, Pages 76 – 83.

[3] Inselberg, A. Multidimensional Detective. *Proceedings* IEEE *Information Visulaization'97,* Pages 100 – 107.

[4] Mihalisin, T., Timlin, J., and Schegeler, J.Visualizing Multivariate Functions, Data, and Distributions.IEEE *Computer Graphics and Applications, 11*(13), Pages 28 – 35.

[5] North, Shneiderman, Plaisant. User Controlled Overviews of an Image Library: A Case Study of the Visible Human. *Proc. ACM Digital Libraries '96 Conference*, ACM Press, pg. 74-82, (1996).

[6] Tweedie, L., Spence, R., Dawkes, H., and Su, H. Externalizing Abstract Mathematical Models. *Proceedings of CHI'96,* Pages 406 – 412.

[7] Kandogan, Star Coordinates, *Proc. Symposium Information visualization* 2000.

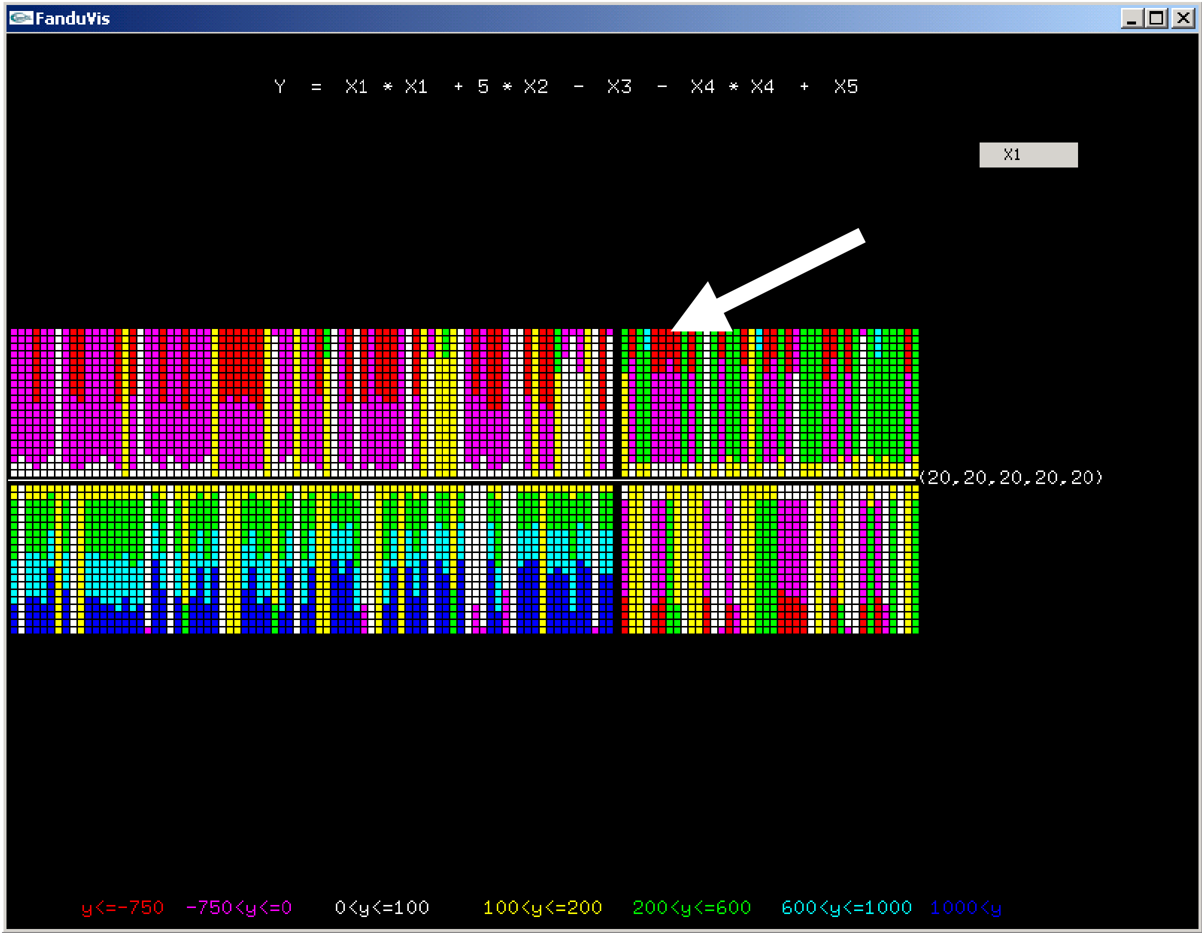

Figure 6 View of 5D data with the default focal point at (20,20,20,20,20) that is in the middle of the parameter space with each dimension ranging from [0, 40]. Clicking on third red point below the white arrow will display figure 7.

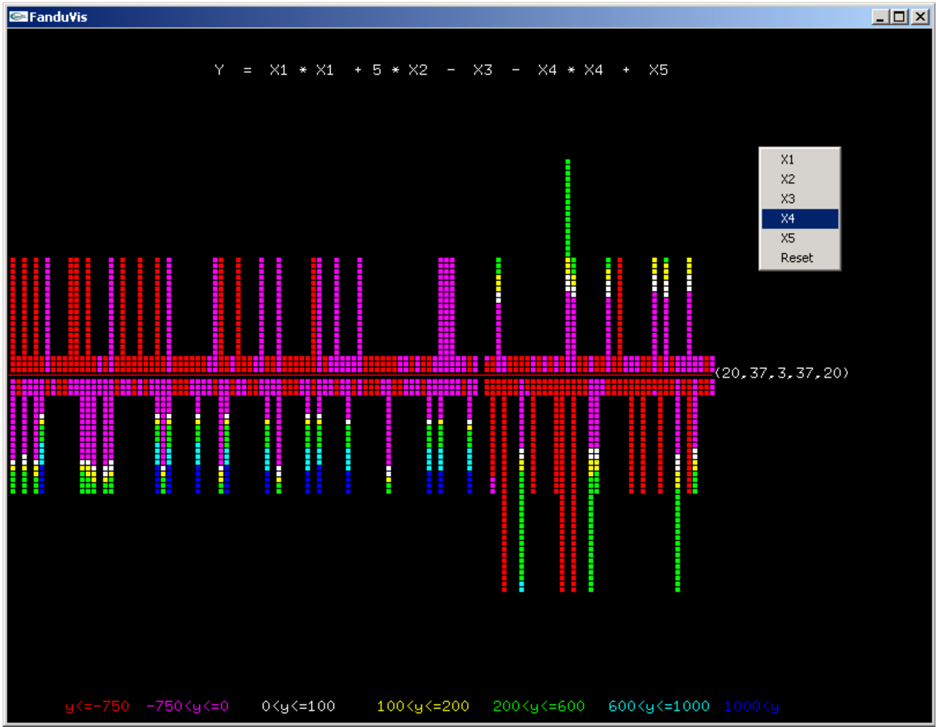

Figure 7 View with a new focal point at (20,37,3,37,20) (ref. arrow in fig. 6) with dimension X4 prioritized showing the effect on the length of rays with a focal point towards the edges in the parameter space. The focal point here has an extreme value shown by the red color of the focal line. Also seen is the drop down menu that is used to change the priority of dimensions and to reset the focal point to the median value.

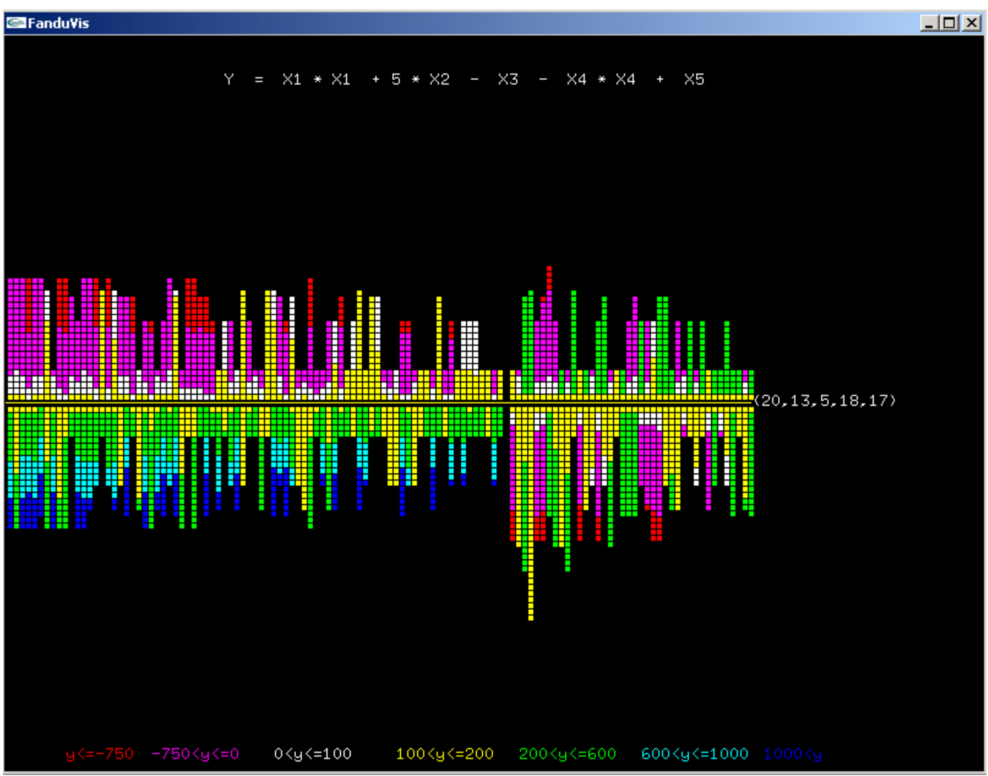

Figure 8 Effect of changing focal point towards the extremes, (20,13,5,18,17) in this case.