# CS 867: Displaying Multiple Time Series

Carmen St. Jean

December 16, 2013

## 0 Abstract

Displaying many time series at once in an understandable manner can be difficult, especially if the total screen space available is limited. We introduce a new technique called the stacked graph, where the time series are horizontally aligned, but each time series starts at one-fourth up the height of the previous series. Combined with semi-transparent colors and textures, this makes for time series which are easily distinguished from each other and, more importantly, have more vertical resolution without increasing overall screen space. Our evaluation compared this technique with small multiples and horizon graphs, which showed that the stacked graph requires more research before being recommended for use.

## 1 Introduction

Time-oriented data might be one of the most common forms of data, used in numerous fields such as medicine, economics, ecology, engineering, and meteorology. Typically, time series are visualized using line charts, which appear frequently in newspapers, textbooks, and academic papers. However, it can be difficult to display many concurrent time series at once in an understandable manner using a line chart. Therefore, we have introduced a new visualization technique called a stacked graph, which we propose may be suitable for showing many time series—i.e., fifteen—at once. To test the merits of our creation, we have evaluated the stacked graph against two existing methods—small multiples and horizon graphs.

### 1.1 Background

Since it was first introduced by William Playfair in 1786 [3], the line chart has been a popular choice for displaying temporal data. However, it has its limitations when it comes to multiple time series. To be plotted on a single chart, each time series must be distinctly colored and/or patterned; as the number of time series increases, it becomes more difficult to make each series
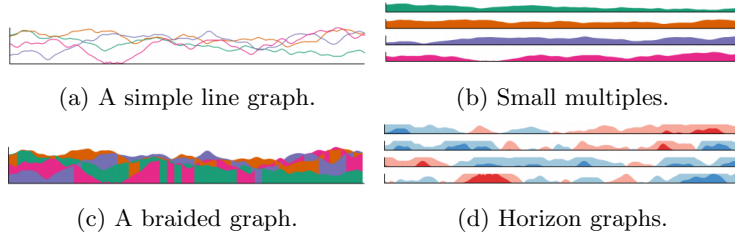
(a) A simple line graph.

(b) Small multiples.

(c) A braided graph.

(d) Horizon graphs.

Figure 1: Four possible methods for visualizing multiple time series [2].



(a) The line chart is divided into evenly-spaced bands, which are colored shades of blue in the upper-half and shades of red in the lower-half.



(b) The bands of similar colors are layered together.
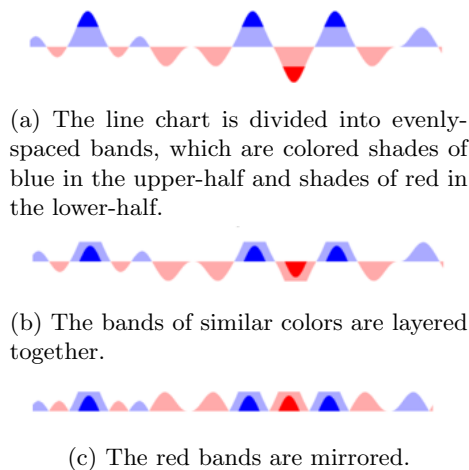


(c) The red bands are mirrored.

Figure 2: Steps for constructing a horizon graph [1].

visually distinct. Even if each time series is designed uniquely, it may be difficult to distinguish two time series which have similar values and overlap frequently.

Recognizing the limitations of the line chart for many time series leads to a question: what is a suitable alternative? Javed et al. set out to answer this question with their evaluation that compared the line chart, as in Figure 1a, with three alternative techniques [2]. The first alternative, called "small multiples" simply places each time series on its own line chart, shown in Figure 1b. The screen space must be split with this approach, which leads to less resolution in the vertical direction. Therefore, Javed et al. introduced a shared space approach called the "braided graph" seen in Figure 1c, which attempts to be easier to read than a line chart by coloring the areas underneath the curves. Additionally, Saito et al.'s horizon graph, as in Figure 1d, was included in the evaluation [4].

Saito et al. designed horizon graphs to increase the amount of data that can be displayed without increasing graph size [4]. To construct a horizon graph, some "mid-line" must be defined; it could represent zero or it could be the average of the global minimum and global maximum values in the dataset. The

areas between the mid-line and the data values are filled in, as in Figure 2a, with blue for areas above the mid-line and red for areas below the mid-line. The data is then further divided into evenly spaced bands, with the color of the band darkening as distance from the mid-line increases. Next, these bands are offset to align with the mid-line, as in Figure 2b. Finally, the red bands are mirrored. The result is the vertical resolution can be increased by a factor of four, as in Figure 2, or even more if more bands are used.

Javed et al. evaluated these four techniques illustrated in Figure 1 by having participants complete three tasks on randomly-generated data:

1. **Maximum:** for a given time point, identify the time series with the highest value.

2. **Slope:** for the entire time period, find the time series with the highest increase.

3. **Discrimination:** at a time point specific to each series, determine which series has the highest value.

In addition to chart type, their study varied the number of time series (2, 4, and 8) and the total chart height (48, 96, and 192 pixels). They found that the simple line chart and the braided graph were best suited for the maximum task, while the small multiples and horizon graphs were better for the slope and discrimination tasks. In general, they concluded that simple line charts and small multiples were generally more robust than horizon graphs and their newly introduced braided graphs. Additionally, they found that decreases in chart size did not affect completion time, but did decrease accuracy. Lastly, as the number of concurrent time series increased, accuracy decreased and completion time increased.

## 2    Method

Javed et al.'s approach considered only eight concurrent time series at most, when it is not unusual for many more to appear together in practice. Therefore, we determined to conduct a similar study but with fifteen concurrent time series. This number of time series makes simple line charts impractical, so they were not included in our evaluation. Additionally, we decided to exclude braided graphs because they did not prove to be particularly effective in Javed et al.'s evaluation. Instead, we have introduced stacked graphs, illustrated in Figure 3, to compare with small multiples and line graphs, using data generated from a random walk. We hypothesize that the stacked graph will prove to be a strong alternative for displaying many time series.

### 2.1    Design

Stacked graphs are called so because each time series is vertically-offset from the preceding time series, resulting in a stacked appearance. An $x$-axis and a
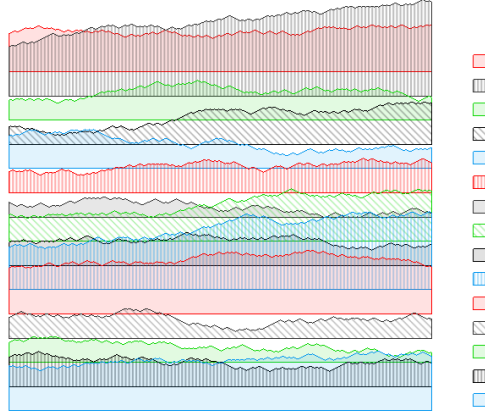
3

Figure 3: Fifteen time series represented with stacked graphs.

$y$-axis are drawn for each chart, though the latter is aligned for all charts. We chose to use an offset of one-fourth of the individual chart height, which means that, at any point, a chart may overlap three other charts. Therefore, we chose to fill the areas under each time series with semi-transparent, alternating colors and textures that were designed to have maximum see-through. Our textures are solid, striped, and slanted stripes, while our colors are red, green, blue, and black. The purpose of the stacked graph is to increase the amount of space each chart occupies without increasing the overall space that all time series occupy.
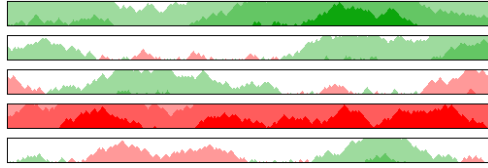


Figure 4: Five horizon graphs using green for values above the mid-line.

We have also proposed a different color scheme for the horizon graph, shown in Figure 4, because it can be difficult to remember the meaning of the traditional colors. If one thinks of a temperature color scale, then red is associated with higher values and blue with lower values, which is the opposite of their intended meanings for horizon graphs. Users already require instruction and practice to interpret the bands properly, so it would be helpful for them if the colors were less ambiguous. Therefore, we have replaced blue with green, shown in Figure 4, so that perhaps users will be reminded of the usual financial meanings for these colors: green for above some threshold and red for below it.

The tasks from Javed et al.'s study where used in our experiment, except for the discrimination task [2]. This was omitted because it seemed unrelated to how time series are read in reality. Typically, users are concerned with how

time series compare at the same point in time or across the same time range. The discrimination task asked users to compare time series at different times, which we felt was unusual.

## 2.2 Display

All experimentation was conducted on a standard Dell desktop computer with a 20-inch monitor set to 1280 by 1024 pixels resolution. The portion of the display where the time series were drawn occupied 500 by 515 pixels (approximately 5 by 5.25 inches); the rest of the display was colored white. Participants sat so their eyes were approximately 18 inches from the screen.

## 2.3 Procedure

The evaluation consisted of blocks of ten trials, one for each task with each visualization type. This made for a total of 60 trials per participant. A trial showed fifteen time series, from which the user selected an answer. Each time series had a small box drawn next to it, as shown in Figure 3, which the participant clicked to submit a response. Response times and correctness of response were recorded.

## 2.4 Tasks

Both tasks involved selecting one time series out of a set of fifteen. The two tasks used were based on the tasks by Javed et al. [2]:

1. **Maximum:** We asked the user to identify the time series with the highest value at a given point in time that was marked on all time series. The time was randomly chosen for each trial so it was no closer than 10% from the beginning or end of the chart.

2. **Slope:** Two times were marked on all time series. The user was instructed to identify the time series which increased the most from the first marked time to the second marked time. As with the maximum task, these times were randomly chosen so that they were no closer than 10% from the beginning or the end of the chart. Furthermore, the distance between the two times was designed to be at least 25% the width of the chart.

## 2.5 Conditions

The only condition that varied in our study was the visualization type. Our fifteen time series were represented in one of the following ways:

- **Small multiples:** one line chart per time series, with the area between the values and the $x$-axis filled. The same $y$-axis scale was used across all charts. Each chart was 500 by 26 pixels.

|  | Task | |
|---|---|---|
|  | Maximum | Slope |
| Small Multiples | 0.15 | 0.5 |
| Horizon Graphs | 0.125 | 0.625 |
| Stacked Chart | 0.65 | 0.7 |

|  | Task | |
|---|---|---|
|  | Maximum | Slope |
| Small Multiples | 6.58 | 15.41 |
| Horizon Graphs | 6.67 | 21.18 |
| Stacked Chart | 19.98 | 20.57 |

(a) The percent error (out of 1.0).        (b) Mean response time (seconds).

Figure 5: The results of the evaluation.

- **Horizon graphs:** one three-band horizon graph per time series using the red-green color scheme. The baseline was set to the average of the global minimum and global maximum values of all time series. The same $y$-axis scale was used across all charts. Each chart was 500 by 26 pixels.

- **Stacked graphs:** one semi-transparent line chart per time series, with the area between the values and the $x$-axis filled with a texture. Each chart starts at one-fourth up the height of the previous chart. The same $y$-axis scale was used across all charts. Each chart was 500 by 115 pixels.

## 2.6   Subjects

Four volunteers were used as subjects. All four were graduate students in Computer Science at the University of New Hampshire. Their vision was either 20-20 vision or corrected to 20-20 with glasses.

## 3   Results

The results of our evaluation are shown in tabular form in Figure 5 and as charts in Figure 6. We have recorded and calculated the mean response time and percent error; smaller values are desired for both. The results show that the slope task seems to have been much more difficult than the maximum task for all three chart types since the times approximately doubled from the maximum task. Small multiples seem to have been slightly better suited to the slope task, but the mean response times and errors were fairly similar across the different chart types. Horizon graphs led to slightly lower errors than small multiples for the maximum task, though they were fairly equivalent and both were much better than stack graphs in terms of time and error. For both tasks, the stacked charts proved to have high errors and high response times.

## 4   Discussion

There were only slight differences between the chart types for the slope task, but the stacked graph proved to be inefficient for the maximum task, attaining an error—0.65—over four times as a high as the other chart types. Furthermore,

(a) The percent error (out of 1.0).  (b) Mean response time (seconds).
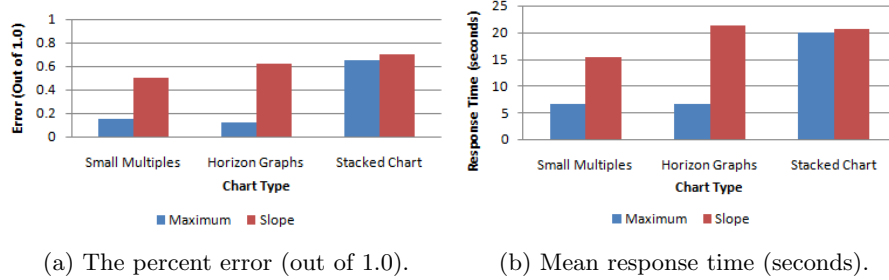
Figure 6: The results of the evaluation graphed.

response times were approximately three times higher for the stacked graph with the maximum task, meaning participants took longer to reach the wrong answer for the stacked graph. This matches with the commentary provided by the different participants following completion of the evaluation; the stacked graphs were described as "confusing" and "a mess."

These results seem to show that the stacked graph is inferior to the horizon graph and small multiples and should, therefore, be avoided when plotting many concurrent time series. While this may be true, it could also be possible that our implementation was suboptimal. In the design phase, we incrementally attempted to improve the colors and textures used for the stacked graphs. However, there may be other color and texture combinations that would lead to a better performance. Also, it is possible that a different vertical offset could lead to better results.

There were major differences between the two tasks—maximum task and slope task—for the small multiples and horizon graphs. Both chart types had reasonably low errors and response times for the maximum task, but much higher errors and times for the slope task. There could be a number of possible explanations for this—e.g., none of these chart types are suited to this task, the task is too complicated with fifteen time series at once, or the task was improperly explained to participants. More research would be required to determine the reason for the overall difficulty with the slope task.

## 4.1   Conclusion

Our study disproved our hypothesis unfortunately; for these tasks, the stacked graph seemed to be inefficient and, worse, misleading for representing many time series. Adjustments to the coloring and texturing of the stacked graph may lead to improvements, though these changes should be evaluated before being used in practice. Small multiples or horizon graphs were much better for the two tasks involved and seemed to be fairly equivalent to each other. When representing several concurrent time series, we recommend using either small multiples or horizon graphs. To chose between the two chart types, one may follow Javed et al.'s recommendations, which found small multiples to be more

robust than horizon graphs [2].

## 4.2 Future Work

There are many different combinations of color and texture possible for use with a stacked graph, some of which might lead to improvements. Furthermore, additional techniques such as fading opacity could be used to enhance the stacked graph's appearance.

Our idea to use a red-green color scheme for the horizon graph leads to the question: what color scheme is best for the understandability of the horizon graph? Previous evaluations on the horizon graph have focused on the overall size of the chart or the number of overlapping bands; perhaps it might be useful to evaluate different color schemes for the horizon graph.

# 5 References

[1] Jeffrey Heer, Nicholas Kong, and Maneesh Agrawala. Sizing the horizon: The effects of chart size and layering on the graphical perception of time series visualizations. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '09, pages 1303–1312, New York, NY, USA, 2009. ACM.

[2] Waqas Javed, Bryan McDonnel, and Niklas Elmqvist. Graphical perception of multiple time series. *IEEE Transactions on Visualization and Computer Graphics*, 16(6):927–934, November 2010.

[3] William Playfair. *The Commerical and Political Atlas: Representing, by means of stained copper-plate charts, the progress of the commerce, revenues, expenditure and debts of England during the whole of the Eighteenth Century.* 1786.

[4] Takafumi Saito, Hiroko Nakamura Miyamura, Mitsuyoshi Yamamoto, Hiroki Saito, Yuka Hoshiya, and Takumi Kaseda. Two-tone pseudo coloring: Compact visualization for one-dimensional data. *IEEE Symposium on Information Visualization*, 0:23, 2005.