High Level Internet Scale Traffic Visualization Using Hilbert Curve Mapping

B. Irwin and N. Pilkington

Abstract A high level analysis tool was developed for aiding in the analysis of large volumes of network telescope traffic, and in particular the comparisons of data collected from multiple telescope sources. Providing a visual means for the evaluation of worm propagation algorithms has also been achieved. By using a Hilbert curve as a means of ordering points within the visualization space, the concept of nearness between numerically sequential network blocks was preserved. The design premise and initial results obtained using the tool developed are discussed, and a number of future extensions proposed.

1 Introduction

This paper describes the tool developed for providing high level Internet scale visualization of network traffic. The development of the tool was prompted by the need to have a tool which was able to display large volumes of IP traffic collected at network telescopes, while being be able to communicate some kind of semantic and sequential relationship between the nodes representing networks displayed on the resultant plot. This work was to a large extent inspired by the work done by Randall Munroe, author of the xkcd.com web comic who published a "Map of the Internet" (Munroe, 2006a, b) based on aggregation of IP space by class A (/8 bit) in December 2006. This work was based on an interpretation of the IP address to Hilbert curve mapping algorithm that was used in his work, which was then extended for producing higher order and hence more fine grained curves. The real value of the application of the Hilbert mapping is that it preserves the locality of adjacent netblocks when the one-dimensional numerical ordering of octets is rendered to a two dimensional grid.

B. Irwin and N. Pilkington
Department of Computer Science, Rhodes University, Grahamstown, South Africa, e-mail: b.irwin@ru.ac.za, nicholas.pilkington@gmail.com

The pure information security aspects of this tool are not focused on in this paper, but rather the value of the technique for plotting IP Network data. Using this layout mechanism as a basis any number of colouring or other additional attributes can be added as an augmentation.

The remainder of the paper is organized as follows. A brief introduction to the Hilbert curve is presented as a prelude to the details of the implementation of the tool. The bulk of the paper is focused on sample output obtained and the interpretation thereof. Conclusions and reflection on planned extensions conclude the paper.

2 Related Work

The Hilbert curve is a continuous fractal curve, the limit of which fills a square. This was conceived by German mathematician David Hilbert in 1891 (Riemersma, 2006; Weisstein, 2007) and falls into a larger family of space filling curves including the Peano curve family. The Hilbert curve can be used to extrapolate data from one dimension into two dimensions while still maintaining properties of the original one dimensional data – particularly the notion of ordering and closeness to sequential nodes within the sequence. A Hilbert curve maintains locality of data on the curve. This means that data ordered a certain way in one dimension will still be ordered the same way along the curve in two dimensions. Another interesting property of the curve is that it visits every lattice point in a square with side length a power of two. This is especially useful in extrapolating data which occurs in powers of two, onto a plane.

The process of generating a Hilbert curve fractal can be formally expressed with a Lindenmayer production system as shown in Fig. 1.

A somewhat less formal, and possibly more intuitive, description of the operation of the curve can be presented graphically. The Hilbert curve is comprised of what are known as “cups” and “joins”. Cups are squares with one side open, and joins are straight lines that connect two cups. Each can be orientated in any way as long as they remain parallel to the cardinal axes. The simplest Hilbert curve is the first order one which fills a 2×2 square grid and it just a cup with the top side open (Fig. 2a). In order to generate the next (second) order curve – one replaces the existing cup

$$
\begin{aligned}
&Alphabet: L, R\\
&Constants: F, +, -\\
&Axiom: L\\
&Production\ rules:\\
&L \rightarrow +RF - LFL - FR +\\
&R \rightarrow -LF + RFR + FL -
\end{aligned}
$$

Fig. 1 Lindenmayer representation of the Hilbert curve

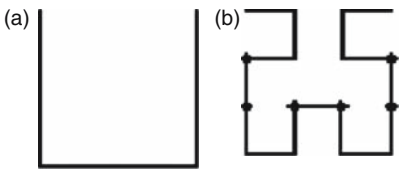

Fig. 2 (**a**) Order 1 Hilbert curve. (**b**) Second order Hilbert curve showing cups

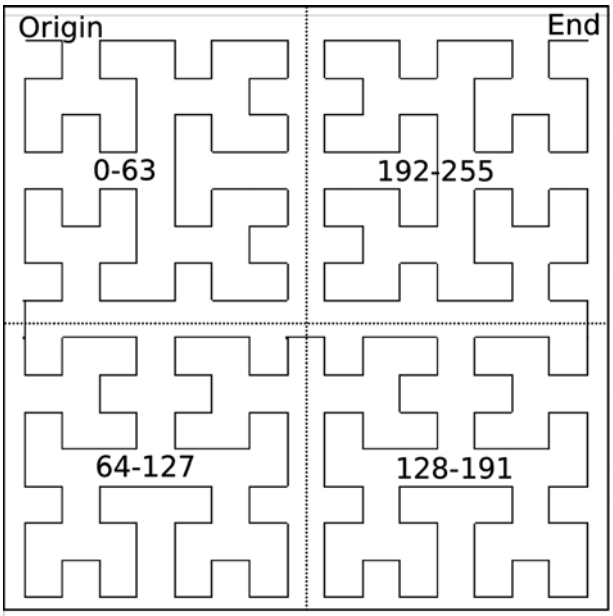

Fig. 3 Fourth order Hilbert curve showing plotting path for class A (/8) network blocks. 0/8 is mapped to the top left, and 255/8 to the top right corners, respectively

with four other cups connected together with joins as shown in Fig. 2b below. This process can then be repeated in the same way to generate all higher order curves as needed.

The Hilbert curves of order 4, 8, 12, and 16 are especially interesting as they have 256, 65536, 16,777,216 and 4,294,967,296 points, respectively. These values correspond to the natural grouping of Internet networks blocks by Class A (/8), Class B (/16), and Class C (/24) with the range of 32-bit Internet address space. A 16th order curve provides the same number of points as 2^{32} which is the same as the total potential number of addressable nodes on the IP protocol version 4 (IPv4) Internet.

Figure 3 shows the layout used for plotting class A network blocks using a fourth order curve. The curve starts with 0.0.0.0/8 mapped to the top left, and 255.0.0.0/8 to the top right corners, respectively. The grid can be divided into quadrants showing the placement of the 256 network blocks, which can be used as a guide when interpreting output such as that shown in the next sections. Raising the order of the

curve for /16 and /24 networks used the same layout, but increases the tightness of the curve within the respective quadrants.

It is worth noting that a similar pixel based plotting approach was proposed by Teoh et al. (2002), which uses a quadrant based mapping scheme based on the most significant bits of an IP address. The Hilbert plotting scheme has a more natural binning effect (to those habituated to dealing with traditional "classful" subnets) than the aforementioned method.

Two other projects are making use of the same Hilbery Layout for performing similar work (in both cases also inspired by Munroe's work). The first is the ANT Censuses of the Internet Address Space (Heidemann and Pradkin, 2007b), using the curve for plotting Address space usage as part of the Internet Census project (Heidemann et al., 2007a). The second related project is that being run by Measurement Factory using this to vizualise BGP route data (Wessels and Claffy, 2007). In many ways similar to the original goals and purpose of the system evaluated in Teoh et al.

3 Technical Approach

A proof of concept implementation of the Hilbert curve based layout system was developed in C++ and OpenGL. While development and testing was on the Microsoft Windows® platform, it should be portable to any system POSIX and OpenGL compliant system. Other implementations have been found in php (Bosci, 2007), but follow a similar system. The base layout code was developed to map a given substring of a dotted-quad IP address representation to a particular point on the grid being produced. In effect this associated a bin or bucket holding IP networks of a given ordinarily (representing natural 8-bit divisions of the dotted-quad, and hence clustered by most significant octets) to nodes on the curve.

As previously discussed, curves of orders 4, 8 and 12 map easily to the natural netmasks of pre-CIDR class A, B and C network blocks. Hilbert curves of 16th order were not implemented due to the complexities of mapping these to a reasonable screen size, and memory constraints. While it is recognized that much Internet address space allocation today does make use of CIDR based variable length masks, for most purposes, the traditional masks are still useful. The net result is a graphical output showing nodes which are coloured as containing elements aggregated dependant on the order of the curve. Initially a true/false flagging system was used where a node was drawn if it contained at least one member in the aggregation bucket. This was extended to use colouring based on criteria such as unique hosts with the bucket, and total number of packets within the bucket.

A number of discrete implementations were produced, each focusing on separate aspects of the visualization, but all using the same underlying Hilbert curve generation for layout. Specifically versions were produced that provided colour-indexed quantifications of the number of unique hosts with in a specific network block bucket, and the number of packets received within a bucket.

In the program the vertices were generated recursively using the Lindenmayer System representation of the Hilbert – this was the simplest representation and allowed the curve to be generated quickly and accurately. Additionally, the use of this representation also generates the vertices of the curve in numerical order which allows points to be plotted at the same time the curve is generated thus providing much greater efficiency.

The IP addresses to be plotted were stored in a hash table that allowed the required IP address count associated with the current vertex on the Hilbert curve to be quickly recovered from the table and plotted.

Input to the system was by means of simple text files containing a single dotted-quad IP address per line. This format was chosen due to the simplicity to produce form a number of different data sources, such as *libpcap* format packet captures, databases (containing Intrusion detection data), and simulation software. The code could also be trivially extended to work directly with formats such as *libpcap*.

4 Results

Initial results presented by the tool have been promising with tests of up to 63 million individual addresses being rendered with relatively modest hardware requirements by modern measures. The test system used was a dual Xeon 2.2 Ghz with 4 Gigs of DDR2 RAM running Microsoft Windows Server 2003, and an Nvidia Gforce 5200.

Performance was found to remarkably good on the test system with the exception of the colour mapping based on total packet count which was found to be particularly resource intensive, requiring over 1,800 MB of RAM in which to store the state tables, and needing 60 min to render the 63 million addresses on an order 12 curve (equivalent to /24 network bins each holding 256 individual IP addresses). It is anticipated that this can be improved, as zooming and other navigation became near impossible at this level of performance.

The simple flagging and host count based colour mapping were found to perform, responsively, and required in the order of 300 MB of memory to hold the large 63 million address dataset. Smaller datasets required on average 100–150 MB of RAM. Sample images of the tool output should be interpreted bearing the layout show in Fig. 3 in mind as a guide to the relative addresses of nodes within the overall address space.

Generating an overview at class A level provides a quick overview of input data, and particularly allows for sanity checking that data is not being plotted in regions where it is not. Figures 4–6 show the plot of 2.4 million unique data points sampled from CAIDA network Telescope data between 12h00 and 17h00 EST on February 28th 2007 (Shannon et al., 2007). Figure 4 is the fourth order curve with the actual Hilbert path plotted as a guideline. The same data is shown in Figs. 5 and 6 but plotted onto an eighth order curve showing data bins with netmasks of /16 (thus holding 2^{16} IP addresses) and a 12th (/24 bins) order curves, respectively.

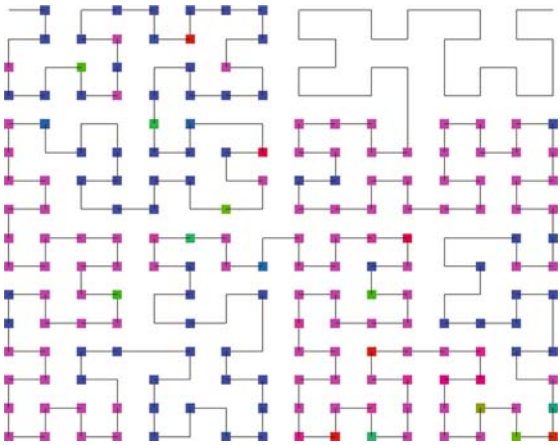

Fig. 4 Plot of the fourth order showing background curve. Colouring is based on number of distinct nodes within the bucket

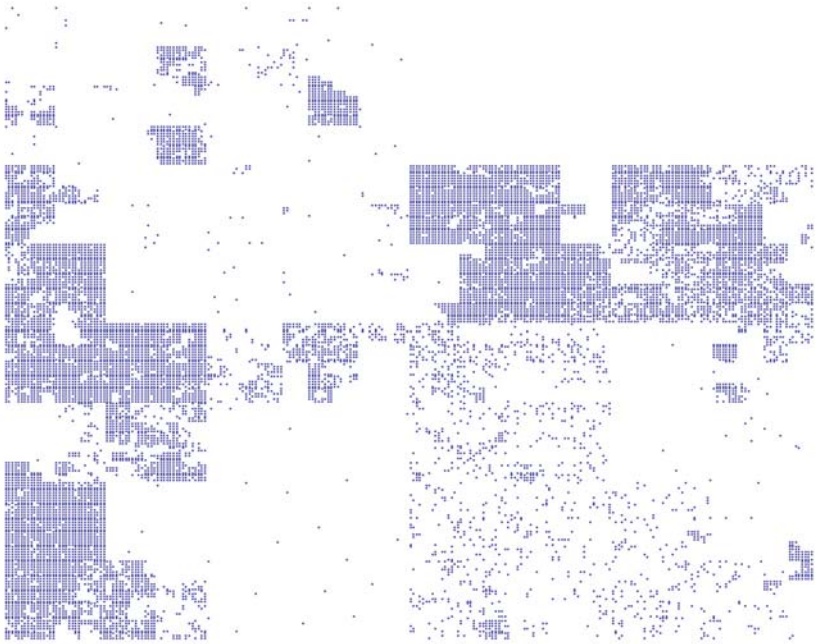

Fig. 5 Plot of eighth order representation of the data in Fig. 4, showing buckets corresponding to /16 networks (Class B)

The increase in granularity can be seen, particularly in the sequences show in Fig. 7 and discussed in the following section.

The above images illustrate the value of increased resolution, which can then be further analysed.

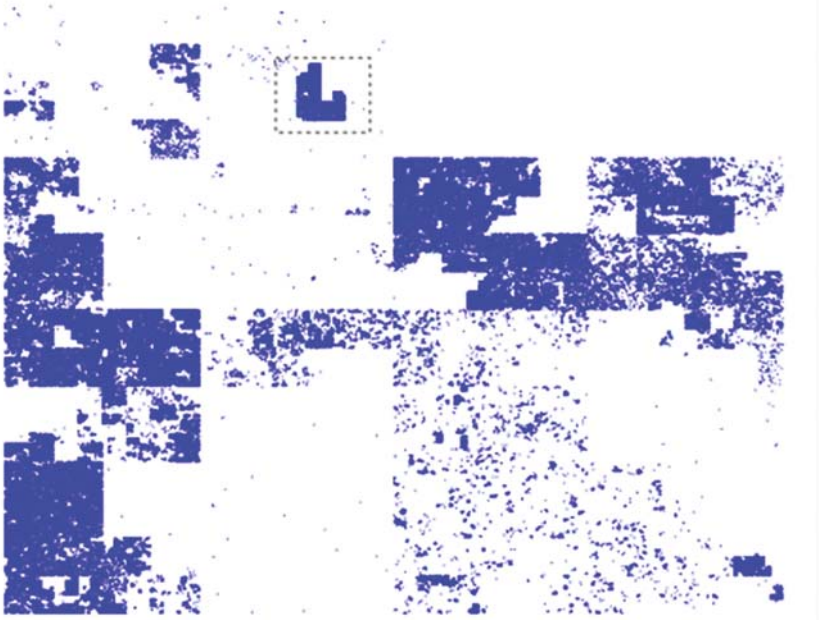

Fig. 6 Plot of data on a 12th order curve. Points correspond to buckets of size/24 (Class C). The boxed area relates to the zone of discussion in Fig. 7, and represents the network ranges from 23.232.0.0/13 to 24.170.0.0/16

Fig. 7 Close up of the view of the 23.232.0.0/13 and 24.0.0.0/8 networks using /16 buckets (A – left), and then with a finer resolution of /24 buckets in B (right) showing a distinctly different pattern. This area is shaded in Fig. 6

4.1 Output Analysis

Once traffic plots have been created, further analysis of the resultant images can be performed. Figure 7 shows close up of the view of the 23.232.0.0/13 and 24.0.0.0/8 networks using /16 buckets, and then with a finer resolution of /24 buckets in Fig. 7b showing a distinctly different pattern, with a number of networks having no traffic originating from them. These ranges are particularly interesting to analyse as the 24/8 network has been used widely for the provision of broadband cable access in the Americas, with large portions belonging to providers such as Comcast, Charter Communications and RoadRunner – major players in the provision of home broadband connectivity.

Portions of the netblock are also allocated to LACNIC where they appear to be sub-allocated to similar sorts of companies, but these do not show up as having been sources within the traffic set used. Data sources networks range from 23.323.0.0/16 to 24.170.0.0/16. This range is also indicated by a grey box on Fig. 6. Interestingly IANA lists the 23.0.0.0/8 netblock as currently reserved, so one can speculate as to the origins of this traffic – possibly misconfigured hosts as the range appears in a number of vendors documentation as example addresses.

One of the strengths of the plotting scheme developed is the aggregation of nearby networks as shown in Fig. 7. This allows for immediate visual recognition of the logical and special relationship between these nodes far more easily than some traditional grid based and wrapped linear plotting methods.

Analysis of the data can be performed either by means of working with the resultant image, or within the application which provides a rudimentary heads up display type information where feedback is provided as to the actual network address of the node over which the mouse cursor is placed.

Using the colour-index based mappings, as a means of shading graph nodes, further information relating to the plotted nodes can be conveyed. Figure 8 gives an example of 63 million events being blotted on a eighth order curve, with colouring indicating the number of unique hosts in the particular block that have originated traffic (or at least purported to based on IP datagram source addresses) to the network telescope. The graduation runs from green through blue orange and finally red. The insert within Fig. 8 shows a zoomed view of the 210.0.0.0–221.0.0.0 network ranges. The area of the image occupied by this insert is traditional Class D (224.0.0.0/4) and E (240.0.0.0/4) address space, which are, respectively, used for Multicast and reserved.

4.2 Other Applications

Following the initial work done with the curve the authors applied it to the visualisation of two other related problem areas. The first being to allow for a visual comparison and evaluation of the effectiveness of network telescopes of differing sizes. The results for this can be seen in Fig. 9, where the general shape of the plot produced when only using data from a single address still maintains the general shape of that produced when using data from the entire Class A telescope. This shows that smaller telescope still have some value, when used over long temporal baselines.

The second area in which it was applied was in evaluating the impact and effectiveness of differing network worm propagation algorithms. In the case study shown below the Hilbert curve is used to map snapshots of activity from a simulated Blaster worm at iterations 5,000, 7,500 and 10,000 within its propagation cycle. From these images it can be easily seen that while the propagation has covered 96% of /8 network bins (Fig. 10c), the coverage at /16 (Fig. 11) and particularly /24 sizing is very sparse.

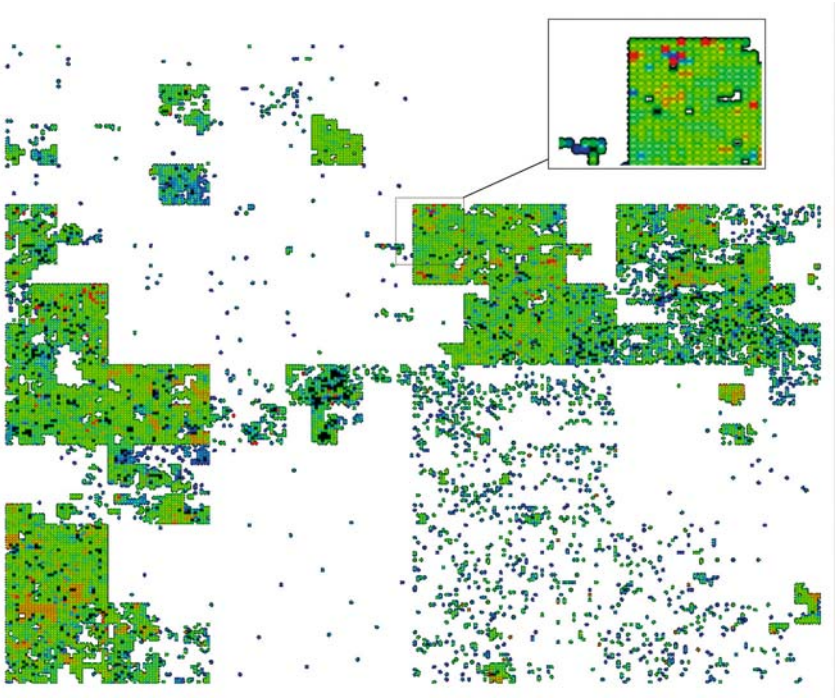

Fig. 8 63 million events plotted using host based colouring. Red dots show networks with the greatest number of unique hosts

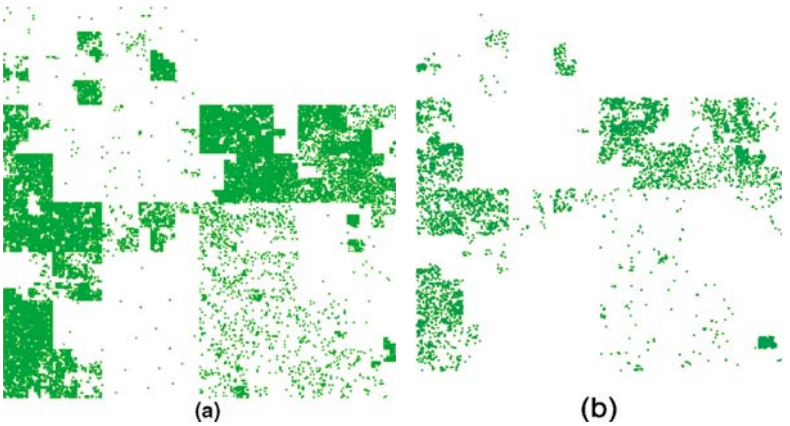

Fig. 9 Comparative mappings of (**a**) Class A telescope (**b**) data collected by a single IP address within the telescope

The ability to use images such as those presented above allows for researches to perform rapid visual evaluations and comparison, on the basis of which further more detailed statistical analysis may be performed.

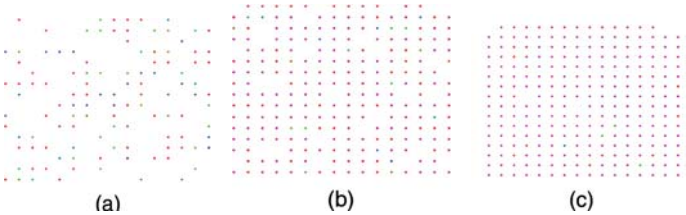

Fig. 10 Progressive coverage of IPv4 space by Blaster. /8 bit bins used for aggregation, and plotted on a fourth order curve. From left showing coverage at 5,000 (**a**), 7,500 (**b**) and 10,000 (**c**) iterations of the scanning and propagation algorithm

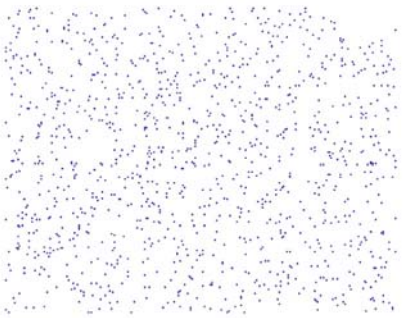

Fig. 11 Coverage of IPv4 space by Blaster after 10,000 iterations of propagation algorithm. Blotted using eighth order curve and bins of size /16. Despite near full coverage of the space with larger bins, the coverage is fairly low at this level of detail

5 Future Work

The authors have anticipated a number of extensions to the tool now that the value of the plotting scheme has been established. A more in-depth usability study needs to be completed, and aspects highlighted there will need to be addressed. From initial informal evaluation and operational use by the authors and researchers within the department, the following items are envisaged as being useful extensions and additions to the toolset:

- IPv6 support – with the move to IP version 6 the curve should scale, provided sufficient memory is available on the host system. Curves of order 32 and above would need to be used.
- Heads-up display capability – in order to provide for improved navigation of the curve, a HUD mode needs to be added, which could also provide further information relating to the specific nodes within the network.
- Overlay mode – allow network block of interest to be defined and input to the application which can then display a toggle-able highlight or shadow mask to show these when navigating. Network blocks defined as Bogons or reserved blocks are likely to be a useful set which to highlight traffic originating from.
- Time sequence views – to add accelerated and retarded time playback of *libpcap* files with the resultant output being captured as frame sequences or video

for analysis. This would illustrate changes in source traffic over time. Currently this has to be implemented through external processing, and only allows for snapshots.

- A Geographical mapping where colour mapping can be performed based on continent, country or regional registry. This may work best for highlighting specific countries rather than as a general mapping mechanism where nearly 200 colours would prove difficult to discern and differentiate, particularly at the /24 level.

The implementation of these features and a more detailed usability study is planned for the forthcoming academic year.

6 Conclusions

Since its initial inception the tool implementing the plotting scheme discussed has been used for detailed analysis of network telescope traffic, in particular to perform quick visual comparisons between data collected on the authors' own network telescope (comprising a single /24 network under the auspices of AFRINIC) and data provided by CAIDA.org (Shannon et al., 2007) from their large telescope located in the United States comprising a class A network – 65,535 times larger than that of the authors! The outputs generated using the Hilbert plotting scheme have shown that the same address ranges are being seen as the sources of both backscatter and malicious traffic, with minor variations between the two networks.

Another use to which the tool has been put is to visualise the output of simulated worm scanning activity in order to assess the effectiveness of various scanning algorithms and defensive measures such as strike back, and counter worms. The resultant images have allowed for quick visual evaluation of datasets ranging in the tens of millions discrete dataset members.

The tool developed has proved useful in providing high level overviews of large volumes of traffic while still maintaining the sequence order of the input data.

From a pure Information Security perspective this tool allows for easily interpreted visual summative reports to be generated from very large volumes of network traffic. While the proof of concept tool developed has noted shortcomings, the actual principle of using the Hilbert curve, or possibly other space filling fractal curves for plotting IP traffic data is a solution portable to other visualisation applications and problem spaces. The use of such visual reporting allows for easier interpretation of data at executive level, and in high workload environments.

Acknowledgements This work was performed in and funded by the Centre of Excellence in Distributed Multimedia at Rhodes University with financial support from Telkom SA, Business Connexion, Comverse, Verso Technologies, Tellabs, StorTech, Mars Technologies, Amatola Telecom, Bright Idea Projects 39 and THRIP. Randall Munroe of XKCD.com provided the initial inspiration for this work.

References

Bosci, H. Map-o-net.com image drawing code. May 2007. http://map-o-net.com/internet.html

Heidemann, J., Pradkin, Y., Govindan, R., Papadopoulos, C., and Bannister, J. Exploring Visible Internet Hosts through Census and Survey. Technical Report ISI-TR-2007-640, USC/Information Sciences Institute, May, 2007a

Heidemann, J. and Pradkin, Y. Mapping the Internet Address Space (poster). USC/Information Sciences Institute. http://www.isi.edu/ant/address/, August 2007b

Munroe, R. Map of the Internet. XKCD.COM http://xkcd.com/195/. http://imgs.xkcd.com/comics/map_of_the_internet.jpg, August 2006a

Munroe, R. Map of the Internet. 11 December 2006b. http://blag.xkcd.com/2006/12/11/the-map-of-the-internet/

Riemersma, T. The Hilbert Curve, August 2006. http://www.compuphase.com/hilbert.htm

Shannon, C., Moore, D., and Abden, E., The CAIDA Backscatter-2007 Dataset February 2007–November 2007. http://www.caida.org/data/passive/backscatter_2007_dataset.xml

Teoh, S.T., Ma, K.-L., Wu, S.F., and Zhao, X. Case study: interactive visualization for internet security. In Proceedings of VIS '02: Proceedings of the Conference on Visualization '02. IEEE Computer Society, 2002

Weisstein, E.W. Hilbert Curve. From MathWorld – A Wolfram Web Resource. http://mathworld.wolfram.com/HilbertCurve.html, 2007

Wessels, D. and Claffy, K., IPv4 Heatmaps: BGP Route Advertisements. http://maps.measurement-factory.com/gallery/Routeviews/2007