

Systems biology

AVIS: AJAX viewer of interactive signaling networks

Seth I. Berger, Ravi Iyengar and Avi Ma'ayan*

Department of Pharmacology and Systems Therapeutics, Mount Sinai School of Medicine, 1425 Madison Avenue, New York 10029, USA

Received on December 5, 2006; revised on July 18, 2007; accepted on August 21, 2007

Advance Access publication September 12, 2007

Associate Editor: Alfonso Valencia

ABSTRACT

Motivation: Increasing complexity of cell signaling network maps requires sophisticated visualization technologies. Simple web-based visualization tools can allow for improved data presentation and collaboration. Researchers studying cell signaling would benefit from having the ability to embed dynamic cell signaling maps in web pages.

Summary: AVIS is a Google gadget compatible web-based viewer of interactive cell signaling networks. AVIS is an implementation of AJAX (Asynchronous JavaScript with XML) with the usage of the libraries GraphViz, ImageMagic (PerlMagic) and overLib. AVIS provides web-based visualization of text-based signaling networks with dynamical zooming, panning and linking capabilities. AVIS is a cross-platform web-based tool that can be used to visualize network maps as embedded objects in any web page. AVIS was implemented for visualization of PathwayGenerator, a tool that displays over 4000 automatically generated mammalian cell signaling maps; NodeNeighborhood a tool to visualize first and second interacting neighbors of yeast and mammalian proteins; and for Genes2Networks, a tool to connect lists of genes and protein using background protein interaction networks.

Availability: A demo page of AVIS and links to applications and distributions can be found at <http://actin.pharm.mssm.edu/AVIS2>. Detailed instructions for using and configuring AVIS can be found in the user manual at <http://actin.pharm.mssm.edu/AVIS2/manual.pdf>

Contact: avi.maayan@mssm.edu

1 INTRODUCTION

Gradual increased intricacy of cell signaling pathways requires sophisticated representations of cell signaling maps. As such, several web-based tools such PATIKAWeb (Dogrusoz *et al.*, 2006), BioPP (Viswanathan *et al.*, 2007) and VisAnt (Hu *et al.*, 2005) have been developed to aid in cell signaling networks visualization and analysis. These visualization tools commonly use JAVA Web Start or JAVA Applet technologies that allow networks to be dynamically and interactively visualized in web-browsers. However, these JAVA-based tools do not allow signaling maps to be embedded inline within other webpage content and easily communicate with the browser, as would

be desired when presenting data for collaboration, or for dynamically displaying pathway diagrams from a database such as the connection maps database CMADES at Science Magazine's Signal Transduction Knowledge Environment (STKE) (Gough, 2002). There is some functionality for linking visualized network maps with other content on web pages and communication with the browser in VisAnt, but this requires the user to have JAVA Web Start properly configured (Hu *et al.*, 2005). Hence, there is a niche for a powerful yet simplified light-weight viewer. Requirements include compromise between details and overall viewing, minimal crossing of links, internet-readiness and compatibility with several protein-protein interactions data formats (Rojdestvenski, 2003). Most interaction maps used for visualization of intracellular signaling networks on the web are presented as either HTML embedded static images (Kitano *et al.*, 2005) or using Scalable Vector Graphics (SVG <http://www.w3.org/Graphics/SVG/>) (Gough, 2002; Li *et al.*, 2002). Both approaches can be made interactive; e.g. scaling smoothly across multiple zoom levels, panning by dragging the images up and down, as well as providing hyperlinks embedded in the map images as mouse-over and mouse-click events. Currently, most popular web browsers, such as Internet Explorer or Safari lack native support for SVG while other browsers like Firefox only recently added some basic SVG support. As such, viewing SVG interactive images requires third party plug-in software, such as Adobe's SVG viewer (<http://www.adobe.com/svg/>). Raster-based images, such as a Joint Photographic Experts Group (JPEG), Graphics Interchange Format (GIF) or Portable Network Graphics (PNG) formats, have more universal native support across platforms, but are static; e.g. lack the ability to zoom in and out on the image. A high-resolution image may allow clear viewing of individual nodes in a rendered signaling network at a high level of zooming, but when attempting to zoom out, the raster image is often distorted, where edges/links may disappear. Alternatively, a low-resolution image can give a clear picture of the overall network, but does not allow clear visualization of individual nodes when zooming in. Additionally, panning raster-based images is often implemented with scroll bars and frames but this design is limited and rough (Viswanathan *et al.*, 2007). Both visualization technologies: SVG and static raster images, allow for hyper-linking nodes to other web pages by clicking on the displayed nodes. This is accomplished by either specifying

*To whom correspondence should be addressed.

hyperlinks in the SVG source code, or by using an 'imagemap' in the HTML code that displays the raster image.

Avant-garde web technologies are emerging to overcome some of the limitations for hosting dynamically interactive maps on web sites, e.g. Google maps and Microsoft live search. Asynchronous JavaScript and XML (AJAX) serves as one of the central components of modern interactive webpage design and it is an integral component of Web 2.0 (Graham, 2005; O'Reilly, 2005). AVIS is an implementation of some of these new powerful web-based programming technologies.

2 IMPLEMENTATION

To advance currently used web-based signaling network visualization techniques, AJAX Viewer of Interactive Signaling Networks (AVIS) provides several important enhancements not currently available in other techniques/systems. It allows cross platform visualization of signaling networks with integrated zoom and pan functionality without requiring installation of a third party plug-in. AVIS is implemented using AJAX (Adams, 2005; Garrett, 2005) web programming paradigm. AVIS uses a combination of server side perl CGI scripts and libraries such as GraphViz, OverLib and PerlMagic (ImageMagic) to layout and render signaling networks as a series of image files at different resolutions from a provided structured text file listing interactions. The signaling pathway maps are served via a series of CGI scripts to the client browser. Cross-platform browser JAVA scripts display the image on the page, change its position and size when panning and zooming, and request new images when the zoom moves out of a specified range. The viewer provides a series of controls that allow zooming and panning. Similar actions can be performed by simply dragging the image within the viewer display area. More information about individual nodes and pop-up windows linking to other pages with details about the nodes and interaction is easily integrated and neatly displayed. The pop-ups are displayed by mouse-over and mouse-click events on nodes using the overLIB library. overLIB is a JavaScript library created to enhance websites with small pop-up information boxes. Information about interactions is displayed by mouse-over events without the need to reload a completely new page. The small pop-up information boxes provide external links to PubMed articles describing interactions and Swiss-Prot pages describing details about proteins in pathways. Animated zooming is implemented to navigate from node to node within the map. The information about nodes and links is provided by an editable text file that can reside at any remote web-based location. Users have the ability to change the URL containing the text describing the interactions visualized or load networks residing on their local client machine. Users can specify and modify the content of the text file with minimal details even after AVIS is loaded. Image files generated by AVIS can be downloaded in Graphics Interchange Format (GIF) format. AVIS is implemented using the Google syndicated Gadget API (Google Gadget API) and thus is trivial to implement it in any webpage or to add AVIS to users' Google home search page (Fig. 1). Website owners and webmasters can use AVIS to host cell signaling

maps with only specifying the interactions in a structured text files.

The layout of the nodes in maps is accomplished by the fileloader module. This provides developers with an easy way to add to AVIS additional layout algorithms. The included fileloaders that we implemented can be configured to use all of currently available Graphviz layout algorithms: dot, neato, fdp, circo and twopi. Alternatively, file formats, such as the Pajek.net file format (Batageli and Mryar, 2003) and a native.avis file format we developed (described in the manual) can include the layout in the input text file. As such, other layout programs can be used before importing into AVIS. Alternatively, a simple Perl wrapper can be added to automatically import other layout algorithms.

The graph rendering is accomplished by one of the two included graph drawing modules or by a user-defined graph drawing module. The default rendering functions generate a layout.dot file, calling Graphviz to do the actual rendering. In AVIS we implemented an alternative option: AVISgraphDraw. AVISgraphDraw uses PerlMagick (ImageMagic) image drawing API to draw the network map images. Both of these options are fairly simple graph rendering tools, but can draw a wide variety of networks as long as they can be described in terms of properties such as positions, shapes, size, colors of nodes, and placement and paths of edges. If more sophisticated graph rendering tools with other features are required, it is simple to add a Perl wrapper for such tools.

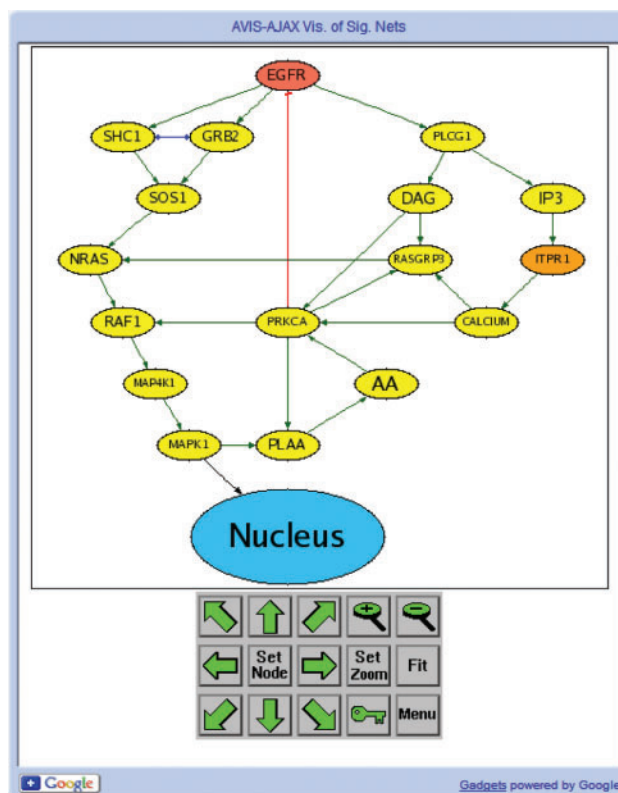


Fig. 1. Screenshot of AVIS as running as a Google Gadget embedded in a webpage with its default settings.

The process of developing such wrappers is described in the user manual.

AVIS supports several standard graph representation formats such as Pajek.net (Batagelj and Mrvar, 2003) and Systems Biology Markup Language (SBML) (Deckard *et al.*, 2006; Kitano *et al.*, 2005) as well as a simple structured text file format we developed called .sig (Ma'ayan *et al.*, 2005). It also supports a more flexible native.avis format which allows editing of any graph features, giving users more control over the placement of nodes, colors of nodes and arrows, shapes of nodes and types of arrows as well as more control on setting up hyperlinks. AVIS was implemented for visualization of PathwayGenerator, a tool that displays over 4000 automatically generated mammalian cell signaling maps created from a large-scale literature-based neuronal cell signaling network we developed (Ma'ayan *et al.*, 2005); AVIS was also implemented for visualization of NodeNeighborhood: a tool to visualize first and second interacting neighbors of mammalian proteins based on data from the IntAct database (Kerrien *et al.*, 2007) as well as two yeast-2-hybrid studies of mammalian protein interactions; and AVIS was implemented in Genes2Networks, a tool that can be used to create subnetworks from lists of genes and proteins by using 10 background mammalian protein interaction networks to 'connect' seed lists of gene symbols.

Other visualization programs are compatible with standards such as BioPAX, and PSI-MI (Stromback and Lambrix, 2005). AVIS can be adapted and upgraded to support any of these formats with simple Perl wrapped plug-ins. We hope that interested developers would find this tool easy to leverage for visualization of their system of interest.

ACKNOWLEDGEMENTS

This research is supported by NIH Grant No. GM-072853 and an advanced center grant from NYSTAR to R.I.

Conflict of Interest: none declared.

REFERENCES

- Adams,C. (2005) Remote Scripting with AJAX. XML.com. <http://www.xml.com/pub/a/2005/08/22/ajax.html>
- Batagelj,V. and Mrvar,A. Pajek (2003) Analysis and visualization of large networks. In Jünger,M. and Mutzel,P. (eds.), *Graph Drawing Software*. Springer, Berlin, pp. 77–103.
- Deckard,A. *et al.* (2006) Supporting the SBML layout extension. *Bioinformatics*, **22**, 2966–2967.
- Dogrusoz,U. *et al.* (2006) PATIKAwEB: a Web interface for analyzing biological pathways through advanced querying and visualization. *Bioinformatics*, **22**, 374–375.
- Garrett,J.J. (2005) Ajax: a new approach to web applications. Adaptive Path. <http://www.adaptivepath.com/publications/essays/archives/000385.php>
- Gough,N.R. (2002) Science's signal transduction knowledge environment: the connections maps database. *Ann. NY Acad. Sci.*, **971**, 585–587.
- Graham,P. (2005) Web 2.0. paulgraham.com. <http://www.paulgraham.com/web20.html>
- Hu,Z. *et al.* (2005) VisANT: data-integrating visual framework for biological networks and modules. *Nucleic Acids Res.*, **33**, W352–W357.
- Kerrien,S. *et al.* (2007) IntAct – open source resource for molecular interaction data. *Nucleic Acids Res.*, **35**, D561–D565.
- Kitano,H. *et al.* (2005) Using process diagrams for the graphical representation of biological networks. *Nat. Biotechnol.*, **23**, 961–966.
- Li,J. *et al.* (2002) The molecule pages database. *Nature*, **420**, 716–717.
- Ma'ayan,A. *et al.* (2005) Formation of Regulatory Patterns During Signal Propagation in a Mammalian Cellular Network. *Science*, **309**, 1078–1083.
- O'Reilly,T. (2005) What is Web 2.0. OreillyNet.com. <http://www.oreillynet.com/pub/a/oreilly/tim/news/2005/09/30/what-is-web-20.html>
- Rojdestvenski,I. (2003) Metabolic pathways in three dimensions. *Bioinformatics*, **19**, 2436–2441.
- Stromback,L. and Lambrix,P. (2005) Representations of molecular pathways: an evaluation of SBML, PSI MI and BioPAX. *Bioinformatics*, **21**, 4401–4407.
- Viswanathan,G.A. *et al.* (2007) BioPP: a tool for web-publication of biological networks. *BMC Bioinformatics*, **8**, 168.