NASA/TM-20230001615

Spatial Link Coverage Projections for the Glenn Research Center Communication Analysis Suite (GCAS)

Lucas Shalkhauser, Ian R. Nemitz, and Bryan W. Welch
Glenn Research Center, Cleveland, Ohio

April 2023

NASA STI Program . . . in Profile

Since its founding, NASA has been dedicated to the advancement of aeronautics and space science. The NASA Scientific and Technical Information (STI) Program plays a key part in helping NASA maintain this important role.

The NASA STI Program operates under the auspices of the Agency Chief Information Officer. It collects, organizes, provides for archiving, and disseminates NASA's STI. The NASA STI Program provides access to the NASA Technical Report Server—Registered (NTRS Reg) and NASA Technical Report Server—Public (NTRS) thus providing one of the largest collections of aeronautical and space science STI in the world. Results are published in both non-NASA channels and by NASA in the NASA STI Report Series, which includes the following report types:

- TECHNICAL PUBLICATION. Reports of completed research or a major significant phase of research that present the results of NASA programs and include extensive data or theoretical analysis. Includes compilations of significant scientific and technical data and information deemed to be of continuing reference value. NASA counter-part of peer-reviewed formal professional papers, but has less stringent limitations on manuscript length and extent of graphic presentations.

- TECHNICAL MEMORANDUM. Scientific and technical findings that are preliminary or of specialized interest, e.g., "quick-release" reports, working papers, and bibliographies that contain minimal annotation. Does not contain extensive analysis.

- CONTRACTOR REPORT. Scientific and technical findings by NASA-sponsored contractors and grantees.

- CONFERENCE PUBLICATION. Collected papers from scientific and technical conferences, symposia, seminars, or other meetings sponsored or co-sponsored by NASA.

- SPECIAL PUBLICATION. Scientific, technical, or historical information from NASA programs, projects, and missions, often concerned with subjects having substantial public interest.

- TECHNICAL TRANSLATION. English-language translations of foreign scientific and technical material pertinent to NASA's mission.

For more information about the NASA STI program, see the following:

- Access the NASA STI program home page at http://www.sti.nasa.gov

- E-mail your question to help@sti.nasa.gov

- Fax your question to the NASA STI Information Desk at 757-864-6500

- Telephone the NASA STI Information Desk at 757-864-9658

- Write to:
  NASA STI Program
  Mail Stop 148
  NASA Langley Research Center
  Hampton, VA 23681-2199

NASA/TM-20230001615

Spatial Link Coverage Projections for the Glenn Research Center Communication Analysis Suite (GCAS)

Lucas Shalkhauser, Ian R. Nemitz, and Bryan W. Welch
Glenn Research Center, Cleveland, Ohio

National Aeronautics and
Space Administration

Glenn Research Center
Cleveland, Ohio 44135

April 2023

Acknowledgments

I would like to express my gratitude for my mentors Ian Nemitz and Bryan Welch for their unwavering support and guidance throughout my internship. I would also like to thank NASA's Space Communications and Navigation program and the NASA Glenn Research Center internship coordinators. I am very grateful for this incredible opportunity and the knowledge and skills that I have gained.

Level of Review: This material has been technically reviewed by technical management.

This report is available in electronic form at https://www.sti.nasa.gov/ and https://ntrs.nasa.gov/

Spatial Link Coverage Projections for the Glenn Research Center Communication Analysis Suite (GCAS)

Lucas Shalkhauser, Ian R. Nemitz, and Bryan W. Welch
National Aeronautics and Space Administration
Glenn Research Center
Cleveland, Ohio 44135

Summary

Reliable space communications are fundamental to every NASA mission. Precise planning, simulation, and analysis of communications capabilities are essential to understanding exactly what data rates, visibility times, and signal strengths can be expected during actual missions. In this work, the goal was to take a three-dimensional (3D) icosphere in which each face was a point to be evaluated, and display the link performance on a two-dimensional (2D) projection of celestial bodies. The result would allow for links to a node location that could be spatially located at any of the icosahedral faces. These 2D projections were produced for a wide range of spatial link grid sizes, then compressed into compact MAT files for ease of access and rapid implementation. The final icosahedral projections can be displayed as overlays onto maps of various celestial bodies, using ultra-high-resolution maps for stitching. The results have proven satisfactory, and plans call for them to be used to support various simulation efforts.

Nomenclature

GCAS	Glenn Research Center Communication Analysis Suite
2D	two-dimensional
3D	three-dimensional

Introduction

Simulating line-of-sight communications between assets located in space (such as satellites or the International Space Station) and various ground locations is an essential part of planning and executing a successful space mission. The NASA Glenn Research Center Communication Analysis Suite (GCAS) software offers a powerful tool for simulating complex space communication networks in a wide variety of ways (Refs. 1 to 8). When planning for missions like the ones included in the Moon to Mars programs, GCAS's spatial link coverage projection capability can be employed to simulate and understand various properties of a communication link, such as its data rate, link margin, and equivalent isotopically radiated power (among others) (Ref. 9). GCAS can also graph and animate its output in several ways to help users understand the simulation results (Refs. 9 to 11).

The GCAS spatial dynamic link analysis and spatial link constrained coverage analyses support multiple solve-for modes that can assess link performance across a geographic area. For this effort, a specialized graphing method has been created for visualization of this spatial link analysis data, allowing link budgets to be imaged over a 2D projection of a celestial body's surface (Refs. 10 and 11). To accomplish this, the icosphere that GCAS employs for spatial analysis is projected to 2D, corrected for errors, compressed into a special data structure, and displayed using output data from GCAS.

Icospheres

An icosphere, a geodesic polyhedron based on icosahedral symmetry, is a spherical polyhedron made up of equilateral triangles that can be used to create 3D images of spherical objects (Figure 1). To clarify, an icosphere is a geometric mesh primitive (a commonly used and basic mesh) that can be used to create an approximation of a sphere using solely triangular faces (Refs. 10 and 12). To form these faces, a regular icosahedron's faces are each subdivided into four identical subtriangles (Refs. 10 and 12). Although other practical methods are available for subdividing a sphere into 2D shapes (e.g., UV spheres, which are made up of four-sided faces and a common alternative for generating 3D models) (Refs. 13 and 14), icospheres provide a more uniform texture and layout, maintaining triangular faces of equal, regularly arranged vertices. For this reason, they are more suitable for use in certain settings, such as map projections (Refs. 10 and 12).

The ability to divide the Earth's surface into subsets of equilateral triangles of varying sizes makes this icosahedral method ideal for displaying GCAS data (Ref. 10). The minimum configuration for an icosphere is 20 faces (a dodecahedron) (Refs. 10 and 12). As mentioned previously, to create the next higher order icosphere, each face must be divided into four equilateral triangles, with the distance to the center of each new vertex having been normalized. Therefore, as the order of the icosphere used increases by 1, the number of faces quadruples. The Icosphere MATLAB® (The MathWorks) script takes in n as the next size of the icosphere, with n = 0 indicating 20 faces, n = 1 indicating 80 faces, etc. The formula to calculate the number of faces from n is

$$\text{Number of faces} = 5 \cdot 2^{2 \cdot n+2} . \tag{1}$$

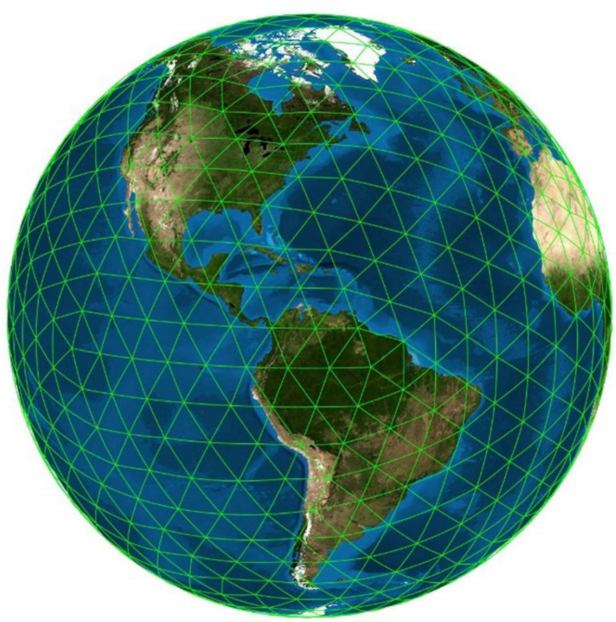

Figure 1.—Representation of icosphere with 320 faces (n = 2) drawn onto Earth's surface.

TABLE I.—ICOSPHERE ORDER, NUMBER OF FACES, AND FACE AREA IF PLACED ONTO EARTH'S SURFACE

n	Face count	Area, km^2
0	20	25,505,000.00
1	80	6,376,250.00
2	320	1,594,100.00
3	1,280	398,515.63
4	5,120	99,628.91
5	20,480	24,907.23
6	81,920	6,226.81
7	327,680	1,556.70
8	1,310,720	389.18
9	5,242,880	97.29
10	20,971,520	24.32
11	83,886,080	6.08

GCAS uses the center of each icosphere face to solve for its various special analysis modes. An icosphere is the ideal shape to divide a sphere, both because each face covers an equal area, and the center of each face is equidistant from the center of each neighboring face. GCAS currently has the capability to use an icosphere with up to 83,886,080 faces ($n = 11$). All icosphere data is generated and processed in advance to speed up script runtimes. Table I shows each currently generated icosphere size with its approximate area per face when placed on the surface of the Earth, based on a surface area of 510.1 million km^2.

Projecting to 2D

The next stage of this project was to take each icosphere generated and project it onto a standard 2D equirectangular projection map, where the center of the map is at 0 degrees latitude North (lat. 0° N) and 0 degrees longitude East (long. 0° E). Each planet and moon that GCAS supports has its own corresponding map. This gives users a perspective of the output of a spatial analysis solve-for mode, a view of the entire surface of the Earth (or the relevant central body of the spatial grid node), in a way that is easy to display, copy, and save for future use and presentation.

The icosphere generation MATLAB® script returns a struct with the icosphere's vertices and faces as cartesian coordinates with a radius of 1, centered at the cartesian coordinate origin. For the first major step, that is, displaying properly on an equirectangular projection map, the vertices must be converted from cartesian coordinates to spherical coordinates. One factor to consider prior to this conversion is that the equirectangular map is a projection map, which causes increasingly intense distortions in area when approaching the North or South Poles. Therefore, a triangle placed on the surface of the Earth in 3D will have curved sides when it is projected onto the 2D map. The simplest way to accomplish this is to interpolate points between each of the three vertices of each triangle. That way, when converting to spherical coordinates, the extra points show the curve in each face. The number of interpolated points is important to consider because adding even a single point between each vertex can nearly double the total size of the output data file. The script allows the user to select the desired number of interpolated points between each vertex. However, once the icosphere reaches 81,920 faces, the faces are small enough that interpolation is no longer necessary. Because the complete data files for the projections were preprocessed and generated, an appropriate number of interpolated points was selected for each icosphere size to allow for an appropriate visual fidelity and compact file size (Table II).

TABLE II.—ICOSPHERE ORDER AND FACE NUMBER WITH CORRESPONDING NUMBER OF INTERPOLATED POINTS

n	Face count	Interpolated points
0	20	13
1	80	9
2	320	7
3	1,280	5
4	5,120	3
5	20,480	1
6	81,920	0
7	327,680	0
8	1,310,720	0
9	5,242,880	0
10	20,971,520	0
11	83,886,080	0

Equations (2) and (3) are the standard equations used to convert the x, y, and z cartesian coordinates to spherical coordinates in degrees for longitude and latitude respectively:

$$\text{Longitude} = \operatorname{atan2}(y,x)\cdot\frac{180}{\pi}. \tag{2}$$

$$\text{Latitude} = \operatorname{asin}\left(\frac{z}{\sqrt{x^2+y^2+z^2}}\right)\cdot\frac{180}{\pi}. \tag{3}$$

Although these equations were satisfactory for converting most points, after feeding in all data, a significant number of visual projection errors were present. As Figure 2 shows, many lines near the North and South Poles converge on lat. ±90° N and long. 0° E. On a sphere, all these lines are connected to the same point directly on the pole. On an equirectangular projection map, all the lines with a point at lat. ±90° N should point straight down. Note the errors along the left and right edges of the map, where all points with long. ±180° E have defaulted to +180° E. This causes faces on the left-hand side of the map to stretch over to +180° E when it should be –180° E. The final main projection errors are near the corners, where some of the faces are stretched across ±180° E and need to be found and split into two separate faces. Each projection error is detected and accounted for by carefully examining their respective latitude and longitude values and adding or modifying points to create the correctly projected faces. The corrected projection with no interpolated points and 80 faces is shown in Figure 3; Figure 4 shows the correction with interpolated points.

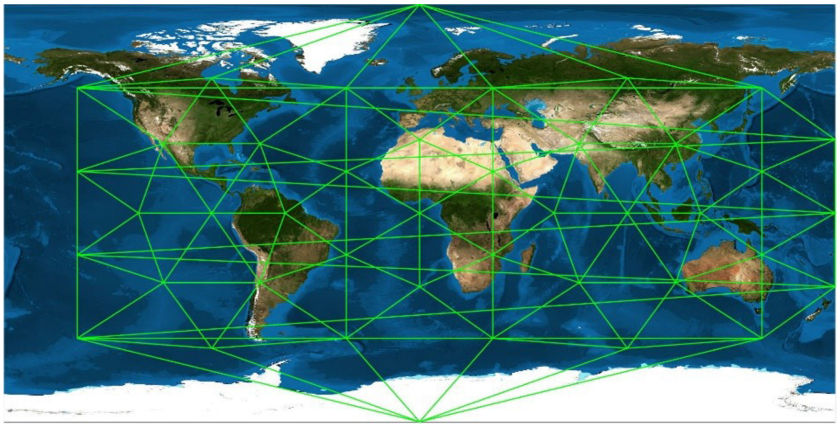

Figure 2.—Icosphere with 80 faces projected onto 2D equirectangular map with no corrections.

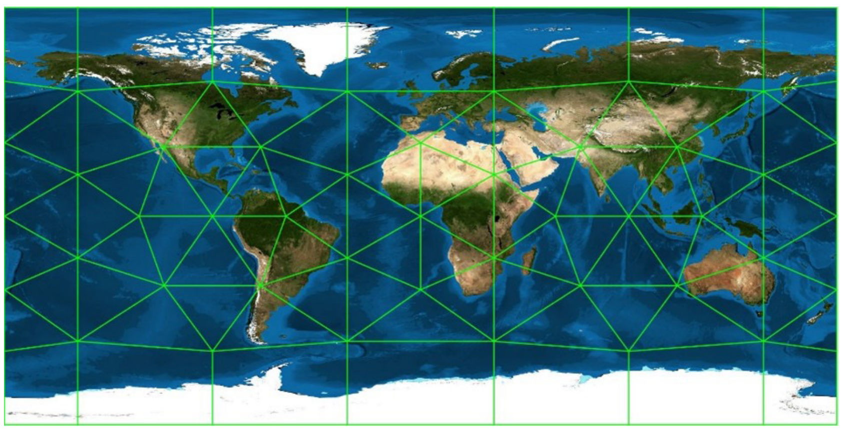

Figure 3.—Icosphere with 80 faces projected onto 2D equirectangular map with corrections.

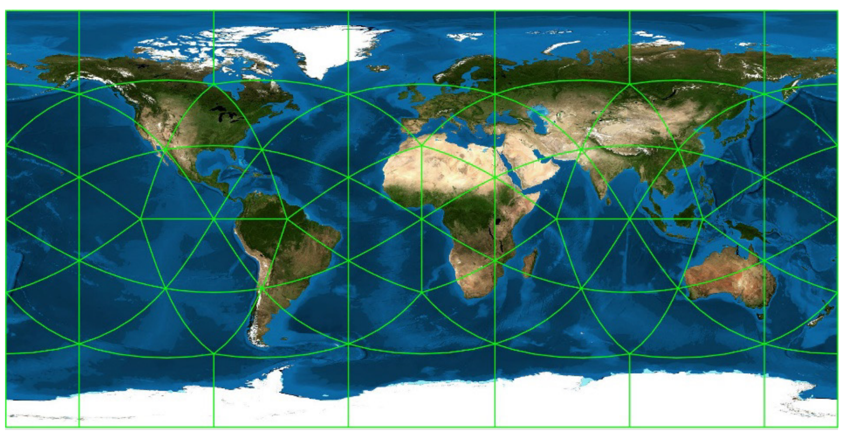

Figure 4.—Icosphere with 80 faces projected onto 2D equirectangular map with corrections and interpolated points.

Compressing

One important aspect of this project was processing the output MATLAB® MAT files in a format that allowed for the smallest file size possible but that also was not computationally intensive to unpack and use the appropriate points for each face. After projecting, all faces are stored in an array of cells, where each cell contains the list of points that make up each face. Because cells contain a significant amount of data overrun when stored in a MAT file, the goal was to store the points in a different, more compact, format. After testing a series of different data structs and MAT file versions, the overall best format for storing the data was in two multidimensional arrays. One array called faces, with a 4-byte Single data type, contains the longitude and latitude of every single vertex, one after the other. The second array called key, with a 4-byte Uint32 data type, contains a row for each face in the icosphere. The first column in key contains the starting index and the second column contains the ending index of each face. These index values reference the location of the vertices in the faces array. The key array also contains two more columns for the starting and ending index of a second face. The extra columns are for the icosphere faces that needed to be converted to two separate faces for the projection map (as noted previously). This format is very easy to interpret and extremely fast to index through using a MATLAB® runtime application.

During testing, when storing the data in its default cell array structure, the file size was slightly larger than 50 GB for the $n = 11$ icosphere. The goal was to reduce the total size to less than 1 GB. After using the new data format and saving the data using a version 7.3 MAT file, the file size for the same icosphere was slightly more than 1.07 GB. Although the desired file size goal was not reached, it was near enough and a considerable improvement over the cell array data structure. The combined total size of all icosphere projections from $n = 0$ to $n = 11$ was 1.36 GB.

Displaying

The final step of the project was to use the icosphere projection to plot out data from a spatial analysis simulation. Figure 5 illustrates an example output that uses a grid size of 320 faces and a single orbiting satellite overhead. The script for displaying the spatial coverage analyses automatically selects the appropriate background map based on the selected planetary body in GCAS. When the latitude and longitude limits are small enough and the planetary body selected is Earth, it uses high-resolution maps that help show more detail of the surface. In Figure 5, note the lightly opaque faces of various colors. Each color corresponds to a link margin value shown on the right-hand side of the map, which is indicative of the quality of the link connection. Yellow and gold indicate superior link performance; blue and black indicate poorer performance. Faces with no visibility show no color, but their black outlines are retained.

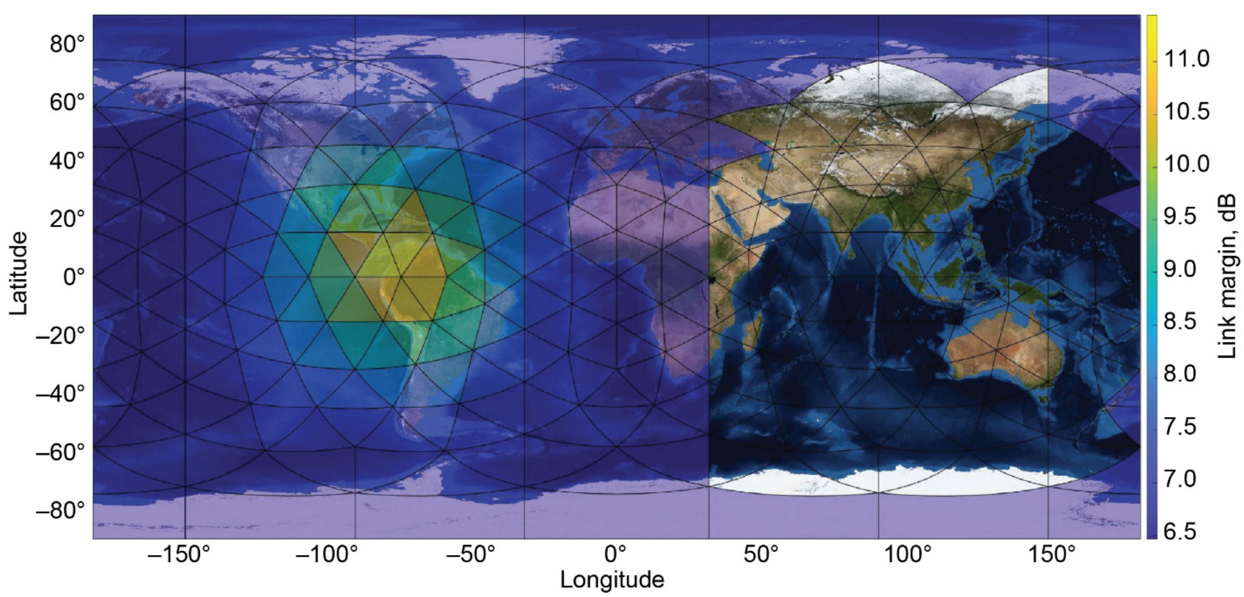

Figure 5.—Example projection with spatial analysis coverage data.

Conclusion

Overall, the two-dimensional (2D) icosphere projection for the spatial analysis coverage mode for Glenn Research Center Communication Analysis Suite (GCAS) works very well and offers an easy way to visualize the output from the simulation. The addition of this tool extends GCAS's capabilities and makes it easier for scientists and engineers to work on upcoming NASA missions. Also, the projection MAT files can be used for further NASA analyses and needs and can be easily exported and used elsewhere in other programming languages.

References

1. Gilbertson, Christian; and Welch, Bryan: Demonstrating High-Accuracy Orbital Access Using Open-Source Tools. NASA/TM—2017-219562, 2017. https://ntrs.nasa.gov
2. Green, Jack L.; Welch, Bryan W.; and Manning, Robert M.: Optical Communication Link Atmospheric Attenuation Model. NASA/TM—2019-219991, 2019. https://ntrs.nasa.gov
3. Green, Jack L.; Welch, Bryan W.; and Manning, Robert M.: Updating the Space Communications and Navigation (SCaN) Link Tool Executable Software to Version 5. NASA/TM—2019-220234, 2019. https://ntrs.nasa.gov
4. Ong, Victor Christopher A.; and Welch, Bryan W.: Bit Error Rate and Frame Error Rate Data Processing for Space Communications and Navigation-Related Communication System Analysis Tools. NASA/TM—2019-220070, 2019. https://ntrs.nasa.gov
5. Shah, Alay M.; and Welch, Bryan W.: Space Communications and Navigation (SCaN) Tool Capabilities Updates for Antenna Pattern and Jitter. NASA/TM—2019-220233, 2019. https://ntrs.nasa.gov
6. Steinkerchner, Leo; and Welch, Bryan: Augmenting the SCaN Link Budget Tool With Validated Atmospheric Propagation. NASA/TM—2017-219563, 2017. https://ntrs.nasa.gov
7. Vitez, Cory T.; and Welch, Bryan W.: Enabling Dynamic Vehicle Analyses With Improved Atmospheric Attenuation Models in Glenn Research Center Communication Analysis Suite. NASA/TM—2020-220512, 2020. https://ntrs.nasa.gov
8. Salama, Hady; and Welch, Bryan W.: On Development of Modern Software Interface to Glenn Research Center's Communication Analysis Suite. NASA/TM—2020-220513, 2020. https://ntrs.nasa.gov
9. Welch, Bryan W.: GRC Communication Analysis Suite User's Guide. NASA Glenn Research Center, Communications and Intelligent Systems Division, unpublished internal document, 2022.
10. Jasper, Lindsey A.; and Welch, Bryan: Verification of an Icosahedral Grid for Strategic Center for Networking, Communications, and Integration User Interface Spatial Capabilities. NASA/TM—2018-219993, 2018. https://ntrs.nasa.gov
11. Shalkhauser, Lucas D.; Henderson, Eric; and Welch, Bryan W.: On Development of Three-Dimensional Visualization Capabilities in Glenn Research Center Communication Analysis Suite. NASA/TM-20205000040, 2020. https://ntrs.nasa.gov
12. HandWiki Encylopedia of Science and Computing: Icosphere. 2021. https://handwiki.org/wiki/Icosphere Accessed Dec. 20, 2022.
13. Blender: What is the difference between a UV Sphere and an Icosphere? 2023. https://blender.stackexchange.com/questions/72/what-is-the-difference-between-a-uv-sphere-and-an-icosphere Accessed March 8, 2023.
14. Blender: Blender 3.4 Manual. 2023. https://docs.blender.org/manual/en/latest/modeling/meshes/primitives.html Accessed Dec. 20, 2022.