

АЛГОРИТМ ПОСТРОЕНИЯ ИЗОЛИНИЙ В РАЗЛИЧНЫХ СИСТЕМАХ КООРДИНАТ

Ю.В. Пенских

Институт солнечно-земной физики СО РАН, Иркутск, Россия
pensikh@iszf.irk.ru

ALGORITHM FOR DRAWING ISOLINES IN VARIOUS COORDINATE SYSTEMS

Yu.V. Pensikh

Institute of Solar-Terrestrial Physics SB RAS, Irkutsk, Russia
pensikh@iszf.irk.ru

Аннотация. Карты изолиний являются одним из основных способов визуализации двумерных скалярных полей. При построении изолиний в декартовой системе координат обычно используется алгоритм «шагающих квадратов». В работе описывается применение этого алгоритма в различных системах координат, в том числе криволинейной. Разработана программа рисования изолиний в полярной, а также в сферической (географической и геомагнитной) системе координат. Программа позволяет отображать изолинии на сферической поверхности заданного радиуса с помощью приложения Google Earth. Алгоритм и программа используются в технике инверсии магнитограмм (ТИМ-ИСЗФ) для визуализации двумерных распределений ионосферных и продольных токов, электрического потенциала и других электродинамических параметров полярной ионосферы в геомагнитной системе координат. Работа выполнена в рамках базового финансирования программы ФНИ П.16 и при частичной поддержке грантов РФФИ № 18-05-00437 и 19-05-00665.

Ключевые слова: изолинии, шагающие квадраты, криволинейные координаты, геомагнитная система координат.

Abstract. Isoline maps are one of base way to visualize scalar fields is contour lines. For constructing contour lines in the Cartesian coordinate system, the «marching squares» algorithm is usually used. The paper describes the application of the algorithm to various coordinate systems, including curvilinear. A software for drawing isolines in the polar and spherical (geographic and geomagnetic) coordinate system has been developed. It allows one to display isolines on a spherical surface of given radius by using the Google Earth application. The algorithm and the software is used in the magnetogram inversion technique (MIT-ISTP) to visualize 2D distributions of ionospheric and field-aligned currents, electric potential and other electrodynamic parameters of the polar ionosphere for geomagnetic coordinate system. The work was performed with budgetary funding of Basic Research program П.16. This work was also partially supported by the Russian Foundation for Basic Research under Grants Nos. 18-05-00437 and 19-05-00665.

Keywords: isolines, marching squares, curvilinear coordinates, geomagnetic coordinate system.

ВВЕДЕНИЕ

Визуализация данных требуется во многих научно-исследовательских задачах. Существуют различные способы визуализации двумерного скалярного поля [Chen et al., 2008], одним из которых являются изолинии.

Техника инверсии магнитограмм (ТИМ) по данным измерений геомагнитного поля на мировой сети наземных магнитных обсерваторий рассчитывает основные электродинамические параметры полярной ионосферы: двумерные распределения эквивалентных токов, электрического потенциала, горизонтальных и продольных токов, а также ряда других [Мишин, 1976; Базаржапов и др., 1990; Mishin, 1990]. Все эти выходные данные необходимо визуализировать в виде 2D карт для более удобного анализа.

ТИМ использует геомагнитную (дипольную) систему координат: геомагнитная широта (Φ) — местное геомагнитное время (MLT). В этой системе координат для каждого полушария строятся карты изолиний двумерных распределений эквивалентной токовой функции, электрического потенциала и других выходных данных ТИМ. Изображения строятся в полярной проекции Постеля. Поскольку полуночный меридиан 0 MLT располагается внизу рисунка, то общепринятую в математике полярную систему координат необходимо повернуть по часовой стрелке на 90° .

Для построения изолиний выходных параметров ТИМ долгое время использовался язык программирования IDL. Помимо достоинств, IDL обладает и

множеством недостатков: ограниченность языка, сложность интеграции с другими системами, крайне низкое количество и качество документации и литературы и др. Построение изображений в векторном формате было одной из задач, поставленных в план модернизации ТИМ. Большинство существующих библиотек для визуализации данных строят растровые изображения изолиний в декартовой системе координат. Нам же было необходимо строить изолинии в полярной системе координат в векторном формате. Популярная библиотека MATPLOTLIB на момент начала работ, согласно багтрекера [<https://github.com/matplotlib/matplotlib/labels/polar>], не соответствовала нашим требованиям. Все это стимулировало нас на поиск нового решения для построения изолиний.

ОПИСАНИЕ АЛГОРИТМА ПОСТРОЕНИЯ ИЗОЛИНИЙ

Для поверхности $F(x, y)$ и множества секущих Z_i , каждая i -я изолиния удовлетворяет $F(x, y)=z_i$, т. е. проходит вдоль равных значений (рис. 1).

Поверхность представляется скалярным полем в виде матрицы чисел. Одним из самых популярных алгоритмов построения изолиний скалярного поля является алгоритм «шагающих квадратов» [Lorenzen, Cline, 1987; Mantz et al., 2008]. Обычно этот алгоритм рассматривают для декартовой системы координат. Однако несложно расширить его и для криволинейной системы координат. Рассмотрим кратко его суть.

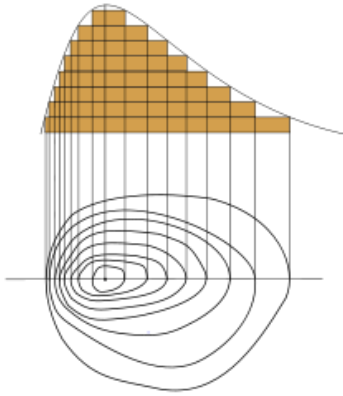


Рис. 1. Изолинии как проекция сечений поверхности на плоскость

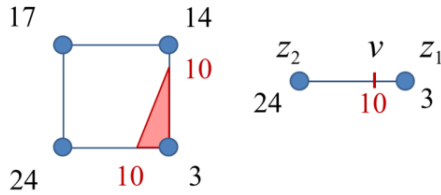


Рис. 2. Пример определения точек пересечения ячейки с изолинией (слева), и стороны (справа)

Проведем ряд упрощений. Рассмотрим построение изолиний только для одной ячейки скалярного поля и только для одного уровня изолинии $F(x_2, y_2)=v$. Таким образом, задача сводится к поиску точек пересечения четырехугольника с плоскостью в 3D пространстве. Четырехугольник можно рассматривать как лист, а стороны четырехугольника — как четыре прямых отрезка. Таким образом, задача сводится к отысканию точки пересечения отрезка с секущей плоскостью. Концы отрезка имеют координаты (x_1, y_1, z_1) и (x_2, y_2, z_2) , т. е. $F(x_1, y_1)=z_1$ и $F(x_2, y_2)=z_2$. Будем полагать, что $z_1 < z_2$, в противном случае просто изменим нумерацию концов отрезков. Очевидно, что плоскость $F(x, y)=v$ пересекает отрезок только в случае $z_1 < v < z_2$. В случае пересечения отрезка можно определить соотношение k , в котором точка v делит отрезок (рис. 2):

$$k = (v - z_1) / (z_2 - z_1) \quad (1)$$

По соотношению несложно найти и координаты точки пересечения (x_v, y_v, z_v) .

$$\begin{aligned} x_v &= x_1 + k(x_2 - x_1) \\ y_v &= y_1 + k(y_2 - y_1) \\ z_v &= v \end{aligned} \quad (2)$$

Аналогично можно получить координаты пересечения с другим отрезком той же ячейки. Если соединить полученные точки отрезком, то получим часть изолинии. Если выполнять три вложенных цикла: по уровням изолиний, по ячейкам и по четырем сторонам ячейки, то в результате будет построена карта изолиний. Точность рисования в любом случае будет зависеть от шага сетки. Чем чаще сетка, тем точнее результат.

Для рисования изолиний в криволинейной системе координат можно использовать два подхода.

Первый подход. В качестве (x, y) используются индексы (i, j) массива скалярного поля. Координаты

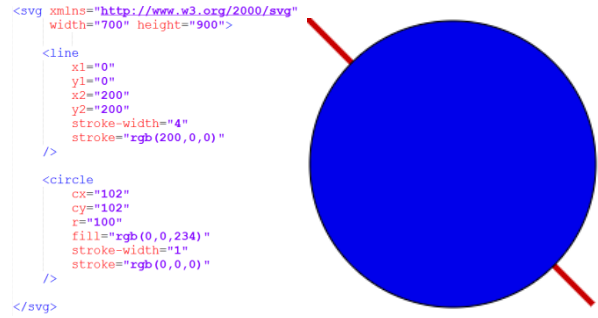


Рис. 3. Пример содержимого SVG файла (слева) и его изображения в браузере (справа)

пересечений определяются по формуле (2) в долях индексов. Затем по этим индексам определяются криволинейные координаты.

Второй подход. По индексам (i, j) массива скалярного поля определяются криволинейные координаты (x, y) в узлах. Координаты пересечений определяются по той же формуле (2), но уже в криволинейных координатах.

Первый подход следует использовать при большом количестве пересечений ячеек с изолиниями (большое количество уровней) и вычислительно сложной формуле преобразования в криволинейные координаты. В противном случае второй подход более эффективен.

ФОРМАТ SVG

Изолинии можно представлять в векторной графике. Одним из популярных форматов векторной графики является SVG (Scalable Vector Graphics) [https://w3.org/TR/2011/REC-SVG11-20110816/intro.html] (рис. 3). Для просмотра SVG изображений не требуется установка дополнительного программного обеспечения, достаточно современного браузера. SVG — язык для описания двумерной графики в XML (Extensible Markup Language) [https://w3.org/XML]. XML — тестовый формат с достаточно простым формальным синтаксисом, который использует структуру с вложенными элементами и атрибутами.

ФОРМАТ KML

KML — это формат файлов, который используется для отображения географических данных в геобраузерах, таких как Google Earth, Google Maps. KML — это открытый стандарт, официально называемый OpenGIS KML Encoding Standard (OGC KML) [https://opengis.org/standards/kml] [Burggraf, 2015]. OGC KML создан на основе стандарта XML и имеет механизм расширений, посредством которых можно увеличить пространство имен. KML позволяет отображать линии, метки и другие элементы на одном или нескольких слоях изображения.

ФОРМАТ KMZ

Данные в текстовом формате имеют больший размер по сравнению с бинарным форматом. Поэтому даже небольшое векторное изображение в формате KML имеет большой размер, что негативно сказывается на работе с таким файлом. Google Earth, помимо обычных файлов KML, может работать со сжатыми данными KMZ (KML ziped). KMZ-файл

является обычным ZIP-архивом, внутри которого содержится KML-файл.

Все языки программирования имеют функции для работы с текстовыми файлами. Зная SVG, KML, XML спецификации и используя формулы (1) и (2), несложно написать программу для генерации векторных изображений, что и было выполнено в ходе данной работы.

ВИЗУАЛИЗАЦИЯ ВЫХОДНЫХ ДАННЫХ ТИМ

На основе вышеизложенного алгоритма построения изолиний была создана программа для генерации SVG, KML (KMZ) файлов по выходным данным ТИМ (рис. 4, 5). Эта программа создает слои изображений:

- эквивалентная токовая функция;
- продольные токи при однородной проводимости;
- мировая сеть станций, по которой проводился сферический гармонический анализ [Gjerloev, 2012];
- координатная сетка геомагнитных координат для соответствующей моменту времени эпохи [Thebault et al., 2015];
- координатная сетка географических координат;
- меридианы полдень-полночь, как в географической, так и в геомагнитных системах координат и др.

Изображения строятся на сфере, как для северного, так и для южного полушария, благодаря расширению алгоритма ТИМ на два полушария [Lunushkin et al., 2019].

Для просмотра KML (KMZ) файлов мы используем приложение Google Earth [<https://google.com/intl/ru/earth>], которое позволяет интерактивно менять видимость слоев изображений, проводить навигацию и масштабирование с точностью до метра. Google Earth позволяет сделать привязку к конкретному моменту времени, показывать Солнце, затенять неосвещенную часть Земли и многое другое. Все это дает исследователю наглядное представление изучаемых магнитосферно-ионосферных процессов.

ЗАКЛЮЧЕНИЕ

Показан и реализован способ применения алгоритма «шагающих квадратов» в криволинейной системе координат. Создана программа, которая

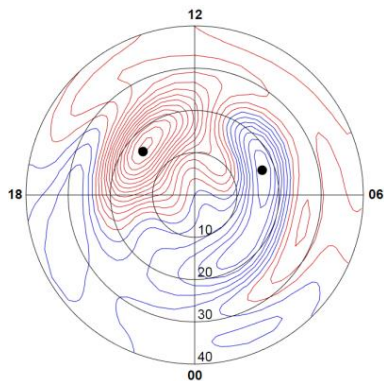


Рис. 4. Эквивалентная токовая функция в формате SVG

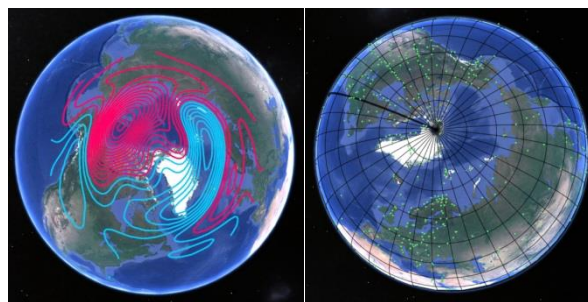


Рис. 5. Эквивалентная токовая функция (слева) и сеть наземных магнитометров в геомагнитной системе координат (справа) на сфере

по выходным данным ТИМ генерирует SVG файлы для полярной системы координат, KML (KMZ) файлы для сферической географической системы координат. Предложенный способ визуализации данных является удобным инструментом в руках исследователя.

Автор выражает благодарность С.Б. Лунюшкину за поставленную задачу и плодотворные дискуссии.

Работа выполнена в рамках базового финансирования программы ФНИ П.16 и при частичной поддержке грантов РФФИ № 18-05-00437 и 19-05-00665.

СПИСОК ЛИТЕРАТУРЫ

- Базаржапов А.Д., Матвеев М.И., Мишин В.М. Геомагнитные вариации и бури. Наука, Новосибирск. 1979. 248 с.
- Мишин В.М. Спокойные геомагнитные вариации и токи в магнитосфере. Наука, Новосибирск. 1976. 208 с.
- Burggraf D. OGC KML 2.3. OGC. 2015.
- Chen C.-H., Härdle W., Unwin A. Handbook of Data Visualization, 2008. DOI: [10.1007/978-3-540-33037-0](https://doi.org/10.1007/978-3-540-33037-0).
- Gjerloev J.W. The SuperMAG data processing technique // J. Geophys. Res.: Space Phys. 2012. V. 117, N A9. P. A09213. DOI: [10.1029/2012ja017683](https://doi.org/10.1029/2012ja017683).
- Lorenson W., Cline H. Marching cubes: a high resolution 3D surface construction algorithm // ACM SIGGRAPH Computer Graphics. 1987. Vol. 21. DOI: [10.1145/37401.37422](https://doi.org/10.1145/37401.37422).
- Lunushkin S.B., Mishin V.M., Karavaev Y.A., et al. Study of the dynamics of electric currents and polar caps in the ionospheres of the two hemispheres during the geomagnetic storm on 17 August 2001 // Solar-Terr. Phys. 2019. V. 5, N 2.
- Mantz H., Jacobs K., Mecke K. Utilizing Minkowski functionals for image analysis: a marching square algorithm // Journal of Statistical Mechanics: Theory and Experiment. 2008. DOI: [10.1088/1742-5468/2008/12/P12015](https://doi.org/10.1088/1742-5468/2008/12/P12015).
- Mishin V.M. The magnetogram inversion technique and some applications // Space Sci. Rev. 1990. V. 53, N 1-2. P. 83-163. DOI: [10.1007/bf00217429](https://doi.org/10.1007/bf00217429).
- Thebault E., et al. International Geomagnetic Reference Field: the 12th generation // Earth Planet Sp. 2015. V. 67, N 1. P. 79. DOI: [10.1186/s40623-015-0228-9](https://doi.org/10.1186/s40623-015-0228-9).
 URL: <https://github.com/matplotlib/matplotlib/labels/polar>.
 URL: <https://w3.org/TR/2011/REC-SVG11-20110816/intro.html>.
 URL: <https://w3.org/XML>.
 URL: <https://openeospatial.org/standards/kml>.
 URL: <https://google.com/intl/ru/earth>.