

География

УДК 556.5, 004.42

ПРОГРАММНОЕ СОЗДАНИЕ И ОБЪЕДИНЕНИЕ SHAPE-СЛОЕВ
ОБЛАСТЕЙ ЗАТОПЛЕНИЯ ТЕРРИТОРИЙ

Е. Б. СЕРГЕЕВ *

*Всероссийский научно-исследовательский институт по проблемам
гражданской обороны и чрезвычайных ситуаций МЧС России,
Москва, Россия*

В статье рассматривается технология автоматического создания и объединения векторных слоев областей затопления на основе данных уровней воды и цифровой модели рельефа местности с целью просмотра затопленных территорий на географической карте. Особое внимание уделяется проблемам формирования полигонов областей затопления из географических точек областей затопления. Приводится программный код создания shape-слоев из полученных полигонов и объединения нескольких shape-слоев в один.

<https://doi.org/10.46991/PYSUC.2025.59.2.549>

Keywords: water levels, digital relief model, flood area, polygon, shape layer, combining shape layers.

Введение. Для просмотра на географической карте области, затопленной в результате наводнений территории, и пострадавших объектов жилой или производственной инфраструктуры необходимо создать так называемые shape-слои [1], которые затем накладываются на электронную географическую карту. Из цифровой модели рельефа местности (ЦМР) на основе данных об уровнях воды в метрах над уровнем моря, географических координатах речного створа (или гидропоста) и размерах расчетной области, можно определить определенный набор точек затопленной территории. Задача состоит в том, чтобы из этого набора точек сформировать описывающий затопленную область shape-слой, который можно добавить на географическую карту в среде QGIS для просмотра. Для создания площадного shape-слоя необходимо вначале определить границы области этого набора точек, и затем из пограничных точек сформировать полигон [2], то есть упорядоченный замкнутый набор точек. При этом начальная и конечная точки полигона должны совпадать между собой. Таким образом, вопрос автоматического создания shape-слоя разбивается на две отдельные задачи:

1. автоматическое формирование одного или несколько полигонов из набора точек затопленной области;

* E-mail: alik574@mail.ru

2. программное создание shape-слоя из полученных полигонов.

Безусловно, разработчики большинства геоинформационных систем, таких как ArcGIS, QGIS и других, имеют в своем арсенале наработанные алгоритмы и программные продукты, решающих подобные задачи. Однако в открытой печати отсутствуют описания этих алгоритмов и программных кодов. Это не отменяет необходимости для гидрологов на основе данных об уровнях воды программным путем определять области затопленной территории при наводнениях. Данная работа и направлена на заполнении пробелов в решении этой задачи.

Материал и методика исследования.

Методика создания *shape*-слоя. Формирование полигона из произвольного набора точек осуществляется следующим образом. Вначале находим так называемые пограничные точки из набора. Пограничные точки определяем как точки, которые соседствуют с точками незатопленной территории. Это хорошо показано на рис. 1, где представлена прямоугольная область в виде ячеек таблицы с указанием высоты над у. м. в м (ячейки таблицы рассматриваются в виде отдельных точек). На рисунке белым цветом показаны ячейки незатопленной территории. Пограничные ячейки – это те, которые соприкасаются хотя бы одной стороной с ячейками незатопленной территории. Они выделены темным цветом. Определим также так называемые внутренние ячейки затопленной области. Это те ячейки, которые со всех четырех сторон окружены ячейками затопленной области (включая и пограничными ячейками). Они показаны серым цветом.

Рис. 1. Область затопления выделена серым цветом, а часть ее пограничных точек – темным цветом.

Таким образом, стоит задача – из семейства ячеек, выделенных темным цветом, программным путем составить такой упорядоченный набор, чтобы он охватывал область, помеченную серым цветом, и был замкнутым, то есть, чтобы первая и последняя ячейка этого набора совпадали. Следует отметить, что координаты всех остальных ячеек полигона, кроме первой и последней, не должны совпадать.

Результаты исследований и их обсуждение. Алгоритм подобного упорядочивания пограничных точек мог бы выглядеть так. Возьмем первую произвольную пограничную ячейку. Запоминаем ее координаты как первую точку полигона, и ищем ближайшую соседнюю пограничную ячейку, чтобы ближайшие внутренние точки оставались только по одну сторону (например, с правой стороны движения – это движение по часовой стрелке). Координаты новой пограничной ячейки запоминаем как вторую точку полигона. Для новой пограничной ячейки повторяем ту же процедуру. И так до того момента, когда координаты какой-либо новой ячейки совпадут с координатами первой ячейки. Тогда первый полигон области затопления сформирован.

Однако такой алгоритм в большинстве случаев не проходит. Очень часто такое движение упирается в “тупик”. Чтобы избежать подобного, необходимо часть пограничных ячеек вывести из рассмотрения. А именно, те пограничные ячейки, которые не соседствуют (хотя бы одной из четырех сторон) с вышеуказанными внутренними ячейками области затопления. На рис. 2 показан пример такого выбора. В нем внутренние ячейки затопленной области закрашены серым цветом, темным цветом – пограничные ячейки, входящие в состав полигонов, а забракованные пограничные ячейки заштрихованы прямоугольной сеткой.

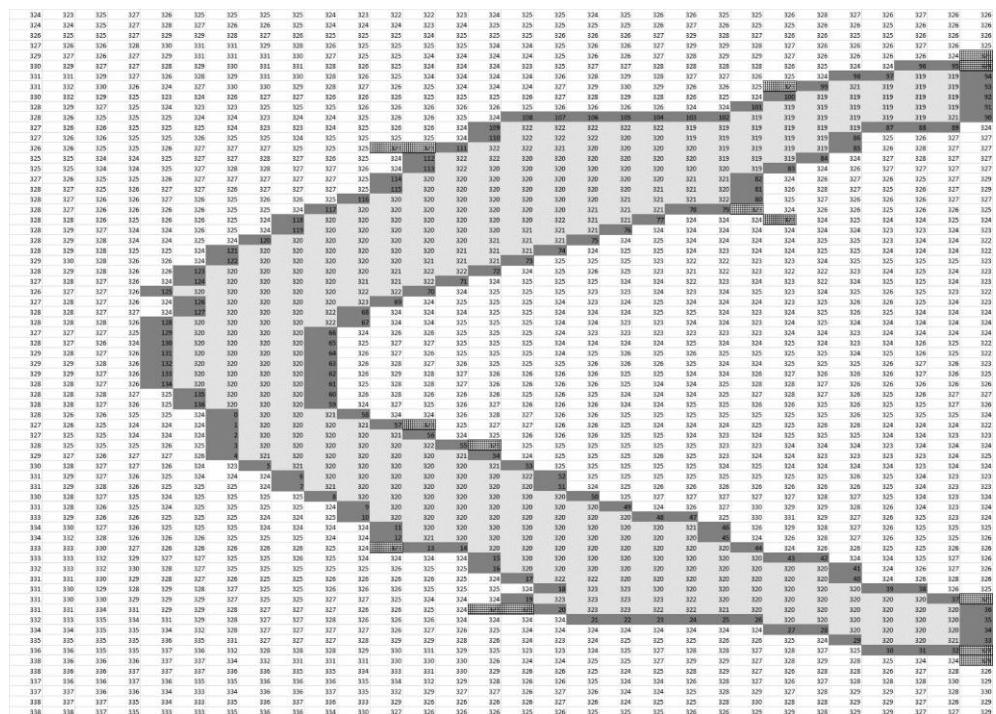


Рис. 2. Пример расчетной области затопления (серый цвет – внутренние ячейки области затопления, темный цвет – ячейки для определения полигона; забракованные пограничные ячейки заштрихованы прямоугольной сеткой).

Таким образом, предлагаемая технология формирования состоит в следующем. Из семейства точек затопления находим пограничные точки. Из их числа удаляем те, которые не соседствуют с внутренними точками.

Из оставшихся формируем один или несколько полигонов по вышеуказанному алгоритму. Для нескольких полигонов очень важно, чтобы движение точек полигонов было одинаковым по отношению к внутренней области затопления. То есть оно было либо по часовой стрелке относительно затопленной области, либо против часовой стрелки.

Перейдем теперь к задаче программного создания shape-слоев из полученных полигонов. На языке Python для создания и манипулирования shape-слоями используются библиотеки “shapely” [3] и “Fiona” [4, 5]. Чтобы загрузить указанные библиотеки в программную среду Python, в командном окне Windows следует выполнить следующие команды:

```
> pip install shapely
> pip install fiona
```

В Python-программе для создания shape-слоя нужно импортировать эти библиотеки:

```
import fiona
import shapely
```

Вначале нужно создать пустой список и заполнить его кортежами с координатами точек полигона. Программно это выглядит так:

```
Coord=[] #создание пустого списка
ShCoord=(Dolg,Shir)
Coord.append(ShCoord) #добавление координат отдельной точки полигона
```

Здесь ShCoord – кортеж с координатами ShCoord одной точки полигона (Dolg – долгота, Shir – широта). Dolg и Shir выражены в десятичных градусах.

Прежде, чем открыть объект shape-слоя, необходимо определить так называемую схему этого объекта командой:

```
schema={'geometry':'Polygon','properties':[('Name','str')]}  
Свойство 'geometry' указывает на тип создаваемого shape-слоя (в данном случае – это полигон). Название 'Name' представляет название записей этого слоя (оно может быть пустым). После определения схемы объекта можно создать (открыть) и сам объект командой:
```

```
PolygonShp=fiona.open(File, mode='w', driver='ESRI shapefile',
schema=schema, crs="EPSG:4326")  
Здесь File – означает директорию с названием файлов записи проекта shape-слоя, mode='w' – означает режим записи, driver='ESRI shapefile' – название драйвера создания shape-слоя, crs="EPSG:4326" – используемая система координат (4326 – означает Российскую Федерацию).
```

Запись координат точек полигона из списка Coord в этот проект shape-слоя осуществляется через словарь rowDict:

```
rowName=''
rowDict={'geometry':{'type':'Polygon','coordinates':[Coord]},'properties':{'Name':rowName},}
PolygonShp.write(rowDict)
```

Здесь 'type':'Polygon' – означает тип создаваемого shape-слоя (в данном случае полигон), 'coordinates':[Coord] – указывает на список Coord для записи координат точек полигона, 'Name':rowName – название записей координат (в данном случае rowName пустое).

В завершении созданный проект shape-слоя необходимо закрыть:

```
PolygonShp.close()
```

В итоге в директории File мы получаем пять файлов проекта с расширениями: *.cpg, *.dbf, *.prj, *.shp и *.shx. После этого в среду QGIS на соответствующую географическую карту можно добавить этот shape-слой. Для этого в меню QGIS выбирать последовательно Слой=>Добавить слой =>Добавить векторный слой. В появившейся форме в текстовом окне «Векторный набор данных» через кнопку поиска файла найти и открыть соответствующий shape-файл и затем нажать кнопку «Добавить». Полученный векторный слой будет добавлен на географическую карту, как показано на рис. 3.



Рис. 3. Пример автоматически созданного shape-слоя, добавленного в карту QGIS.

Наконец рассмотрим вопрос объединения нескольких однотипных shape-слоев в один слой. На языке Python такое объединение можно осуществить с помощью компонента ogr библиотеки gdal [6]. Для этого вначале нужно скопировать с сайта <https://pconlife.com> файл osgeo4w-setup.exe [7]. Затем загрузить указанную библиотеку в программную среду Python. Для этого в командном окне Windows следует выполнить следующую команду:

```
> pip install gdal
```

Для программного использования компонента ogr для манипуляции с shape-слоями в своей программе необходимо сначала импортировать этот компонент командой:

```
from gdal import ogr
```

Общий вид программного кода по объединению нескольких shape-слоев в один shape-слой выглядит так:

```
outputMergefn="Merge.shp"
directory="C:/Output"
filestartswith="C"
FileEndsWith=".shp"
drivername="ESRI Shapefile"
geometrytype=ogr.wkbPolygon
ptdriver=ogr.GetDriverByName("ESRI Shapefile")
if os.path.exists(outputMergefn):
    ptdriver.DeleteDataSource(outputMergefn)
out_ds=ptdriver.CreateDataSource(outputMergefn)
out_layer=out_ds.CreateLayer(outputMergefn,geom_type=geometrytype)
filelist=os.listdir(directory)
for file in filelist:
    if file.startswith(filestartswith) and file.endswith(FileEndsWith):
        print file
        ds=ogr.Open(directory+"/"+file)
        if ds is None:
            continue
        lyr=ds.GetLayer()
        for feat in lyr:
            out_feat=ogr.Feature(out_layer.GetLayerDefn())
            out_feat.SetGeometry(feat.GetGeometryRef().Clone())
            out_layer.CreateFeature(out_feat)
        out_layer.SyncToDisk()
```

где outputMergefn – название конечного (объединенного) shape-слоя; directory – директория, в которой расположены исходные файлы shape-слоев и файлы конечного shape-слоя; filestartswith – начальные буквы названий файлов исходных shape-слоев (по которым будет осуществляться их поиск); FileEndsWith – расширение файлов исходных shape-слоев; drivername – название драйвера формирования shape-слоя; geometrytype – тип геометрии shape-слоя; ptdriver – собственно драйвер shape-слоя, создаваемый на основе компонента ogr; out_ds – источник данных shape-слоя, создаваемый с помощью драйвера ptdriver; out_layer – shape-слой, создаваемый на основе данных out_ds; filelist – список всех файлов, расположенных в директории directory; ds – источник данных исходного shape-слоя; lyr – исходный shape-слой; feat – свойство shape-слоя.

Заключение. В заключение можно сказать, что представленная технология:

- на основе данных уровней воды, географических координат речного створа, размеров расчетной области и данных цифровой модели рельефа местности находит совокупность точек затопленной территории;
- выделяет из указанной совокупности точек пограничные и внутренние точки. Из числа пограничных точек удаляет те, которые не соседствуют с внутренними точками;
- формирует из оставшихся пограничных точек один или несколько полигонов, охватывающих область затопления;
- создает на основе полученных полигонов файлы проекта shape-слоя;
- добавляет сформированный shape-слой на географическую карту QGIS;
- объединяет несколько однотипных shape-слоев в один shape-слой.

Поступила 04.04.2025

Получена с рецензии 26.05.2025

Утверждена 18.05.2025

ЛИТЕРАТУРА

1. Краткое введение в ГИС. Дата обращения 03.03.2025.
https://docs.qgis.org/3.34/ru/docs/gentle_gis_introduction/data_capture.html
2. Создание полигональных объектов. Дата обращения 03.03.2025.
<https://pro.arcgis.com/ru/pro-app/latest/help/editing/create-polygon-features.html>
3. Sean Gillies. *The Shapely User Manual* (электронный текст). Персональный сайт С. Джилиеса (2025). Дата обращения 03.03.2025.
<https://shapely.readthedocs.io/en/stable/manual.html>
4. Sean Gillies. *The Fiona User Manual* (электронный текст). Персональный сайт С. Джилиеса (2024). Дата обращения 08.03.2024.
<https://fiona.readthedocs.io/en/latest/manual.html> (дата обращения 03.03.2025).
5. Kartozza. *Reading and Writing ShapeFiles in Python in Fiona* (электронный текст). Персональный сайт Картоза (2024). Дата обращения 03.03.2024.
<https://kartozza.com/blog/python/reading-and-writing-shapefiles-in-python-with-fiona>
6. Setting up GDAL_OGR with FileGDB Driver for Python on Windows Geospatiality. Дата обращения 03.03.2025.
<https://glenbambrick.com>
7. Osgeo4w-setup.exe File Download & Fix for All Windows OS. Дата обращения 03.03.2025.
<https://www.pconlife.com/viewfileinfo/osgeo4w-setup-exe>

Ե. Բ. ՍԵՐԳԵԵՎ

ԶՐԱԵՂԵՂԻ ՏԱՐԱԾՔՆԵՐԻ ՏՐՎԱԳՐԻ
ՍՏԵՂԾՈՒՄ ԵՎ ՄԻԱՎՈՒՄՈՒՄ

Ա մ փ ո փ ո ւ մ

Հոդվածում քննարկվում է ջրհեղեղված տարածքների վեկտորային շերտերի ավտոմատ ստեղծման և միավորման տեխնոլոգիան՝ հիմնվելով զրի մակարդակի տվյալների և տեղանքի ռելիեֆի թվային մոդելի վրա՝

աշխարհագրական քարտեզի վրա ջրհեղեղված տարածքները դիտելու նպատակով: Հառուկ ուշադրություն է դարձվում ջրհեղեղի տարածքների աշխարհագրական կետերից ջրհեղեղի տարածքի պոլիգոնների ձևավորման խնդիրներին: Տրված է ստացված բազմանկյուններից shape-շերտեր ստեղծելու և մի քանի shape-շերտեր մեջի մեջ միավորելու ծրագրի կողմ:

E. B. SERGEEV

PROGRAMMATICALLY CREATING AND COMBINING SHAPE LAYERS OF FLOOD AREAS OF TERRITORIES

Summary

The article discusses the technology of automatically creating and combining vector layers of flood areas based on water level data and a digital terrain model in order to view flooded areas on a geographical map. Special attention is paid to the problems of forming polygons of flood areas from geographical points of flood areas. The program code for creating shape layers from the obtained polygons and combining several shape layers into one shape layer is given.