

ПРОГРАММИРОВАНИЕ ФУНКЦИЙ ВИЗУАЛИЗАЦИИ ГЕОДАННЫХ В СИСТЕМЕ MATHEMATICA

Д. В. Баровик

Анализ и обобщение типичных математических моделей гидродинамики, аэромеханики, тепломассопереноса, геоэкологии позволяют выделить требования к программному сервису графической визуализации геоданных [1; 2]: двумерная и трехмерная графика с поддержкой вывода фронтов и границ зон; функции расчета и вывода изолиний цифровых полей и карт плотностей; вывод векторных полей скоростей, демонстрация эволюции (анимация).

Система Mathematica содержит набор стандартных средств для графического представления перечисленных видов данных [3]. Особенностью этих средств является то, что отображение происходит в прямоугольной области (для 3D - в параллелепипеде). При этом нет возможности выделения подобластей и включений.

Подобласть – участок (часть плоскости), ограниченный заданным во входной карте (схеме) не имеющим самопересечений контуром (ломаной) со специально для него назначенными атрибутами. Основным смыслом введения понятия подобласти обусловлен особенностями аппроксимации, обработки в ней сеточных функций.

Включение – участок, ограниченный заданным не имеющим самопересечений контуром (ломаной), целиком лежащий внутри подобласти. Во включениях расчет цифрового поля, формирование изолиний, карт плотностей, векторных полей не производится.

ВИЗУАЛИЗАЦИЯ КАРТ В ПОДОБЛАСТЯХ С КРИВОЛИНЕЙНОЙ ГРАНИЦЕЙ

Предлагается следующая методика для визуализации контурных карт и карт плотностей в односвязных областях, не являющихся прямоугольными. Пользователь задает границы области визуализации в виде списка координат вершин замкнутого многоугольника (ограничения на выпуклость многоугольника не накладываются). Обозначим эти координаты через (x_0, y_0) , (x_1, y_1) , ..., (x_n, y_n) , где $x_0 = x_n$, $y_0 = y_n$. Так же описываются ширина и высота ($x_{\max}$ и $y_{\max}$ соответственно) прямоугольной карты, построенной стандартными средствами Mathematica, такими как ContourGraphics или DensityGraphics. Формирование изображения в подобласти обеспечивает алгоритм, который возвращает координаты вершин двух замкнутых многоугольников, «обрамляющих» область визуализации. При наложении этих многоугольников цветом фона на обычное изображение изолиний или карту плотностей, формируемых функциями



Рис. 1. Пример изолиний и карты плотности по подобласти

системы, достигается эффект визуализации заданной карты в требуемой области. Достоинством такой методики является то, что ее можно применять к изображениям (картам), задаваемым стандартными графическими объектами системы Mathematica. При этом, можно использовать любые опции (свойства, настройки) этих графиков: масштабирование, окраска, вывод координатных осей, линий сетки, ее калиб-

ровки, надписей, заголовков, легенды и т. п.

Суть алгоритма заключается в нахождении таких двух точек (x_{start}, y_{start}) и (x_{end}, y_{end}) на множестве вершин (x_i, y_i) , $i = 0, n$, что два отрезка $[(0,0), (x_{start}, y_{start})]$ и $[(x_{max}, y_{max}), (x_{end}, y_{end})]$ не пересекаются с границей области визуализации (очевидно, за исключением общих точек). Координаты вершин первого искомого многоугольника будут следующие: (x_{start}, y_{start}) , $(x_{start}+1, y_{start}+1)$, ..., $(x_{end}-1, y_{end}-1)$, (x_{end}, y_{end}) , (x_{max}, y_{max}) , $(0, y_{max})$, $(0,0)$, (x_{start}, y_{start}) . Координаты вершин второго: (x_{end}, y_{end}) , $(x_{end}+1, y_{end}+1)$, ..., $(x_{start}-1, y_{start}-1)$, (x_{start}, y_{start}) , $(0,0)$, $(x_{max}, 0)$, (x_{max}, y_{max}) , (x_{end}, y_{end}) .

Наложение многоугольников на прямоугольный график цветом фона производится непосредственно функцией `Show` или через ее опцию `Epilog`. Функцией `Line` может быть выведена указываемым стилем и граница области. Также функцией `Show` (или ее опцией `Epilog`) на полученную карту выводятся задаваемыми цветами и стилями всевозможные контура, границы, линии разметки, а если так добавленную подобласть закрасить цветом фона, получим подобласть типа «дырка» (визуализация в многосвязных областях).

На рис. 1 приведен пример визуализации изолиний и зон в подобласти (граница подобласти – контур Брестской области РБ, легенда не выводится, т.к. цифровое поле не конкретизируется), причем кроме изолиний на схеме показаны нанесенные внутренние контура-границы (в данном примере границы административных районов).

ВИЗУАЛИЗАЦИЯ ВЕКТОРНЫХ ПОЛЕЙ

Для отображения векторных полей в областях сложной конфигурации используется следующий прием. Векторное поле формируется одной из стандартных функций (например, `ListPlotVectorField`), далее в полученном объекте `Graphics` удаляются все векторы (объекты `Arrows`), которые

не попадают в область моделирования (т.е. удаляются все вектора вне подобласти).

На *рис. 2* приведен пример визуализации векторного поля, изолиний и зон в подобласти с включением. В примере подобласть – схематично изображенная площадь Республики Беларусь, включение – Минская область.

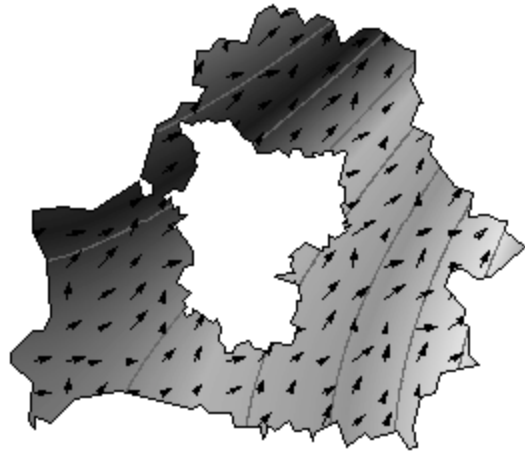


Рис. 2. Пример карты векторного поля по подобласти с включением

Для реализации таких алгоритмов обработки стандартного изображения запрограммирована библиотека *Modelling.m*. Приведем описание основных и вспомогательных функций в реализации пакета, имеющих также самостоятельное значение для некоторых других задач визуализации данных:

`showBorder[gRaphics, borderList, toShow, oPts]` выполняет визуализацию графиков `gRaphics` в области `borderList`. Здесь: `gRaphics` – графический объект, аналогичный первому параметру функции `Show`; `borderList` – список координат вершин многоугольника, задающего границу (контур) двумерной области визуализации; `toShow` – необязательный параметр, принимающий значения `True` (по умолчанию) или `False`; в случае значения `False` визуализация полученной карты на экране не происходит, а вывод происходит с опцией `DisplayFunction -> Identity`; `oPts` – необязательные параметры – опции отображения, которые передаются в функцию `Show` при отображении графиков.

`getRegions[{region1Border, ...}, {region1Type, ...}, {region1BackgroundColor, ...}, {region1BorderStyle, ...}]` возвращает объект типа `Graphics`, представляющий собой совокупность подобластей определенного типа, цвета фона, стиля и цвета границы.

Типы подобластей (возможные значения второго параметра): `regionTypeSolid` (по умолчанию) – подобласть заливается указанным цветом; `regionTypeHole` – подобласть заливается цветом фона `backgroundColor`, даже если явно указан другой цвет; `regionTypeNone` – подобласть является прозрачной, т.е. выводится только граница.

`borderListPlot[listPlot, borderList, toShow, opts]` выполняет визуализацию графиков `DensityListPlot` или `ContourListPlot` в области визуализации `borderList`.

`IsOutsideOfPolygon[gran, xi, yi, isConvex]` – функция проверяет, лежит ли точка (xi, yi) строго внутри замкнутого многоугольника; возвращает `True` или `False`. `gran` – координаты вершин замкнутого многоугольника;

```

(* Function deletes all the vectors from ListPlotVectorField,
   which start point doesn't strictly lay inside of the border listBorder *)
(* Returns cleared ListPlotVectorField *)
deleteVectorsOutsideBorder[listPlotVectorField_, listBorder_] :=
Module[{temp}, temp = listPlotVectorField;
  len = Length[listPlotVectorField[[1, 2]]];
  For[i = len, i >= 1, i--,
    x = listPlotVectorField[[1, 2, i, 1, 1, 1]];
    y = listPlotVectorField[[1, 2, i, 1, 1, 2]];
    If[True == IsOutsideOfPolygon[listBorder, x, y],
      temp[[1, 2, i]] = {}
    ];
  ];
Return[temp];
];

```

Рис. 3 Описание программного кода функции *deleteVectorsOutsideBorder*

x_i , y_i – координаты проверяемой точки; *isConvex* – необязательный параметр, принимающий значения True или False (по умолчанию), указывающий является ли многоугольник выпуклым; в случае True, используется более быстрый алгоритм проверки.

deleteVectorsOutsideBorder[*vectorFieldPlot*, *borderList*] удаляет из графического объекта *vectorFieldPlot*, все вектора, которые не попадают в область визуализации *borderList*. На рис. 3 приведен программный код ее реализации:

Литература

1. Гришин А. М. Общие математические модели лесных и торфяных пожаров и их приложения // Успехи механики. 2002. № 4. С. 41–89.
2. Кулешов А. А. Математическое моделирование в задачах промышленной безопасности и экологии // Информационные технологии и вычислительные системы. 2003. № 4. С. 56–70.
3. Морозов А. А., Таранчук В. Б. Программирование задач численного анализа в системе Mathematica: Учеб. пособие. Мн.: БГПУ, 2005. 145 с.

ИССЛЕДОВАНИЕ АЛГОРИТМОВ КОРРЕКЦИИ ПРОГНОЗОВ НА ОСНОВЕ МОДЕЛЕЙ СО СТРУКТУРНЫМ ИЗМЕНЕНИЕМ В ПРОГНОЗНОМ ПЕРИОДЕ

А. В. Бояр

ВВЕДЕНИЕ

Структурные изменения – один из типовых признаков макроэкономических временных рядов, особенно для переходных экономик [2]. Структурные изменения в периоде оценивания модели обычно учитываются,