Programming assignment #2: Municipal Boundaries in State Plane Coordinates

Date assigned: February 10, 2011

Date due: February 17, 2011 (before class)

The state plane coordinate system is a convenient tool that provides direct mapping between a local Cartesian coordinate system and the global spherical coordinate system of latitude and longitude. In this assignment, we will use a provided Matlab function to convert latitude and longitude to the Texas Central Zone state plane coordinates based on the North American Datum of 1983 (NAD-83). Once points are mapped to the Texas Central Zone, simple Cartesian geometry will be used to determine whether a point is within the city limits of Wixon Valley, Texas. Section 1 introduces more engineering background and a description of Wixon Valley, Texas. The details of the programming assignment are provided in Section 2, followed by the challenge problem in Section 3.

1 Permitting and Legal Land Descriptions

A large industrial company was applying for well permits for a large number of wells. Each well had to be plotted on a map and the relevant legal authority identified. Since the jurisdiction of each legal authority was easily mapped by a series of intersecting lines, a program can be created to compare the coordinates of each well to each authority’s boundaries and determine the relevant legal authority. Figure 1 shows a Google-Earth image of Wixon Valley, Texas, which is 9.5 mi north, north-east of Bryan/College Station, Texas. Table 1 provides the latitude and longitude of each corner of the Wixon Valley city limits.

<table>
<thead>
<tr>
<th>Latitude</th>
<th>Longitude</th>
</tr>
</thead>
<tbody>
<tr>
<td>30°45'52.15''</td>
<td>96°18'43.86''</td>
</tr>
<tr>
<td>30°45'48.37''</td>
<td>96°18'48.47''</td>
</tr>
<tr>
<td>30°45'13.02''</td>
<td>96°17'50.67''</td>
</tr>
<tr>
<td>30°44'49.40''</td>
<td>96°18'14.27''</td>
</tr>
<tr>
<td>30°45'39.66''</td>
<td>96°19'15.87''</td>
</tr>
<tr>
<td>30°45'33.23''</td>
<td>96°19'23.63''</td>
</tr>
<tr>
<td>30°45'40.55''</td>
<td>96°19'33.01''</td>
</tr>
<tr>
<td>30°45'49.05''</td>
<td>96°20'09.04''</td>
</tr>
<tr>
<td>30°45'50.41''</td>
<td>96°20'12.11''</td>
</tr>
<tr>
<td>30°45'53.15''</td>
<td>96°20'16.76''</td>
</tr>
<tr>
<td>30°46'16.77''</td>
<td>96°20'26.61''</td>
</tr>
<tr>
<td>30°46'22.40''</td>
<td>96°20'5.62''</td>
</tr>
<tr>
<td>30°46'45.73''</td>
<td>96°19'24.77''</td>
</tr>
</tbody>
</table>

In engineering applications, longitude/latitude based coordinates are not common, as the distance between two points or areas of polygons cannot be easily calculated. Also, the longitude/latitude system is for representing points on a sphere. The points need to be mapped to a flat surface using some sort of map projection methods. The map projection from latitude/longitude system to other coordinate system can be found in a lot of geography software packages.
In this exercise, we will map the longitude/latitude coordinates to Texas State Plane coordinates, using the Matlab function gcpg83.m provided on the class website. This function is an excerpt from the FORTRAN program spcs83.exe, distributed by the National Geodetic Survey and available for download from http://www.ngs.noaa.gov/TOOLS/spc.html. The program provided here is only valid for the Texas Central Zone NAD-83 state plane coordinate system. In this coordinate system, the north-south position of a point on a map is termed “northing”, in units of meters, and the east-west position is termed “easting”, which is in units of meters as well (see Figure 2).

Figure 1: City Limits of Wixon Valley. Image from Google Earth, Copyright 2009 Tele Atlas.
2 Assignment

Write a Matlab computer program to 1) generate 100 random points in decimal latitude and longitude, 2) convert them to the state plane coordinate system, 3) classify the points as within or not within a special area in the City. The plane coordinates of the corners of the area of interest are listed in Table 2. The first is easting and the second northing.

### Table 2: Plane Coordinates of the Four Corners for the trapezoid Area of Interest

<table>
<thead>
<tr>
<th>Easting (m)</th>
<th>Northing (m)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.0830E+06</td>
<td>3.1295E+06</td>
</tr>
<tr>
<td>1.0838E+06</td>
<td>3.1295E+06</td>
</tr>
<tr>
<td>1.0843E+06</td>
<td>3.1285E+06</td>
</tr>
<tr>
<td>1.0830E+06</td>
<td>3.1285E+06</td>
</tr>
</tbody>
</table>

You are provided function gcpg83.m (available from the class website), which takes latitude and longitude in decimal degrees as input and convert them into Texas State Plane coordinates. The data in Table 1 are provided for download in the file WV Boundary.dat (available from the class website). You are also provided a Matlab function (get_points.m) to generate 100 random points near Wixon Valley, Texas, for classification. Finally, you are provided a third Matlab function that determines whether a point is inside a polygon or not (isinsidepolygon.m). You need to read the provided functions carefully to understand how they work, what input arguments are needed and what the outputs from the functions are.

Your program should create a plot of the city limit boundaries and the special Area of Interest together with all of the processed points, coloring the points inside the area of interest as red stars and those outside as blue dots. Figure 2 illustrates the output from the working solution program.

![Figure 2: Example solution to the problem.](image)

Organize your results using the memorandum template from the Assignments page of the course web site. In the body of the memorandum, you should include a copy of the final figure produced by your working computer code and a listing of your Matlab program as an Appendix. Be sure your program is correctly indented and include sufficient descriptive text for me to understand your program.
To help you get started, the following provides a skeleton for your completed program:

```matlab
% PROGRAM Wixon_Valley.m
%
% This program reads in the locations of the boundary corners defining
% Wixon Valley, Texas, converts them to state plane coordinates, and then
% checks whether random points are inside or outside city limits.
%
% Define Variables:
% Fill in with appropriate text
% PROGRAM Wixon_Valley.m
%
% This program reads in the locations of the boundary corners defining
% Wixon Valley, Texas, converts them to state plane coordinates, and then
% checks whether random points are inside or outside city limits.
%
% Define Variables:
% Fill in with appropriate text
%
% Full name
% UIN
% Date
% Load in the way points
MV_Boundary=load('WV_Boundary.dat');
% Convert points to decimal degrees
%
% Load in points to classify
%
% Convert points to state plane coordinates
%
% Classify points
%
% Create plot and put proper labels
```

3 Challenge Question (Optional)
Generate 100 random points again. Group them into 3 categories: a) outside the city limit, b) inside the city limit but outside the area of interest and c) inside the area of interest. Plotting these points on the map using three different types of symbols.