

基于 Visual C#.NET 的空间缓冲区分析开发

赵亚萍

(同济大学,上海 200092)

摘要:随着计算机技术特别是网络、通讯技术的发展,GIS技术得到了飞速发展,并朝着网络化、大众化的方向发展。同时,由于社会对地理信息的巨大需求,地理数据的应用日益广泛。但是空间数据中隐含着大量的知识信息与各类模式,因此,人们迫切需要一些有效的方法来从中提取出一些潜在的、有价值的知识。文中采用 Visual C#.NET 与 ArcObjects 相结合,开发出了空间缓冲区分析模块,实现了对空间缓冲区分析的功能,通过对其点、线、面的缓冲区分析,从而高效、直观地提取出隐含在空间数据中的信息。

关键词:GIS;空间数据;空间缓冲区;空间缓冲区分析

中图分类号:TP311

文献标识码:A

文章编号:1673-629X(2009)12-0029-03

Spatial Buffer Analysis Development Based on Visual C#.NET

ZHAO Ya-ping

(Tongji University, Shanghai 200092, China)

Abstract: With the development of computer technology, especially the network and communication technology, GIS technology has made rapid development, and developed in the direction of network and public. At the same time, because of the great demand for geographic information in the community, geographic data is widely used. But there have been increasing demands for efficient methods that extract rules and patterns from spatial data. This paper gets the module and foundations of spatial buffer analysis with Visual C#.NET and ArcObjects. After buffer analysed the points, lines, polygons, can get more potential information efficiently and intuitionistically from the spatial data.

Key words: GIS; spatial data; spatial buffer; spatial buffer analysis

0 引言

空间分析方法^[1]是一种利用 GIS^[2,3]的各种空间分析模型和空间操作对 GIS 数据库中的数据^[4]进行深加工,从而产生新的信息和知识;通过空间分析,不但可以知道数据库中的数据,而且可以通过这些数据去揭示更深刻、更内在的规律和特征,它特有的对地理信息(特别是隐含信息)的提取、表现和传输功能,是地理信息系统区别于一般信息系统的主要功能特征。常用的空间分析方法^[5]有综合属性数据分析、拓扑分析、缓冲区分析^[6]、距离分析、叠置分析、地形分析、趋势面分析、预测分析等,可发现目标在空间上的相连、相邻和共生等关联规则,或发现目标之间的最短路径、最优路径等辅助决策知识。

文中利用 Visual C#.NET^[7]与 ArcObjects 相结合,对 ArcGIS 进行二次开发,实现了对空间分析方法

之一——空间缓冲区分析的功能。

1 空间缓冲区与空间缓冲区分析方法

所谓缓冲区就是地理空间目标的一种影响范围或服务范围,图1为点对象、线对象、面对象及对象集合的缓冲区示例。

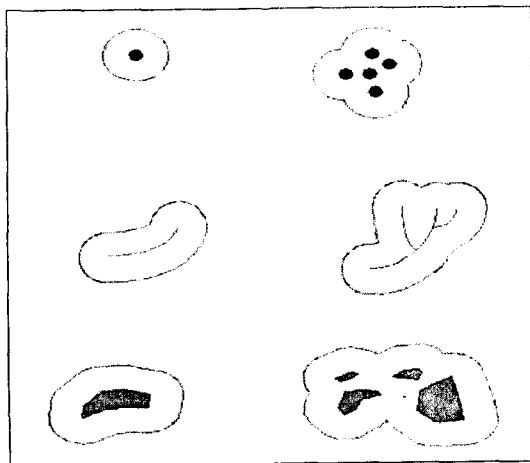


图1 点、线、多边形的缓冲区示例图

收稿日期:2009-04-02;修回日期:2009-07-18

作者简介:赵亚萍(1965-),女,内蒙古乌兰察布人,硕士研究生,乌兰察布职业学院经济师,研究方向为软件工程。

空间缓冲区分析^[8]根据用户需要给定一个点缓冲、线缓冲或面缓冲的距离,从而形成一个缓冲区的多边形,系统检索出位于该缓冲区多边形内的所有的空间对象,显示所检索到的空间对象的几何参数信息和与之关联的属性信息。

2 空间缓冲区分析模块设计

可以用于空间缓冲区分析的几何类型有:点、线、面(圆、矩形、多边形),下面分别对之进行介绍(如图 2 所示):

(1)点缓冲区分析:用户在图形区域中选择一组点状地物或一类点状地物或一层点状地物,根据用户给定的缓冲区距离,系统自动形成点缓冲区多边形图层,系统再将此点缓冲区图层和其它指定的做空间叠置分析。

(2)线缓冲区分析:用户在图形区域中选择一类或一层的线状空间地物,根据用户给定的缓冲距离,系统自动形成线缓冲区多边形图层,系统再将此线缓冲区图层和其它指定的做空间叠置分析。

(3)面缓冲区分析:用户在图形区域中选择一类或一层面状地物,根据用户给定的缓冲区距离,系统自动形成面缓冲区多边形图层。面缓冲区有外缓冲区和内缓冲区之分,外缓冲区仅在面状地物的外围形成缓冲区,内缓冲区则在面状地物的内侧形成缓冲区,当然也可以在面状地物的边界两侧均形成缓冲区。

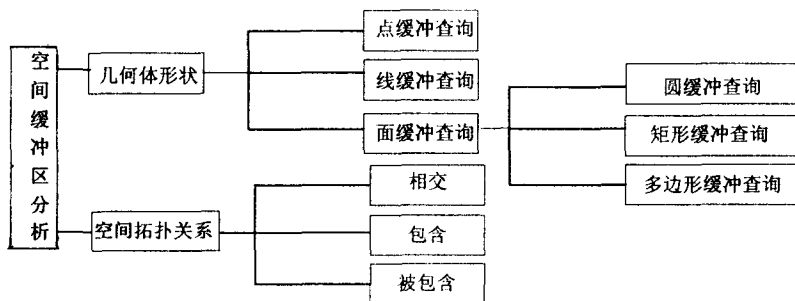


图 2 空间缓冲区分析功能模块图

可以用于空间缓冲区分析的空间拓扑关系类型有:相交、包含、被包含三种空间关系,这三种空间关系比较常见,也比较简单,在此不再多做解释。

3 程序界面及功能

3.1 几何形状和空间拓扑关系简介

3.1.1 几何形状和空间拓扑关系界面图简介

可以用于空间缓冲区分析的几何类型有(如图 3 所示):点、线、面(圆、矩形、多边形),上节分别对之进行了详细介绍;可以用于空间缓冲区分析的空间拓扑关系类型^[9]有:相交、包含、被包含三种空间关系,这

种空间关系比较常见,也比较简单。

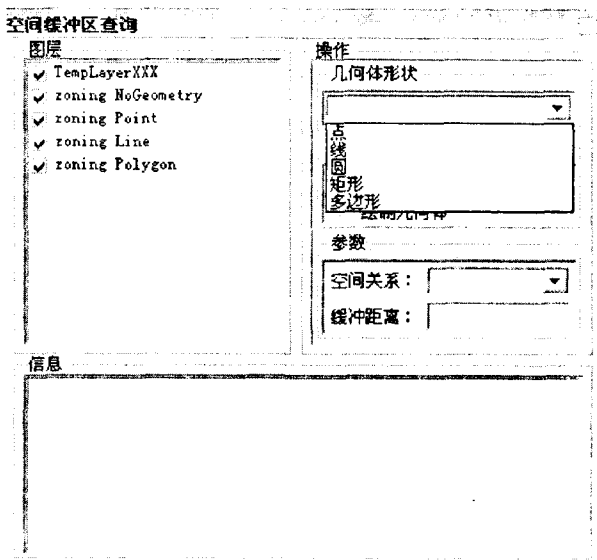


图 3 界面简介图

3.1.2 简要代码

```
this.label2.AutoSize = true;
this.label2.Location = new System.Drawing.Point(3, 32);
this.label2.Name = "label2";
this.label2.Size = new System.Drawing.Size(65, 12);
this.label2.TabIndex = 0;
this.label2.Text = "缓冲距离:";
this.label1.AutoSize = true;
this.label1.Location = new System.Drawing.Point(3, 7);
this.label1.Name = "label1";
this.label1.Size = new System.Drawing.Size(65, 12);
this.label1.TabIndex = 0;
this.label1.Text = "空间关系:";
this.groupBox5.Text = "获取几何体";
this.radioButton2.TabIndex = 0;
this.radioButton2.TabStop = true;
this.radioButton2.Text = "绘制几何体";
this.radioButton2.UseVisualStyleBackColor =
this.radioButton1.AutoSize = true;
this.radioButton1.TabStop = true;
this.radioButton1.Text = "选择几何要素";
this.radioButton1.UseVisualStyleBackColor = true;
this.groupBox4.Controls.Add(this.panel4);
this.groupBox4.TabStop = false;
this.groupBox4.Text = "几何体形状";
this.toolStripButton1.DisplayStyle = System.Windows.Forms.
ToolStripItemDisplayStyle.T
```

3.2 属性表选项

3.2.1 属性表选项界面图

在要素范围中选择“全部要素”或者“仅选择要素”

选项,且在属性表类型中选择“普通属性表”或者“属性报表”选项,就可以根据需要生成报表(如图 4 所示)。

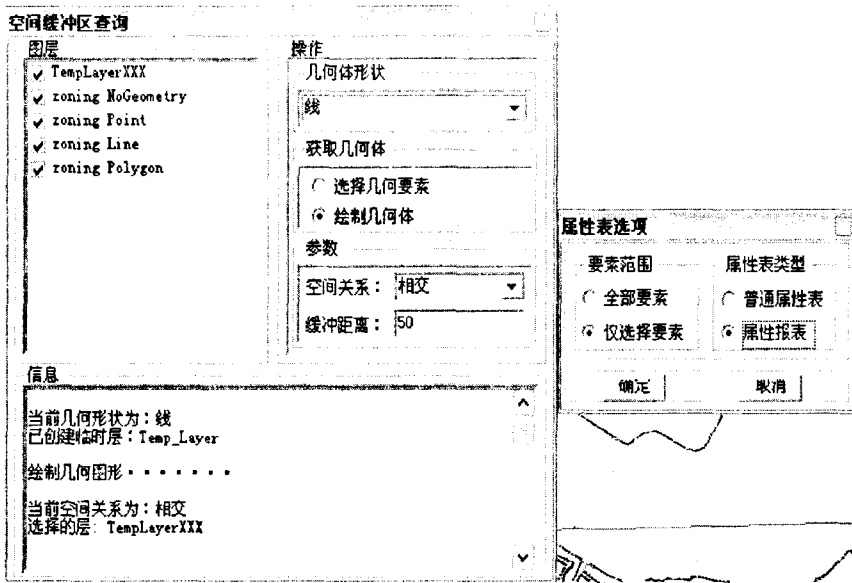


图 4 属性表选项图

3.2.2 简要代码

```
this.toolStripButton1.Name = "toolStripButton1";
this.toolStripButton1.Size = new System.Drawing.Size(55,16);
this.toolStripButton1.Text = "分析";
this.toolStripButton2.Size = new System.Drawing.Size(55,16);
this.toolStripButton2.Text = "属性表";
this.toolStripButton3.Text = "清除选定";
this.toolStripButton4.ImageTransparentColor = System.Drawing.Color.Magenta;
this.toolStripButton4.Name = "toolStripButton4";
this.toolStripButton4.Size = new System.Drawing.Size(33,16);
this.toolStripButton4.Text = "取消";
this.textBox1.Dock = System.Windows.Forms.DockStyle.Fill;
this.textBox1.Location = new System.Drawing.Point(0,0);
this.Controls.Add(this.groupBox3);
this.Controls.Add(this.groupBox2);
this.Controls.Add(this.groupBox1);
this.FormBorderStyle = System.Windows.Forms.FormBorderStyle.FixedToolWindow;
this.Name = "BufferAnalysisFrm";
this.StartPosition = System.Windows.Forms.FormStartPosition.CenterScreen;
this.Text = "缓冲区查询";
```

4 空间缓冲区分析实验

在选中或者绘制地物的前提下,选择空间关系,输入缓冲的距离,点击“查询”按钮后,即会显示出缓冲区结果和显示缓冲区范围,并高亮显示当前缓冲区内的地物。

几何类型以面为例(点缓冲区分析与线缓冲区分析类似,不再赘述),面缓冲区有外缓冲区和内缓冲区之分,外缓冲区仅在面状地物的外围形成缓冲区,内缓冲区则在面状地物的内侧形成缓冲区,当然也可以在面状地物的边界两侧均形成缓冲区。

因此在面缓冲区分析时增加了一个对话框选项,选项分别为:多边形边界外缓冲、多边形边界内缓冲、多边形边界内外缓冲、多边形内缓冲和普通多边形缓冲。其运行情况与点缓冲区分析、线缓冲区分析类似,运行结果如图 5 所示。

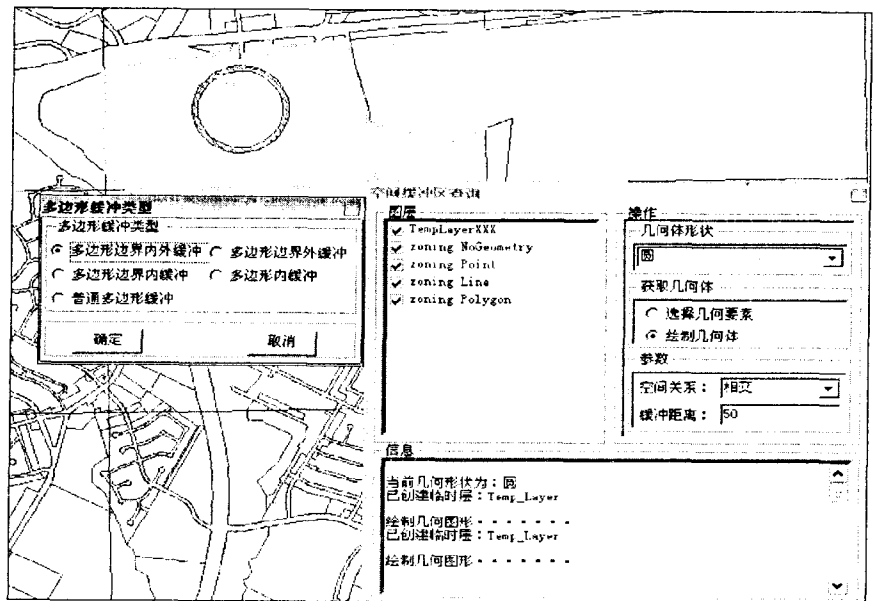


图 5 多边形缓冲区分析图

5 结束语

文中利用 Visual C#.NET 与 ArcObjects 相结合进行了对 ArcGIS 的二次开发,实现了对点、线、面缓冲区相交、包含、被包含三种空间关系分析的功能。并进行了实验,从实验的运行情况看,通过对其点、线、面的缓冲区分析,可以高效、直观地提取出隐含在空间数据中的信息,实现了空间数据有用信息分析。

(下转第 35 页)

变慢,同时,算法收敛精度不高,尤其是对于高维多极值的复杂优化问题^[8]。这里仅模拟一维、二维、三维的情况,文献[5]证明了佳点集在高维下优点更突出,所以模拟结果会更好。在函数 1,3,4 中精度 pre=2,而函数 2,5 精度 pre=3,比较平均迭代次数和最优值。

表 1 计算结果比较表

函数	平均迭代次数			最优值		
	无交叉	单点交叉	佳点集交叉	无交叉	单点交叉	佳点集交叉
1	8.5	6.7	1.04	1.21899	1.21899	1.21899
2	10.2	9.1	8.6	1048560	1048573	1048580
3	16.4	17.6	9.5	0.148147	0.148147	0.148147
4	3.5	5.6	1.02	78.6432	78.6432	78.6432
5	2.7	9.8	4.3	0.011952	0.009477	0

从表 1 可以看出,无论是收敛速度还是精度采用佳点集交叉的算法所得的结果都有不同程度的提高和改善。算法的全局收敛性、搜索效率和算法的稳定性都得到了提高和增强。

在实验中还发现对于多峰值函数在佳点交叉算法中参数精度越高所得的结果越好,只是精度越高对应着交叉时的编码越长,空间和时间耗费都会增加,因此在具体问题时应在这两者之间折衷。同时无交叉的粒子群算法在 100 次模拟中会有更多次数的达不到最优,单点交叉和佳点交叉相对会多次重现最优值。而相对单点交叉,佳点交叉又具有较少的迭代次数并提高优化精度。

4 结束语

文中针对基本粒子群算法容易陷入局部极值,后期收敛速度慢和精度低等缺点,基于数论中的佳点集理论和方法即把佳点集算法用于交叉中,文中提出了基于佳点集交叉的粒子群算法,并从试验上证明了将

佳点集交叉算法用于粒子群算法的优化问题上无论从精度还是速度上都比其他算法好,尽管时间复杂度上有所增加,但是其优化却达到了很好的效果。粒子群算法总体表现收敛速度快,在试验中则表现为迭代次数少。即便在收敛速度相对较快的情况下采用佳点集交叉仍然能一定程度减少迭代次数并提高精度,进一步提高了粒子群算法的优化效率。尽管文中提出的算法相对基本粒子群算法增加了交叉概率这个参数,但这也给试验以更大的实施空间,是一种有效可行的方法。

参考文献:

[1] Kennedy J, Eberhart R C. Particle Swarm Optimization[C]//Proc. IEEE International Conference on Neural Networks, IV. Piscataway, NJ: IEEE Service Center, 1995: 1942-1948.

[2] Chen Debao, Zhao Chunxi. Particle swarm optimization with adaptive population size and its application[J]. Applied Soft Computing, 2009(9): 39-48.

[3] 苏守宝, 汪继文, 方杰. 粒子群优化技术的研究与应用进展[J]. 计算机技术与发展, 2007, 17(5): 249-253.

[4] Shoubo S, Shuho Y V. Particle Swarm Optimization Based on Good Point - Set[C]//International Symposium on Intelligence Computation & Applications. Wuhan, China: [s. n.], 2005.

[5] 张铃, 张钊. 佳点集遗传算法[J]. 计算机学报, 2001, 24(9): 917-922.

[6] 张丽平, 陈德钊. 粒子群优化算法的理论与实践[D]. 杭州: 浙江大学, 2005.

[7] 刘洪波, 王秀坤, 谭国真. 粒子群优化算法的收敛性分析及其混沌改进算法[J]. 控制与决策, 2006, 21(6): 636-640.

[8] 胡旺, 李志蜀. 一种更简化而高效的粒子群优化算法[J]. Journal of Software, 2007, 18(4): 861-868.

(上接第 31 页)

参考文献:

[1] 李德仁, 王树良, 李德毅. 空间数据挖掘理论与应用[M]. 北京: 科学出版社, 2006.

[2] 邬伦. 地理信息系统—原理、方法和应用[M]. 北京: 科学出版社, 2001.

[3] 黄镇谨, 李春贵, 欧阳浩. 基于 GPS 和 GIS 的公共交通监控平台[J]. 计算机技术与发展, 2009, 19(3): 16-18.

[4] Han J. Conference tutorial notes: data mining techniques [C]//In: Proceedings of ACM SIGMOD International Conference'96 on Management of Data (SIGMOD'96). Montreal, Canada: [s. n.], 1996.

[5] Chen M S, Han J W, Yu P S. Data Mining: An Overview from Database Perspective[J]. IEEE Transaction on Knowledge and Data Engineering, 1996, 8(6): 866-883.

[6] 周海燕, 王家耀, 吴升. 空间数据挖掘技术及其应用[J]. 测绘通报, 2002(2): 25-26.

[7] 胡顺扬, 瞿有甜, 周波, 等. 基于 .NET 平台的软件构件开发方法研究[J]. 计算机技术与发展, 2008, 18(11): 9-11.

[8] Yao Xin, Xu Yong. Recent Advances in Evolutionary Computation[J]. Computer Science & Technology, 2006, 21(1): 1-18.

[9] 王海起, 王劲峰. 空间数据挖掘技术研究进展[J]. 地理与地理信息科学, 2007, 21: 18-20.