

# 基于 IDL 的栅格地图代数实现与应用

卜坤<sup>1,2</sup>,张树文<sup>1</sup>,张宇博<sup>3</sup>

BU Kun<sup>1,2</sup>,ZHANG Shu-wen<sup>1</sup>,ZHANG Yu-bo<sup>3</sup>

1.中国科学院 东北地理与农业生态研究所,长春 130012

2.中国科学院 研究生院,北京 100049

3.吉林大学 软件学院,长春 130012

1.Northeast Institute of Geography and Agroecology,CAS,Changchun 130012,China

2.Graduate University of Chinese Academy of Sciences,Beijing 100049,China

3.School of Software,Jilin University,Changchun 130012,China

E-mail:bukunu@gmail.com

**BU Kun,ZHANG Shu-wen,ZHANG Yu-bo.Application of high performance map algebra with IDL.Computer Engineering and Applications,2008,44(9):174-177.**

**Abstract:** ArcGIS offers the powerful function of map algebra,performs the spatial analysis on raster datasets.However,this function becomes the bottleneck in some projects sometimes because of its poor efficiency.In this research,the ArcGIS is thought to perform data resampling before map algebra.IDL is used to realize the function of map algebra here.A model for map algebra is established first,then each raster layer gets its boundary and pixel size according to the model.The raster datasets used in map algebra in IDL are prepared after data resampling.Different algorithm should be desinged according to different application.IDL gives a pretreatment to the data to avoid data resampling in the computation,and reduce the time for map algebra computation.When used in soil erosion computation of SanJiang Plain with USLE equation,this work significant saves time.

**Key words:** IDL;map algebra;spatial analysis

**摘要:** ArcGIS 提供了功能强大的地图代数计算功能,可以有效地针对栅格数据进行空间分析。但是,由于其运算效率较低而成为一些项目应用的瓶颈。研究认为 ArcGIS 进行地图代数计算效率比较低的主要原因在于它在计算时首先对数据进行重采样。通过使用 IDL 提供的栅格数据处理功能,对数据进行预处理,建立进行地图代数计算时的模版,将进行地图代数计算的各个图层依照地图代数模版重新进行范围的分割,并根据原始栅格数据及研究内容确定像元大小,对栅格数据进行重采样,得到可以用 IDL 进行地图代数运算的栅格数据集。根据不同研究领域的的内容,设计地图代数的算法,实现地图代数运算。最后,以使用 USLE 方程计算三江平原土壤侵蚀模数为例,验证了方法的有效性。

**关键词:** IDL;地图代数;空间分析

**文章编号:**1002-8331(2008)09-0174-04 **文献标识码:**A **中图分类号:**TP302.7

## 1 引言

对于模拟具有一定空间内连续分布特点的地理现象来说,基于栅格数据模型的观点是合适的。栅格数据模型是基于连续铺盖的,它将连续空间离散化,即用二维铺盖或划分覆盖整个连续空间<sup>[1]</sup>。栅格数据模型是许多地理现象最好的科学表达方法,大量的地理模型建立与空间分析都是基于栅格数据模型进行的<sup>[2-4]</sup>。地图代数是栅格数据空间分析的语言,它的语法和传统代数相似。地图代数的输入可以是简单的栅格或者属性数据集,操作可以是对每一位置乘以一个变换因子;或者输入可以是一系列栅格或矢量数据集,操作可以是多个栅格数据集或者层的叠加。地图代数允许用户创建复杂的表达式并把其作为一个简单的命令处理。由于地图代数的强大功能,ESRI 公司从 20 世纪 70 年代以来就在 Arcinfo Workstation 的空间分析模块

中提供了这一功能;在后来发布的 ArcGIS 套件的 Workstation 与 Dekstop 版本中,重新以 ArcGIS Spatial Analyst 扩展模块发布。地图代数可以与 ArcGIS Spatial Analyst 的所有功能结合,将多层栅格要素进行结合,执行可行性分析,赋权重和识别关系,是栅格模型分析所必不可少的工具。

在实际应用中,ArcGIS 在执行地图代数分析的过程中计算速度是比较慢的。通过分析发现,导致 ArcGIS 进行栅格地图代数计算速度慢的主要原因是 ArcGIS 在进行地图代数计算时要首先对栅格数据进行重采样。这项技术简化了用户在进行地图代数计算时的理解与操作,但是同时却严重消耗了计算机资源。应用于空间分析的数据一般比较大,尤其在涉及到一些实验的时候,往复调试会导致运行时间非常长。在实际的项目应用中,这种有效的空间分析手段,有时却会由于耗时较长而成

**基金项目:**中国科学院知识创新工程重要方向项目(No.KZCX2-SW-320-1)。

**作者简介:**卜坤(1980-),在读博士,主要从事地图学与地理信息系统应用研究。

**收稿日期:**2007-11-06 **修回日期:**2008-01-07

为项目应用的瓶颈。

本文使用 IDL 对栅格数据的地图代数功能重新进行了实现,避免了空间数据进行地图代数计算之前的重采样,将地图代数计算的时间大大缩短。

## 2 关键技术实现

随着计算机技术的快速发展,人们对满足实际需要的实时交互可视化数据分析工具的需求越来越高。美国 ITT VIS(ITT Visual Information System)公司推出的第 4 代可视化交互数据语言 IDL(Interactive Data Language)是新一代交互式、跨平台(运行于 Unix, VMS, Windows, Macintosh 等),面向对象的应用程序开发语言,具有较强的数据分析和可视化功能。IDL 是进行交互数据分析和可视化应用工程开发的高效软件和理想工具<sup>[5-8]</sup>。IDL 完全面向积矩阵运算。这个特点使其具有快速分析超大规模数据的能力,极大地提高了数据分析和数据可视化的速度。IDL 将内存的栅格数据均视为二维数组,这个特点大大简化了用户的编程。由于 IDL 对数组计算进行了优化,可以将数组作为整体进行操作,其速度远远快于对数组元素的循环操作<sup>[8]</sup>。

本研究利用 IDL 提供的可视化开发平台,实现了高性能地图代数的计算。图 1 为使用 IDL 实现的地图代数的技术流程。

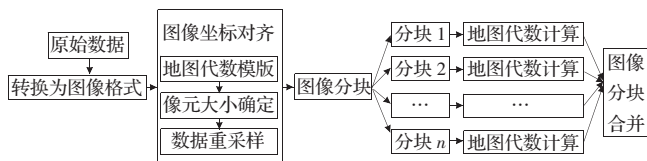


图 1 利用 IDL 实现地图代数的计算流程

### 2.1 数据格式的转换

IDL 具有读写各种格式数据的功能。类似于 C 和 Fortran, IDL 能够读写有格式(ASCII)和无格式(二进制)数据系列。IDL 拥有各种内建的数据类型以及大量数组和结构体,能确保处理任何 ASCII 或二进制数据。IDL 还支持普通图像的读写和科学数据格式的读写。为了进行地图代数的计算,可以将原始的栅格数据转换为任何 IDL 可以直接读取的图像格式。本研究以 Tiff 格式为例作为地图代数计算过程中标准的栅格文件格式。

### 2.2 栅格数据同名像元图像坐标对齐

遥感与 GIS 所使用的栅格数据一般都具有两套空间系统。一套是栅格空间,这个坐标系统一般以图像的左上角为原点(0,0),然后以像元所在的行、列确定其在栅格空间中的坐标。这种表示方法与 IDL 访问数组下标的方式一致。另外,栅格数据还有一套模型空间系统,即像元所对应的地理实体(像元所对应的地面矩形区域)在不同的坐标系统下的坐标<sup>[9]</sup>。

IDL 所有的运算都是直接针对栅格空间所进行的,并没有直接读取模型空间坐标的功能。为了减少系统内部计算所消耗的资源,本研究在设计地图代数计算的时候,以像元与像元的

叠加为操作单元,而不是以地理坐标之间的叠加为操作单元。为此,需要将栅格数据进行预处理,使不同图层相同图像坐标的栅格数据为同名像元(即大小与坐标一致)。

栅格数据集可以以多种多样的文件格式存储,如 GeoTiff、ESRI GRID、Erdas Imagine 等。在数字图像处理中,一般采用二维数组(矩阵)来表示数字图像  $F$ (静止的单色图像),即

$$F = \begin{pmatrix} f(0,0) & \cdots & f(0,N-1) \\ \vdots & & \vdots \\ f(M-1,0) & \cdots & f(M-1,N-1) \end{pmatrix}$$

数字图像中的一个点(即二维数组中一个元素)称为一个像素(Pixel)或像元,像素的坐标( $m, n$ )是二维平面的整数网格点。

而对 GIS 中的栅格数据集来讲,由于每一个栅格单元都与地面的一小片区域相对应,所以它具有一定的地理意义。一般用栅格单元的中心点的坐标来代表这个栅格单元的地理坐标。用来定义栅格图像中像元的地理坐标一般采用仿射变换模型。仿射变换由 6 个参数确定,也被称为 6 参数变换<sup>[9]</sup>。

6 参数模型的 6 个参数可以分为 3 组。一组指定了栅格数据左上角点在模型空间中的坐标。这确定了图像在模型空间中的定位,在该研究中使用( $geoXStart, geoYStart$ )来表示。第二组指定了图像在  $X$  方向与  $Y$  方向的像元所代表的在模型空间中的度量。在该研究中用( $pixSizeX, pixSizeY$ )来表示。还有一组用来表示图像在模型空间中的旋转。由于使用旋转变换的一般比较少,故一般为 0,该研究中使用( $rotA, rotB$ )来表示。

那么,在已知图像坐标的情况下( $imgX, imgY$ )的情况下,就可以求得这点的图像坐标( $geoX, geoY$ )。要注意的是,一般情况下,图像的坐标是从左上算起的,而模型空间中的坐标则是从左下算起。

$$geoX = geoXStart + imgX * pixSizeX + imgY * rotA$$

$$geoY = geoYStart - imgX * rotB + imgY * pixSizeY$$

这就是经过几何纠正的图像坐标与地理坐标的关系式。

对于本文研究的内容来讲,并不包含旋转变换,所以上面的公式可以简化成

$$geoX = geoXStart + imgX * pixSizeX$$

$$geoY = geoYStart - imgY * pixSizeY$$

#### 2.2.1 定义地图代数模版

为了满足 IDL 进行的栅格与栅格叠加运算的要求,必须要首先定义一个地图代数模版。地图代数模版定义了生成的栅格数据集的大小及左上角地理坐标。所有的应用于地图代数计算的栅格数据都要根据地图代数模版进行重采样,从而得到可以直接用于 IDL 计算的栅格数据集。

首先在 IDL 中定义一个仿射变换参数的匿名结构体,它由 4 个参数构成:

$$IMG = \{geoULx:0, geoULy:0, sizeX:0, sizeY:0\}$$

然后,用上面的结构体定义两个变量来对应两幅图像:img1, img2。

再定义一个地图代数模型匿名结构体,它由四个参数构成:

$$MODEL = \{geoULx:0, geoULy:0, geoLRx, geoLRy\}$$

然后,用上面结构体定义一个地图代数模版 model。这里, ( $geoULx, geoULy$ )表示图像左上角像元的地理坐标; ( $geoLRx, geoLRy$ )表示图像右下角像元的地理坐标

根据研究需要的不同,地图代数模版可以取这两幅图像交集的范围,或者是这两幅图像的并集的范围。一般来讲,地图代

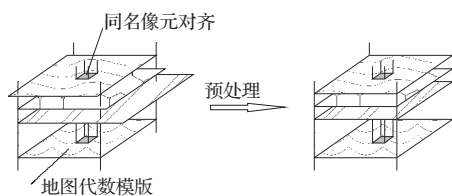


图 2 进行地图代数计算之前的数据预处理

数模版要取图像的交集的范围,不然,会有一些区域落在某些要素无数据的地方,对于计算结果是没有意义的。下面以取交集的范围说明计算两幅图像地图代数模版的算法:

首先要计算这两幅图像的右下角的地理坐标(左上角坐标  $geoULx$  与  $geoULy$  已知,  $n\_elements()$  为 IDL 中计算数组元素个数的函数):

```
img1.geoLRx=img1.geoULx+sizeX*(n_elements(img1[0,*])-1)
img1.geoLRy=img1.geoULy+sizeY*(n_elements(img1[*],0))-1)
那么,这两幅图像的地图代数模版地理坐标范围为:
model.geoULx=img1.geoULx gt img2.geoULx ?
img1.geoULx:img2.geoULx
model.geoULy=img1.geoULy lt img2.geoULy ?
img1.geoULy:img2.geoULy
model.geoLRx=img1.geoLRx lt img2.geoLRx ?
img1.geoLRx:img2.geoLRx
model.geoLRy=img1.geoLRy gt img2.geoLRy ?
img1.geoLRy:img2.geoLRy
```

多个图像地图代数模版的确定依次类推即可。

### 2.2.2 像元大小的确定

像元大小的确定一方面要根据这两幅图像的像元,另一方面还要根据研究内容的不同来确定。对于一幅像元大小确定的图像来讲,可以使用重采样的方法提高这幅图像的空间分辨率。重采样的方法可以使用最邻近距离法、双线性插值法或者是三次卷积法。但是,不管是哪种方法,这种空间分辨率的提高都是完全依赖于原始图像中的信息的。也就是说,使用重采样的方法提高图像的空间分辨率并不会增加图像的信息量。另一方面,对于给定大小的图像来讲,随着像元的空间分辨率的提高,数据量是呈几何数增加的。所以还应该根据研究内容,来确定像元的大小,以避免计算机资源的浪费。

### 2.2.3 数据重采样

根据地图代数模版,每幅栅格图像都得到了由地理坐标定义的地图代数模版的范围,然后,根据得到的由地理坐标定义的范围,再反过来求得每幅图像的由图像坐标定义的范围,即图像在栅格坐标空间中地图代数模版的范围。

当所有的图像在栅格坐标空间中的地图代数模版与像元后确定之后,下面就要根据对应关系,对原始数据进行重采样,从而得到可直接用于 IDL 的数据集。重采样分为空间位置和像素属性值的确定两个步骤。

空间位置的确定一般分为两种:前向映射和后向映射<sup>[10]</sup>。前向映射是把纠正前影像的像素逐一对应到纠正后的影像去,在这一过程中会出现一对多和对应空缺的情况,造成纠正后的影像不连续。目前,前向映射方法用得很少。后向映射是把纠正后影像的像素逐一映射到纠正前的影像中去。纠正后影像的每一个像素都能在纠正前的影像中找到对应像素,避免了前向映射所造成的影像不连续。该方法在影像处理中用得最为广泛<sup>[11]</sup>。本研究使用后向映射来实现,首先确定原始图像与生成图像(由地图代数模版决定)的同名像元。然后,根据原始图像中同名像元的值来确定结果图像的值。

IDL 提供了两个改变图像大小的命令:REBIN 和 CONGRID。函数 CONGRID 及 REBIN 可以用来对一维、二维、三维的数组进行重设大小。CONGRID 函数可以以任意数目来变换一幅图像的数组。而 REBIN 函数要求输出结果的维必须是原始图像维数的整数倍。这两个命令都提供了最邻近距离法与双线性插值法来对数据进行重采样。本研究使用 CONGRID 函数的双线性插值法来对数据进行重采样。

## 2.3 图像分块处理技术

当要处理的数据量太大,以致于超过内存的大小的时候,IDL 会显示不能正确分配内存的错误,尤其当地图代数计算涉及到多个图层计算的时候,对内存的需求量就更大。在这种情况下,就要使用图像分块处理技术。所谓图像分块处理,是指将图像按照一定的规格(一般是矩形),将图像分成几块,然后分块进行运算;再将结果合并到一起的技术。使用图像分块处理技术,可以有效地在计算过程中节省内存的使用,从而有效地进行地图代数和计算。在不涉及邻域计算的情况下,可以依次按照行( $1*N$ )或列( $M*1$ )进行“分块”处理。

在实现图像分块的过程中,本研究使用了一个二维数组作为图像分块的索引,数据的索引由下面的一个数组构成:

```
[ID, xstart, ystart, width, height]
```

然后,使用 IDL 的 ReadTiff 命令,就可以按照分块读取原始栅格文件中的数据并进行地图代数计算了。下面是用 IDL 编写的获取图像分块索引的函数:

```
; width,height 为图像的大小,subsize 为图像分块的大小
```

```
function GETIDLINDEX ,width ,height ,subsize
xnum=ceil(float(width)/subsize)
ynum=ceil(float(height)/subsize)
outarr=indgen([4,xnum*ynum])
index=0
for j=0,ynum-1 do begin
for i=0,xnum-1 do begin
xstart=subsize*i
ystart=subsize*j
xend=subsize*(i+1)-1
yend=subsize*(j+1)-1
subsize_x=xend gt width-1 ? width-xstart:subsize
subsize_y=yend gt height-1 ? height-ystart:subsize
rect=[xstart,ystart,subsize_x,subsize_y]
outarr[0:3,index]=rect]
index=index+1
endfor
endfor
return, outarr
end
```

## 2.4 地图代数函数的实现

在对数据预处理完成之后,这一点相对就比较简单了,所有的操作都可以看作是针对矩阵的操作。ArcGIS 所提供的地图代数计算可以实现将众多的函数集成的功能,在 IDL 中,可以以更加灵活的方式实现。在这里,根据计算时候所用到的像元及图层的数目,分为基于像元的计算与基于邻域的计算。下面分别举例说明。

基于像元的计算比较简单,以三个图层相乘为例说明。假设  $A$ 、 $B$ 、 $C$  分别为读入内存的三个图像的变量名称,那么,为了得到 3 个图层相乘的结果,在 IDL 中,可以运行:

```
D=a#b#c
```

(“#”是 IDL 中矩阵乘的符号)

那么, $D$  将与  $A$ 、 $B$ 、 $C$  具有相同的维数及行列数。

基于邻域的计算一般针对单个图层进行。下面以求取 DEM 数据的起伏度为例进行说明。起伏度有多种算法,一般局部区域最大值与最小值的差作为起伏度的度量。实际计算中,需要根据地理距离(如  $1 \text{ km}^2$  大小)来确定窗口的大小;为了说明的方便,将局部区域限定为  $3 \times 3$  的窗口。在这个计算过程中,首先需要对数据求得局部最大值与局部最小值,计算过程如下。

对于某一个像元来讲,假设它的行列数(坐标)为 $(i,j)$ ,那么,取出原始数据中的一个窗口:

$$A = \begin{pmatrix} a_{i-1,j-1} & a_{i,j-1} & a_{i+1,j-1} \\ a_{i-1,j} & a_{i,j} & a_{i+1,j} \\ a_{i-1,j+1} & a_{i,j+1} & a_{i+1,j+1} \end{pmatrix}$$

然后计算  $u = \max(A)$ ,  $v = \min(A)$ 。

这里  $\max()$  与  $\min()$  分别为 IDL 内置的求取数组中最大值与最小值的函数,这样就分别得到窗口的最大值与最小值。像元 $(i,j)$ 的起伏度即为  $u-v$ 。

### 3 项目应用实例

以使用 USLE 方程计算三江平原土壤侵蚀模数的计算为例,对使用 ArcGIS 进行地图代数计算与使用 IDL 进行地图代数计算所消耗时间进行比较。由于遥感/GIS 数据一般比较大,所以在计算过程中使用了分块处理的技术。

测试使用惠普 dc7100 机器进行测试。硬件配置为:CPU:64 位 P4 3.2G,内存:512M,硬盘:SATA1 7200 转 160G;操作系统:MS Windows XP Pro sp2;应用软件:ArcGIS 9.0 sp2;IDL 6.3。

使用通用流失方程(USLE)来进行三江平原水土流失量的计算共需要 6 个因子:

$$A = R * L * S * C * P$$

式中, $A$  为年平均土壤侵蚀量 ( $t/hm^2a$ ); $R$  为降雨侵蚀因子( $J/m^2$ ); $K$  为土壤可蚀性因子( $t/ha$ ); $L$  为坡长因子, $S$  为坡度因子,均无量纲; $C$  为作物经营管理因子,无量纲; $P$  为水土保持措施因子,无量纲。

三江平原位于黑龙江省东北隅,占地面积约为 108 800  $km^2$ 。通过对样本区数据进行处理,按照 30 m 的地面分辨率进行重采样,最后得到单幅栅格数据的大小(像元数目)为 12 555\*200 58,所占磁盘空间大小约为 960 MB。然后,分别使用 IDL 与 ArcGIS 对进行地图代数计算,经过对数据进行多时期计算求平均值。其中,IDL 使用 GeoTiff 文件格式进行计算,ArcGIS 使用 ESRI Grid 文件格式进行计算。计算所使用的时间为:

ArcGIS 平均计算每 6 个因子的乘积要用 6 个小时 18 分钟,而使用 IDL 计算却只需要 28 分钟。

### 4 结论与讨论

该研究针对 ArcGIS 软件在进行地图代数计算时速度缓慢

(上接 158 页)

### 4 Blog 信息抽取的应用

目前对于 Blog 的研究越来越广泛,如对于 Blog 世界中社区的发现、对于 Blog 中热点话题的分析、对于评论信息中争论内容的分析与识别、Blog 话题内容的相关度等等。那么在进行这些应用前首先要做的工作就是去准确识别并抽取这些主题信息。所以将本文的方法应用到上述提到的这些应用中,则可以准确且较快地抽取需要的内容,这样可以大大地提高对 Blog 后期应用的效率以及准确率。

### 5 结束语

针对目前 Blog 使用的快速增长以及对于 Blog 的大量应用,本文提出了一种基于模板化的 Blog 信息抽取方法。该方法可以较快速并准确地抽取 Blog 网页中的主题信息。同时由于事先提取了模板,用户不必对每一个 Blog 网页都进行分析,这为后续对 Blog 的应用带来了很大的便利。但是,本文的方法尚有不足之处需要改进。第一,对于不在模板库中的 Blog 网页,本文的方法只能实时地将该页面收集到后台进行分析,然

的缺点,使用第 4 代可视化交互数据语言 IDL 重新实现了 ArcGIS 地图代数的功能,并且针对 IDL 数据处理的一些问题,如数据格式、进行地图代数之前的数据预处理等进行了深入探讨。通过实践验证,该研究能够大大缩短地图代数的计算时间,提高计算性能,对于栅格数据的空间分析具有重要的现实意义。在使用 USLE 方程计算三江平原土壤侵蚀模数的实践应用中,地图计算计算时间缩短为原来的 7.4%。

目前所存在的一个问题是,由于在使用 IDL 进行地图代数计算之前需要对数据进行预处理,所以会造成一定的数据冗余。另外,由于 IDL 并没有像 ArcGIS 一样内置许多地图代数函数,所以在应用过程中需要用户自行编写代码。随着 IDL 在国内外应用的普及,相信各种标准函数库会逐渐增多起来,不断满足用户的要求。对于某些要求实现特定功能的用户,则需要对 IDL 语言的语法及各种函数都比较熟悉。

### 参考文献:

- [1] 邬伦,刘瑜,张晶,等.地理信息系统——原理、方法和应用[M].北京:科学出版社,2001.
- [2] 陈天恩,冯启民,陈红,等.基于遥感影像的城市震害模拟[J].自然灾害学报,2006(2):121-126.
- [3] 耿协鹏,胡鹏,杨传勇.多重多边形叠置栅格算法[J].测绘工程,2006(1):34-37.
- [4] 闫建军,樊红,张成刚.基于 AO 的栅格分析在土地淹没统计中的应用[J].地理空间信息,2006(1):34-36.
- [5] 翟建军,王汉东,吴想凤,等.基于 IDL 的三峡永久船闸三维可视化的实现[J].地理空间信息,2006,4(5):10-12.
- [6] 何全军.基于 IDL 的三维地形可视化系统开发[J].测绘信息与工程,2006(1):19-20.
- [7] 韩培友.IDL 可视化分析与应用[M].西安:西北工业大学出版社,2006.
- [8] 闫殿武.IDL 可视化工具入门与提高[M].北京:机械工业出版社,2003.
- [9] 陈清明,徐建刚.GIS 数据采集中的坐标变换模型应用[J].测绘通报,1999(9):16-18.
- [10] Gonzalez R C, Woods R E.数字图像处理[M].2 版.北京:电子工业出版社,2002.
- [11] 蒋耿明,刘荣高,牛铮,等.MODIS 1B 影像几何纠正方法研究及软件实现[J].遥感学报,2004,8(2):158-164.

后手动提取该网页的模板并加入到模板库中,而不能实现机器的自动学习功能。第二,对于同一个 Blog 网站中有多个 Blog 模板的情况并未考虑在内。这些都是未来努力的方向。

### 参考文献:

- [1] 欧健文,董守斌,蔡斌.模板化网页主题信息的提取方法[J].清华大学学报,2005,45(S1):1743-1747.
- [2] 孙承杰,关毅.基于统计的网页正文信息抽取方法的研究[J].中文信息学报,2004,18(5):17-22.
- [3] Mishne G, Glance N. Leave a reply: an analysis of weblog comments[C]//Third Annual Workshop on the Weblogging Ecosystem, Edinburgh, Scotland, May 2006.
- [4] Oka M, Abe H, Kato K. Extracting topics from weblogs through frequency segments[C]//Proceedings of the WWW06 Workshop on Web Intelligence, 2006.
- [5] 杨宇航.基于内容与链接分析的重要 Blog 信息源发现[D].哈尔滨工业大学计算机科学与技术学院,2006.
- [6] 国内八大著名中文博客系统横向评测[EB/OL].(2006-02-21).http://tech.sina.com.cn/s/2006-02-21/1503847398.shtml.
- [7] HTML parser[EB/OL].(2006-09-17).http://htmlparser.sourceforge.net/.