

基于 REST 的空间信息服务互操作协议的研究

李波¹ 丁仙峰¹ 伊文英¹ 张敏芳²

(西安交通大学 西安 710049)¹

(中国航空工业集团公司洛阳电光设备研究所 洛阳 471023)²

摘要 REST 式的 Web 服务充分地利用 HTTP 协议的分布式特性,可以使 Web 服务的开发变得更加高效、简洁、可维护。着力于 REST 式空间信息服务互操作协议研究,通过分离 OGC Web 服务定义中的动作、意图和表示的方式,确定服务中的资源。最后将 REST 的“资源”概念与空间信息“数据”共享结合起来,利用 HTTP 协议定义的操作,设计了 REST 式空间信息服务互操作规范 REST-WMS 与 REST-WFS。

关键词 REST,空间信息服务,WMS,WFS

Research on Geospatial Information Service Interoperability Protocol Based on REST

LI Bo¹ DING Xian-feng¹ YI Wen-ying¹ ZHANG Min-fang²

(Xi'an Jiaotong University, Xi'an 710049, China)¹

(Electronics Optics Institute of Luoyang, Aviation Industry Corporation of China, Luoyang 471023, China)²

Abstract Restful Web services is full use the distributed characteristics of HTTP protocols so that it can make the development of Web services become more efficient, simple, and maintainable. The main research focused on geospatial information service interoperability protocol. The resource of the service defined by separating the operation, intention and representation defined in the OGC Web service. Then using the operations defined by HTTP by combining the “resource” concept and geospatial information “data” sharing, designed the RESTful geospatial information Service as REST-WMS and REST-WFS.

Keywords REST, Spatial information service, WMS, WFS

1 引言

随着 Web 服务应用的普及,形成了两种主流架构:RPC 式架构和 REST 式架构。RPC 式架构主要采用 SOAP 协议栈实现,其复杂性不符合 Web 的简单性^[1]。REST 模式以其高性能、高可伸缩性、简单性、一致性、可靠性变得越来越流行。越来越多的 Web 服务开始采用 REST 风格来进行设计和实现^[2]。目前,大多数 Web 服务供应商,如 Yahoo、Ebay、Amazon、Google 都提供了 REST 风格的 Web 服务 API。据著名的 Web2.0 网站 programmableweb.com 统计,目前可用的 REST 风格的 Web 服务 API 数量为 1241 个,而基于 SOAP 的 Web 服务 API 数量只有 335 个,而且 REST API 出现的时间较短,足可见其发展的迅猛^[3]。

自 Roy Thomas Fielding(罗伊·托马斯·菲尔丁)博士在其毕业论文《Architectural Styles and the Design of Network-based Software Architectures》中提出的 REST 概念^[4]起,如何将 REST 方法与 GIS 领域结合成为国内外研究的热点。

在国内,姚鹤岭博士在其论文中首次将 REST 与 GIS 联系在一起^[5],设计出面向服务架构的 GIS Web 服务。詹睿博士在其论文中提出 GIS Web 服务的概念,对比了 REST 风格的 Web 服务与基于 SOAP 协议栈的 Web 服务,其提出的 REST 风格的 Web 服务更贴近于网络本身的基础特性,更适应于 GIS 在 Web 上提供地图服务^[6]。国外开始推出一些 REST 式 GIS 方面的项目,较为关注的有开源项目 GeoREST。ESRI 公司也推出了 ArcGIS Server REST API。Geoserver 在 2010 年 1 月发布的最新版本中也添加了对 REST 的支持。Autodesk 的技术总监 Geoff Zeiss 在 2009 年 FOSS4G 会议中发表题为《An Open Architecture for RESTful Geospatial Web Services》的演讲,提出以 REST 方式提供对开放式地理信息访问是种很好的方式。Institute of Methodologies for Environmental Analysis 的 P. Mazzetti 等人已经在地球与空间科学领域应用了 RESTful 实现^[7]。

为促进地理信息服务共享,国际开放地理信息系统协会(OGC)建立了 OpenGIS Web Service 研究计划,提出了

本文受西安市科技计划江河流域水污染预警预报软件系统研发项目(CXY1010),陕西省自然科学基金研究计划重点项目时空推理中的空间变化演算研究及应用(2010JM8036)资助。

李波(1968—),男,博士生,副教授,主要研究方向为分布式计算及软件工程、模型数值模拟及可视化、地理信息系统、决策支持系统, E-mail: boblee@xjtu.edu.cn; 丁仙峰(1987—),男,硕士生,主要研究方向为地理信息系统及网络地理信息处理互操作; 伊文英(1985—),女,硕士生,主要研究方向为地理信息系统及网络地理信息处理互操作; 张敏芳(1985—),女,硕士生,主要研究方向为瞄准显示图像处理。

WMS、WFS、WCS等一系列服务规范,用来进行 Web 之上的空间数据发现、存取、集成、分析、利用和可视化、位置信息及地学处理等^[8,9]。目前基于 OGC 规范的研究和应用大多采用基于 SOAP 协议栈的平台实现^[10]。然而 SOAP 协议栈并不是特意为 GIS 设计的。使用 SOAP 协议栈实现的地理信息服务,在数据之外包裹了多层协议,并且需要获取很多关于服务描述的元数据才能获取服务,对服务请求者较为复杂。此外,SOAP 协议栈设计之初并非用于地理领域,而 GIS 数据具有空间参考、海量存储等特点使得在使用过程中必须修改 SOAP 协议栈才能使其适应具体领域并得以应用^[6]。因此,利用 REST 的核心思想:资源,来构建 REST 式 WMS 和 REST 式 WFS。

2 REST 式地理信息服务互操作协议设计思想

REST 将接口限定在一组广为人知的标准动作中(比如 HTTP 的 GET、PUT、DELETE 等)以供调用。此类 Web 服务重点放在与那些稳定的资源进行互动,而不是消息或动作^[11]。而 OGC 定义的 WMS 和 WFS 规范访问方式是基于 HTTP 协议的 GET 和 POST 操作,除 GetMap 操作只支持 GET 方法外,其他操作均支持 GET 和 POST 两种方法,其中支持 GET 方法是强制的,支持 POST 方法是可选的,如表 1 所列。

表 1 WMS 和 WFS 操作分析^[12,13]

| option | HTTP method | Request parameter |
|--|-----------------|---|
| GetCapabilities | GET、POST | service,request |
| WMS http:// Host [: port]/ path[? {name [=value]&}] | GetMap | GET version, request, layers, styles, CRS, bbox, width, height, format |
| GetFeatureInfo | GET、POST | version, request, mapre- quest part, query_layers, info_format |
| WFS http:// Host [: port]/ path[? {name [=value]&}] | GetCapabilities | GET、POST service,request |
| DescribeFeature- Type | GET、POST | service,request,typenamees |
| GetFeature | GET、POST | version, service, request, typenamees |

根据对 WMS 和 WFS 提供操作的分析,抽取以下内容:

- 1) WMS 的 GetMap 操作和 WFS 的 GetFeature 操作是核心内容,目的是向客户端提供模型数据和视图数据;
- 2) WMS 和 WFS 的 GetCapabilities 操作和 WFS 的 DescribeFeatureType 操作的目的是向用户提供资源信息和对最终资源的访问方式;
- 3) 这些操作的核心参数是 service、request、layers(WMS) 和 typenamees(WFS)。

因此,本文提出的 REST 式 WMS 和 REST 式 WFS 从广度上应该可以覆盖 WMS 和 WFS。根据上述分析,本文从 WMS 和 WFS 服务中抽取以下资源:

- (1) 图层列表。
- (2) 图层,根据图层名称提供地图服务。
- (3) 不同响应形式的图层,根据图层名称和响应形式(JPEG、GIF、PNG 等)提供地图服务,默认请求形式为 PNG。

(4) 不同范围的图层,根据图层名称和 bbox 提供地图服务,默认 bbox 为地图边界最大值。

(5) 不同范围、不同响应形式图层,根据图层名称、响应形式和 bbox 提供地图服务。

(6) 要素列表(对应图层列表)。

(7) 要素(对应图层),根据类型名称(对应图层名称)提供要素服务(默认请求形式为 GML)。

(8) 要素上的某一点,根据类型名称和非空间属性(由 filter 指定)提供特定要素服务。

前 5 点属于 WMS 的范围,第(6)和(7)点属于 Basic-WFS 的范围,第(8)点属于 Transaction-WFS 的范围。

按照 REST 的设计原则“网络上的所有事物都当作资源”,客户端不论是对模型数据还是对视图数据的请求都应该视为对资源的请求,而不是对操作的请求^[14]。根据对 OGC 服务规范的分析 and REST 设计原则,将所有要素分为 3 类:意图(Intent)、动作(Action)和表示(Representation),如图 1 所示。

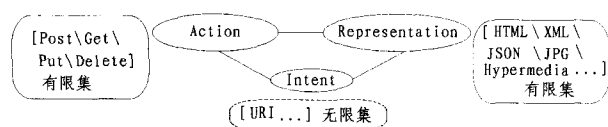


图 1 三角架构

意图表述了用户的请求目的,采用名词来表达,它是唯一的资源标识,也是资源的地址,包括静态的 HTML 页面、地图 JPG 图像、KML 文件、动态请求 Web 应用生成的查找结果 KML 文件等。动作是个抽象概念,表达了用户对意图即资源予以施加的操作,采用动词来表达,可以是 POST、GET、PUT、DELETE 等 HTTP 方法,是施加于名词上的有限操作。表示指响应的表达方式,是对资源形态的抽象。因此本文提出的 REST 式地理信息服务规范(REST-WMS 和 REST-WFS)将设计的重点放在资源而不是操作上。

在设计前,本文做如下定义:

1)“惯例重于配置”原则

客户端只要遵循一定的 URI 设计惯例和地图数据分层、命名习惯,即可在不查阅元数据描述信息的基础上设计出合适的 URI,向服务器发出请求。

2) 分层系统规范

service = { maps, features }, groups = { layers, typenamees }, group = { layername, typename }, format = { xml, gml, png }。service、groups 与 group 构成 3 层系统,代表资源的路径与分类。format 代表资源的不同表现形式。

3) 操作即动词的定义

利用 HTTP 方法 GET、POST、PUT 和 DELETE。

4) 对于选择资源的表示形式提供两种方式

一是放在 HTTP 请求头里;二是利用 URI 后缀显式表达。

5) 表示是超媒体的

不仅包含数据,还包含指向其他资源的链接。使得客户端应用状态可以在服务器提供的“超媒体”指引下产生变迁,也可用此方式实现服务的发现与定位。

6) URL 组成定义,见表 2。

表 2 URL 组成定义

| 定义 | 说明 |
|--|-----------------------------|
| http://host[:port]/path1/[path2] [.format][? {name [=value]}&.] | []:0 或 1, {}:0 或多个 |
| name [=value]&. | 采用 name/value 的方式 表示请求参数 |

本文面向客户端提出服务接口,希望根据 HTTP 请求协议的第一行即可对资源进行明确定位以及确定对资源的操作行为,即通过 URI 定位各种资源,通过 HTTP 标准动作确定对资源怎样运行。在上述基础上定义表 3 所列的接口。

表 3 接口定义

| 序号 | HTTP 操作 | URI |
|----|---------|--|
| 1 | GET | http://host[:port]/service/capability[.format] |
| 2 | GET | http://host[:port]/service/groups[.format] [? {name [=value]}&.] |
| 3 | GET | http://host[:port]/service/groups/group[.format] [? {name [=value]}] |
| 4 | GET | http://host[:port]/service/groups/group/id[.format] [? {name [=value]}] |
| 5 | POST | http://host[:port]/service/groups/ |
| 6 | POST | http://host[:port]/service/groups/group/ |
| 7 | PUT | http://host[:port]/service/groups/id |
| 8 | PUT | http://host[:port]/service/groups/group/id |
| 9 | DELETE | http://host[:port]/service/groups/group/id |

表 4 REST-WMS 操作说明

| 操作 | 配置说明 | HTTP method | Request URI |
|-----------------------|--|-------------|---|
| GetCapabilities (WMS) | service=maps format=xml/html | GET | /maps/capability.xml /maps/capability.html (REST-WMS) |
| GetMap (WMS) | service=maps groups=layers group=layername format=png | GET | /maps/layers/layername (REST-WMS) |
| GetFeatureInfo (WMS) | service=maps groups=layers group=layername format=gml | GET | /maps/layers/layername/ id.gml (REST-WMS) |

表 5 REST-WFS 操作说明

| 操作 | 配置说明 | HTTP method | Request URI |
|----------------------------|---|-------------|---|
| GetCapabilities (WFS) | service=features groups=typenamees format=gml | GET | /service/typenamees.xml (REST-WFS) |
| DescribeFeature Type (WFS) | service=features groups=typenamees group=typename format=xml | GET | /features/typenamees/ typename.xml (REST-WFS) |
| GetFeature (WFS) | service=features groups=typenamees group=typename format=gml | GET | /features/typenamees/ typename.gml (REST-WFS) |
| INSERT (WFS) | service=features groups=typenamees group=typename | POST | /features/typenamees/ typename (REST-WFS) |
| UPDATE (WFS) | service=features groups=typenamees group=typename | PUT | /features/typenamees/ typename/id (REST-WFS) |
| DELETE (WFS) | service=features groups=typenamees group=typename | DELETE | /features/typenamees/ typename/id (REST-WFS) |

以这种方式设计的 REST-WMS 和 REST-WFS 规范将

资源与其表现形式暴露给用户,并将动词、名词以及表示分离。其中 id 代表参数,可以是其他表示形式。URI 是将 Web 上的信息相互联系在一起纽带,保证了信息的全局性和唯一可达性。因此,在设计时要保证 URI 在语义上可以对资源做清晰的描述。

具体针对 WMS 和 WFS 的操作做表 4、表 5 所列说明。

说明:HTTP 的具体响应机制决定是对静态图片和静态资源响应,还是对 Web 应用的响应;服务器端哪个处理程序负责处理哪个请求,可根据需要设置 Web 服务器或者应用服务器完成。

3 REST 式地理信息服务互操作协议实现

上述接口的定义,可以从广度上覆盖 OGC 提出的 WMS 和 WFS 规范。由于 REST 自身的可扩展性,因此接口可以根据业务需求进行具体扩展。

1) 基本请求的实现方式

基本请求是指可以对应到 WMS 与 Basic-WFS 的请求。/capability[.format] 会列举出 REST-WMS 和 REST-WFS 提供的所有的服务。format=html 是面向浏览器的 human Web 应用的,format=xml 是面向应用程序的可编程 Web 应用的。示例如下:

发送 HTTP 请求 GET /maps/capability.html,会返回如下 HTML 片段:

```
<link rel="chapter" type="application/html" title="黄河专题"
href="http://host:port/maps/groups/yellowrivertopic">
<link rel="chapter" type="application/html" title="居民地"
href="http://host:port/maps/groups/settlement">
...
<link rel="chapter" type="application/html" title="交通要素"
href="http://host:port/maps/traffic">
```

发送 HTTP 请求 GET /maps/capability.xml,响应格式为:

```
<maps>
<map>
<name>黄河专题</name>
<atom:link xmlns:atom="http://www.w3.org/2005/Atom" rel="
"chapter" href="http://host[:port]/maps/groups/yellowrivrtopic.
xml" type="application/xml"/>
</map>
<map>
<name>居民地</name>
<atom:link xmlns:atom="http://www.w3.org/2005/Atom" rel="
"chapter" href="http://host[:port]/maps/groups/settlement.
xml" type="application/xml"/>
</map>
.....
</maps>
```

对于具体图层发送 GET /maps/groups/settlement/bridge.html,会返回如下 HTML 片段:

```
<link rel="alternate" type="application/json" title="桥梁的 GeoJ-
SON 表现形式" href="http://host:port/maps/groups/settlement/
bridge.geojson">
<link rel="alternate" type="application/xml" title="桥梁的 KML
```

表现形式" href = " http://host: port/maps/groups/settlement/bridge.kml">

(link rel = " alternate" type = " application/xml" title = " 桥梁的 GeoRSS 表现形式" href = " http://host: port/maps/groups/settlement/bridge.georss">

2) 查询的实现方式

利用操作 4 可获得图层或者要素上的某一点,操作 4 可进行扩展成如下两种实现方式:

方式一 在 URI 中以路径的形式表示参数,发送 HTTP GET 请求,如 GET/maps/groups/settlement/bridge/yellowriverbridge,表示含义为返回 bridge 图层中,名为 yellowriverbridge 的桥梁信息。

方式二 将请求名称等作为参数,以“名称/值”对的形式附加在 URI 路径后面。为了与第一种方式以及对静态资源的 URI 区别,修改 URI 路径为:/maps/groups/settlement/bridge/.format。构建 HTTP 请求协议为:GET/maps/groups/settlement/bridge/.xml? name = yellowriverbridge。该方式可通过 form 表单实现。

3) 组合的实现方式

从上节针对 WMS 和 WFS 抽取的资源可以看出,图层是资源的最小单位,并且互相独立。OGC 忽略了图层之间的组合状态。对于组合问题提供了两种解决方式:

方式一 服务提供者事先按照某种组合关系将具有共同性质的多个图层合并成组发布,如本文给出的接口定义中对 groups 的请求。这种方式的优点是用户可忽略图层组与图层的区别,将图层组简单看成一个图层处理,不必考虑复杂的请求方式;其缺点是扩展性差,图层组由服务提供者固定,不利于客户端个性化的请求。

方式二 由客户端服务使用者根据自己的需求进行多个图层的组合。对于这种需求,目前基于 OGC WMS 的客户端应用的实现方法是:采用脚本语言构建 HTTP 请求来访问所需的多个图层,在客户端通过设置图片的透明度完成多个图片的叠加。这种方式的弊端是需要发送多个 HTTP 请求。本文采用矩阵 URIs(在 URI 中用分号表达图层的并列组合关系)的方式实现一次请求多个图层的目标。修改 URI 路径为:/maps/groups/group/layer1; layer2; layer3; 或者直接忽略服务提供者提供的图层组 group 的概念:/maps/layer1; layer2; layers 组合与查询提供的 URI 路径设计方式也可进行高层次的合并,以提供在“组合”基础上的“查询”。如 URI 路径定义为:/maps/groups/group/layer1; layer2; layer3? id=3。

本文基于黄河资源委员会(黄委会)提供的黄河地图,实现了 REST-WMS, REST-WFS,并通过 Openlayers 客户端显示服务提供的地图,验证了本文研究正确性,如图 2 所示。

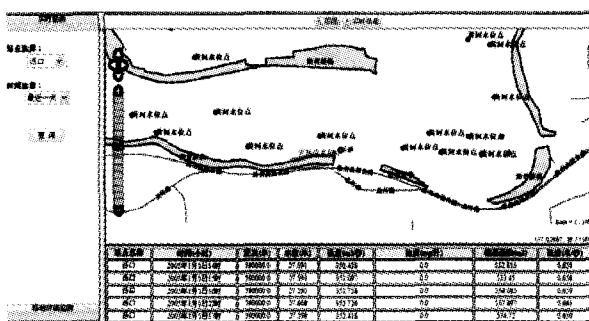


图 2 自定义 REST 式图层

4 分析与评价

REST 式地理信息服务规范不仅提供了一个崭新的架构和 URI 设计参考,还可通过 URI 来设计系统的结构。每个 URI 都代表一个资源,而整个系统就是由这些资源组成的^[15]。设计良好的 URI 之间会组成一定的关系,可以由这些关系来决定系统的结构和各部分应用之间的调用关系。因此,参照 REST 方式设计的 REST-WMS 规范和 REST-WFS 规范通过定义好的 URI 即可明确系统的结构。

功能完整性:WMS 和 WFS 结合就可覆盖以矢量地理信息数据为主的地理信息系统的基本功能。REST-WMS 和 REST-WFS 的设计方案参照 OGC 标准规范 WMS 和 WFS,功能上与之——对应。因此采用 REST-WMS 和 REST-WFS 规范设计的地理信息服务在信息共享方面具备完整性。

功能扩充性:REST-WMS 与 REST-WFS 从“组合”的角度扩充了 WMS 与 WFS。使得客户端在不基于脚本语言的基础上即可轻松地构建“组合”URI 请求多个图层。

可扩展性:向 http://host[: port]/service/capability[. format]发送 GET 请求,返回的是 XML 文档,若客户端为可编程的应用接口或带有脚本的动态 HTML 文件,即可解析 XML 文档,构造新的 HTTP 请求。同时,WFS 只提供 GML 格式的响应,而本文设计的 REST-WFS 可提供除 GML 之外的多种响应格式(如 KML、HTML 等)。

状态迁移:REST 式 OGC 规范符合 REST 的最基本目的,决定怎样使一个定义良好的 Web 程序向前推进,因此在请求的返回文件中会设置下一状态的超链接,以方便 human Web 或者 programmable Web 状态向前推进。

解耦性:资源只和 URI 相关,与具体实现无关,因此 REST 具有更好的解耦性。本文提出的设计方式正是基于客户端与服务器端实现分离的原则。服务器端的具体实现对客户端是隐藏的。如对于“GET /group”这样一个请求,在 Rails 的实现中,可以直接映射到名为 Group 的 Model 上,也可以在 Action 里面由业务逻辑组合成为抽象资源。

可伸缩性:采用 REST 式设计强制所有操作都是无操作性的,没有上下文的约束;同时对客户端和服务器端均衡地任务分配,则服务器端只负责提供资源以及操作资源的服务,客户端根据资源中的数据 and 表现形式自己做映射,这样可减少服务器的开销。

结束语 REST 式地理信息服务规范将资源作为一种抽象的概念,并将其映射到相应的一套 URI 规则上。通过对资源的操作来实现分布式,而不是通过对象的方法调用来实现分布式。它不仅提供了一个崭新的架构和 URI 设计参考,还可通过 URI 来设计系统的结构。每个 URI 都代表一个资源,而整个系统就是由这些资源组成的。设计良好的 URI 之间会组成一定的关系,可以由这些关系来决定系统的结构和各部分应用之间的调用关系。因此,参照 REST 方式设计的 REST-WMS 规范和 REST-WFS 规范通过定义好的 URI 即可明确系统的结构^[16]。

参考文献

- [1] Richardson L, Ruby S. RESTful Web services [M]. O'Reilly Media, Inc., 2007

点实际位置, R 是节点的通信半径。本文在节点分布相同的情况下, 对 CLA、WCLA 和 WCLA-PSO 和 WCLA-QPSO 算法进行仿真, 各算法重复运行 10 次后求均值, PSO 和 QPSO 的迭代次数都为 100, 其仿真结果如图 1、图 2 所示。

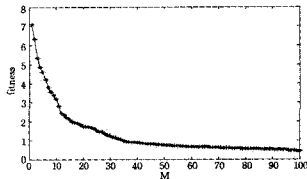
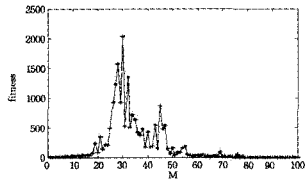


图 1 WCLA-PSO 定位算法收敛图 图 2 WCLA-QPSO 定位算法收敛图

图 1、图 2 中横坐标 M 是迭代次数, 纵坐标是每次迭代的适应值。根据图 1 分析其收敛性可知, WCLA-PSO 算法容易陷入局部极值, 其适应值从第 18 次到第 57 次迭代之间突变较大, 从第 68 次到 79 次迭代仍然有较小抖动, 81 次以后趋于稳定, 其适应值为 3.4895, 第 100 次时适应值为 3.4333。

图 2 所示的 WCLA-QPSO 算法的适应值随迭代次数递减, 收敛速度快, 能有效地克服 WCLA-PSO 算法极易陷入局部极小值的缺点。第 1 次迭代适应值为 7.0831, 从第 35 次迭代开始适应值小于 1, 最后一次迭代适应值为 0.4278。

图 3 中横坐标是各盲节点, 纵坐标是各盲节点的平均定位误差。从图 3 可以看出, 由 QPSO 优化的节点定位精度与由 PSO 优化的相比, 其改善明显。进一步从误差均值、误差方差、计算量方面进行比较分析, 如表 1 所列。

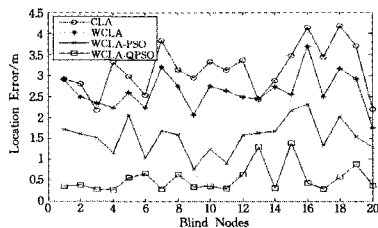


图 3 CLA、WCLA、WCLA-PSO 和 WCLA-QPSO 算法的平均定位误差比较

从表 1 可以看出, WCLA-QPSO 算法的均方误差均值最

小, 方差最小, 计算量比 PSO 减少接近一半, 更容易在相对较短的时间内获得较高精度, 且定位精度的稳定度更高。

表 1 定位算法性能指标比较

| 定位算法 | 平均定位误差均值 | 平均定位误差方差 | 计算量(s) |
|-----------|----------|----------|--------|
| CLA | 3.1413 | 0.3251 | 0.3750 |
| WCLA | 2.6177 | 0.1862 | 0.3947 |
| WCLA-PSO | 1.5355 | 0.1675 | 1.5197 |
| WCLA-QPSO | 0.5265 | 0.1058 | 0.8291 |

结束语 在加权质心定位算法的基础上, 提出了结合 RSSI 测距技术的量子粒子群优化算法, 并对其收缩扩展系数进行了改进。从结果可以分析出: 量子粒子群优化改进加权质心定位算法收敛速度快, 定位精度高, 计算量小。因此, 利用量子粒子群优化改进加权质心定位算法对无线传感器网络节点定位, 能在不增加额外硬件和通信量的基础上有效地获取目标位置。同时, 该方法对移动通信等其它定位问题也具有一定的应用价值。

参考文献

- [1] Kannan A, Guoqiang M, Vucetic B. Simulated annealing based wireless sensor network localization with flip ambiguity mitigation[A]// IEEE 63rd Vehicular Technology Conference[C]. VTC 2006-Spring-Proceedings, 2006:1022-1026
- [2] 章磊, 段莉莉, 钱紫鹃, 等. 基于遗传算法的 WSN 节点定位技术[J]. 计算机工程, 2010, 10(36): 85-87
- [3] 王取凤, 王岩. 基于矢量的无线传感器网络节点定位综合算法[J]. 通信学报, 2008, 29(11): 227-231
- [4] 陈星舟, 廖宏宏, 林建华. 基于粒子群优化的无线传感器网络节点定位改进[J]. 计算机应用, 2010, 30(7): 1736-1738
- [5] 姚金杰, 韩炎. 基于改进自适应粒子群算法的目标定位方法[J]. 计算机科学, 2010, 37(10): 190-192
- [6] 史洪宇, 燕莎, 曹建忠. 无线传感器网络节点定位的混沌粒子群优化算法[J]. 探测与控制学报, 2010, 32(10): 46-49
- [7] Sun J, Feng B, Xu W B. Particle swarm optimization with particles having quantum behavior[C]// Proceedings of 2004 Congress on Evolution Computation. Piscataway, NJ: IEEE Press, 2004: 25-331
- [8] 张俊. 构建 REST 风格的 Web 应用程序[D]. 北京: 北京邮电大学, 2009
- [9] 李峰, 李春旺. Mashup 关键技术研究[J]. 现代图书情报技术, 2009(1): 44-49
- [10] Fielding R. Architectural styles and the design of network-based software architectures[D]. Citeseer, 2000
- [11] 姚鹤岭. 基于 REST 的 GIS Web 服务研究[D]. 郑州: 中国人民解放军信息工程大学, 2005
- [12] 詹睿. 基于 Ajax/REST 的 GIS WEB 服务研究与实现[D]. 北京: 中国地质大学, 2008
- [13] Mazzetti P, Nativab S, Caronc J. RESTful implementation of geospatial services for Earth and Space Science applications[J]. International Journal of Digital Earth, Supplement 1, 2009, 2
- [14] 李新通, 何建邦. GIS 互操作与 OGC 规范[J]. 地理信息世界, 2003(5)
- [15] 杨靖宇, 谢超, 柯希林, 等. 地理信息服务的思考与探索[J]. 测绘工程, 2009(1)
- [16] 高升, 陈能成, 龚健雅, 等. 基于多协议的地理信息服务集成[J]. 测绘信息与工程, 2006(06)
- [17] Pautasso C, Zimmermann O, Leymann F. Restful Web services vs. bigWeb services; making the right architectural decision[C]// ACM. 2008: 805-814
- [18] OGC. Open GIS Web Map Service implementation specification [S/OL]. http://portal.opengeospatial.org/files/?artifact_id=5316, 2004
- [19] OGC. Open GIS Web Feature Service implementation specification 2.1 [S/OL]. [https://portal.opengeospatial.org/files/\(accessed05-05-2007\)](https://portal.opengeospatial.org/files/(accessed05-05-2007)), 2005
- [20] 王建涛. 基于 Web 的地理信息服务的研究与实践[D]. 郑州: 中国人民解放军信息工程大学, 2005
- [21] Al-Zoubi K, Wainer G. Using REST Web-Services Architecture for Distributed Simulation[J]. Pads 2009; 23rd Workshop on Principles of Advanced and Distributed Simulation, Proceedings, 2009: 114-121, 153
- [22] 张敏芳. 基于 REST 与 Ajax 的地理信息服务研究与应用[D]. 西安: 西安交通大学, 2010

(上接第 112 页)

- [2] 张俊. 构建 REST 风格的 Web 应用程序[D]. 北京: 北京邮电大学, 2009
- [3] 李峰, 李春旺. Mashup 关键技术研究[J]. 现代图书情报技术, 2009(1): 44-49
- [4] Fielding R. Architectural styles and the design of network-based software architectures[D]. Citeseer, 2000
- [5] 姚鹤岭. 基于 REST 的 GIS Web 服务研究[D]. 郑州: 中国人民解放军信息工程大学, 2005
- [6] 詹睿. 基于 Ajax/REST 的 GIS WEB 服务研究与实现[D]. 北京: 中国地质大学, 2008
- [7] Mazzetti P, Nativab S, Caronc J. RESTful implementation of geospatial services for Earth and Space Science applications[J]. International Journal of Digital Earth, Supplement 1, 2009, 2
- [8] 李新通, 何建邦. GIS 互操作与 OGC 规范[J]. 地理信息世界, 2003(5)
- [9] 杨靖宇, 谢超, 柯希林, 等. 地理信息服务的思考与探索[J]. 测绘工程, 2009(1)