

Wiki 远程支援终端系统的设计与实现

马 力 郑国宁 孙 煜

(海军装备技术研究所 北京 102442)

摘 要 舰员作为远程支援系统的主要使用对象,是整个系统的使用核心。舰员远程支援系统由于远离大陆,离线支援是非常重要的部分。详细描述了完整的 Wiki 支援系统的设计需求、设计方案和系统组成,并给出了关键技术的实现代码。

关键词 Wiki,远程支援,终端,系统

中图分类号 TP302.1 **文献标识码** A

Designing and Realizing Distance Support Terminal System Bases Wiki

MA Li ZHENG Guo-ning SUN Yu

(The Institute of Naval Equipment and Technology, Beijing 102442, China)

Abstract Crewman is the key points and targets of the distance support system in warship. Because far away from mainland, the system should work normally off line. The whole wiki support system, include requirement design alternative and system composition, is described in the paper, and the core codes are listed in the thesis.

Keywords Wiki, Distance support, Terminal, System

随着信息技术和网络技术的发展,以知识管理为核心的远程支援系统取得了很大的发展,已经在舰船装备保障中得到应用,包括通过卫星通信为远航编队提供技术支援,显现出了其作为信息技术保障手段的意义和作用。其中, Wiki 是远程支援系统中知识管理和共享的一项重要技术,但对远程支援系统最主要的支援对象之一:舰员的支援作用还不突出。其一直以岸基有线网络为主要运行基础设施,因此对海上舰船的支持力量受到很大的限制。鉴于此状况,研制远程支援终端软件,从而为舰员提供离线的数据资源、故障案例甚至音视频来协助舰员排除装备故障,同时,终端软件记录排故过程,终端软件连接数据库服务器后将案例上传,以为其他人员提供参考。系统设计针对便携终端的 Wiki 数据库下载、更新和离线浏览技术,并开发离线浏览软件平台,对实现远程支援系统知识库对舰船维修保障有着重大的实际意义。

1 系统设计需求

通过对国内外同类产品的性能分析、理论以及相关技术资料的收集和分析及对远程支援系统的技术状况和使用中常见问题等情况进行调研,根据主要技术指标要求,运用数据增量技术,利用加密技术,针对知识框架,采用 xml 数据格式转换技术存储 Wiki 数据库的方法研制远程支援舰员终端软件。整个系统终端的控制流程图如图 1 所示。

基于整个系统的设计需求,有如下几点主要工作内容:

(1)数据加密技术。知识管理终端是便携式终端,根据 Wiki 数据的密级要求,需要对终端内的数据做存储加密,在用户接入岸基远程支援系统时需要严格的身份认证和授权机制。

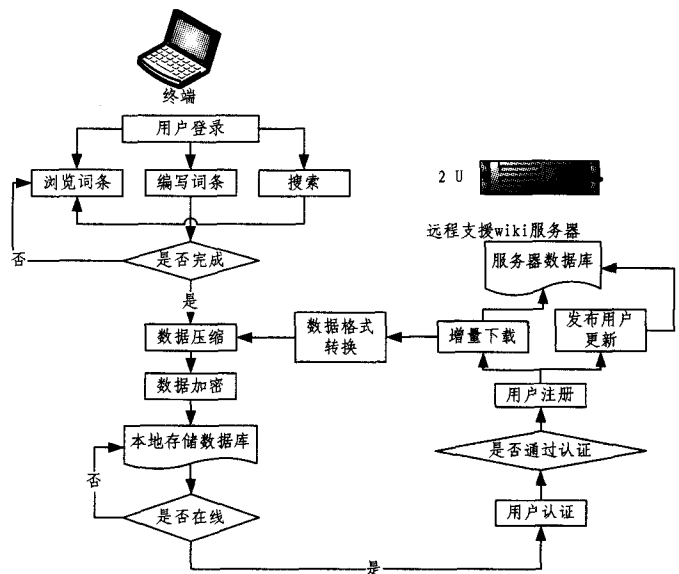


图 1 终端控制流程图

(2)增量更新技术。由于 Wiki 的数据量大,终端需要下载和存储大量的数据,对于本来网络状况就不理想的地区,要下载这么大量的数据非常困难,甚至无法完成。为了达到远程支援舰员终端与 Wiki 数据库服务器数据的同步,需要舰员级终端具备增量更新能力,以减小每次同步需要下载的数据量。

(3)数据格式转换与离线浏览技术。Wiki 数据库采取 xml 数据库存储,离线数据存储格式与 Wiki 数据库不同,在舰员级终端中,采取适当的数据格式存储数据、更适宜离线浏

马 力(1979—),女,硕士,主要研究方向为数据库与知识库;郑国宁(1970—),男,硕士,主要研究方向为网络安全;孙 煜(1975—),女,硕士,主要研究方向为装备综合测试诊断。

览是需要研究的重点内容;同时,终端软件还应该提供给舰员简便操作的数据格式转换功能,以方便用户自己进行小批量的数据格式转换。

(4)远程支援请求处理。用户可以通过便携终端创建支援请求,同时也可以参与支援请求的处理过程,在网络连通的情况下,用户创建的支援请求可以同步上传到远程支援服务器上,同时也可以下载远程支援服务器上存储的针对支援请求的处理过程信息。

2 总体方案设计

2.1 系统的总体架构

系统架构主要采用.net 框架开发,系统各层次之间功能独立且耦合度低,每个层中再划分独立模块,组件化封装,实现与接口之间的分离。这种结构不仅利于并行开发,指导并规范管理软件分析、设计、编码、测试、部署各阶段工作,提高代码正确性、可读性、可维护性、可扩展性、伸缩性等,而且有利于增强系统的扩展性,提高系统的稳定性和可维护性。系统总体架构图如图 2 所示。

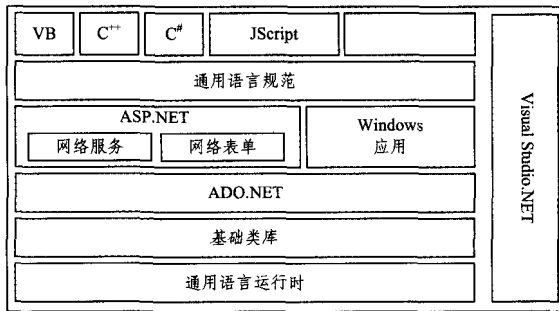


图 2 系统总体架构

2.2 系统的应用架构

系统的应用架构需要保证未来的系统在应用部署上的可行性、健壮性、灵活性和可维护性。因此远程维修支援便携终端软件平台采用 4 层的应用架构,如图 3 所示。

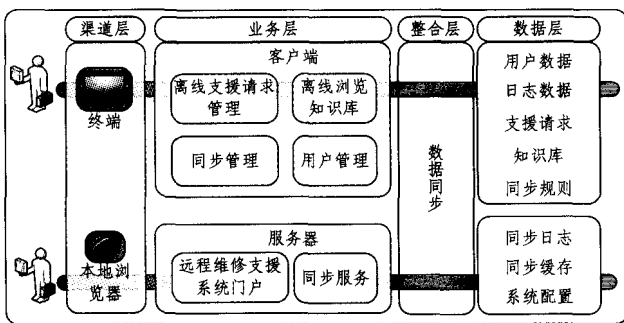


图 3 系统应用架构

渠道层:定义了舰员通过什么方式使用远程维修支援系统便携终端提供的功能。

终端:在终端上部署远程维修支援客户端,舰员通过客户端访问本地的离线支援请求、离线知识库,或在联网状态下通过客户端或者本地浏览器直接访问远程维修支援系统门户,使用远程维修支援系统提供的各种业务功能。

业务层:定义了便携终端系统的业务流程、逻辑和实现,以接受和处理舰员在终端中进行的操作,是远程维修支援便携终端系统的核心部分。

2.3 系统的数据架构

数据架构体现了系统的核心领域模型,远程维修支援终端平台系统的数据架构如图 4 所示。

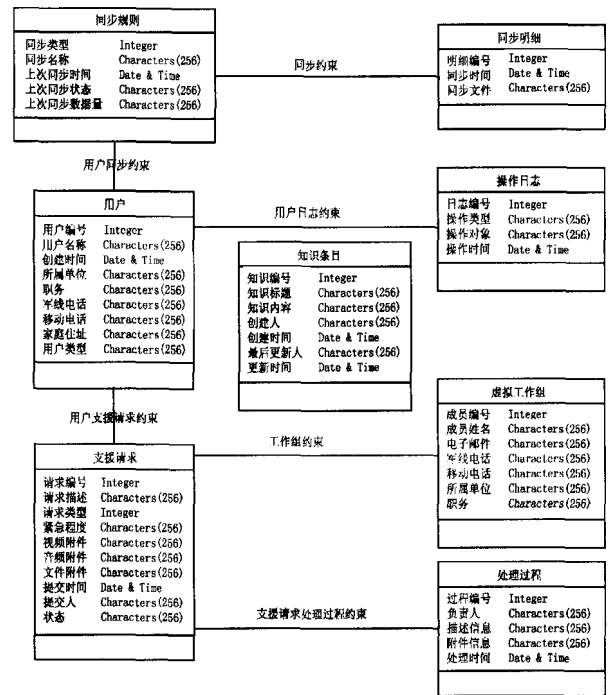


图 4 系统数据架构图

同步规则:用同步规则模型来表达舰员在进行数据同步时涉及到的数据种类不同的种类对应不同的规则。

同步明细:每种不同的同步规则会对应自己的同步历史,系统需要根据同步历史做增量同步的判断,用同步明细模型来表达同步历史。

用户:特指舰员,用户模型用来表达用户的基本属性。

操作日志:根据审计的要求,用户在终端上进行的操作需要记录,操作日志模型用来表达日志记录。

支援请求:支援请求是系统的核心领域模型,用来表达舰员创建的、存储在终端上的支援请求。

虚拟工作组:虚拟工作组模型附属于支援请求模型,用来表示所属支援请求负责人的相关信息,虚拟工作组的信息需要从服务器上同步到终端上。

处理过程:处理过程模型同样是附属于支援请求模型,用来表示虚拟工作组成员对当前支援请求的处理过程,处理过程数据需要从服务器上同步到终端上。

知识库:知识库模型是便携终端系统的核心领域模型,用于表示词条信息,需要从服务器上同步到终端上。

3 系统的设计实现

3.1 技术方案的选择

Windows 操作系统是发布最早的操作系统,也是市场占有率最大的操作系统,用户通过 Windows 操作系统已经养成了固定的操作习惯,便携终端系统采用 Windows 作为操作系统。

使用基于 Windows 的 Microsoft .NET 技术,设计 C/S 结构,在处理知识库相关内容时以 B/S 结构实现,并将其嵌入到整体的 C/S 结构中。

3.2 总体布局

现有远程支援系统为目标用户所提供的渠道已经覆盖了

现有的全部远程通信方式。但是在这些通信方式都不方便使用,或者需要用户在装备位置进行现场处理等情况下,这些渠道仍然有所不足。

根据这些情况下的特殊需求,我们认为需要对远程支援系统的渠道进行一些补充,利用便携终端来进行离线的知识管理、案例研究,辅助用户进行故障的诊断和修复;同时还应该在装备位置第一线进行离线的支援请求创建,为后面的处理流程提供更加精确的故障描述等信息。

系统总体布局描述如图 5 所示。

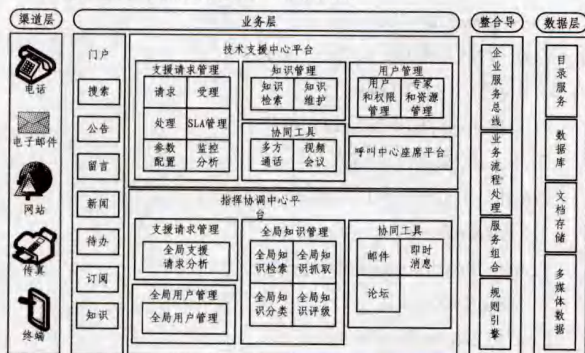


图 5 系统总体布局图

3.3 系统组成

系统由三大部分构成:1)远程支援客户端;2)Wiki 知识库管理;3)用户管理。具体内容如表 1 所列。

表 1 系统功能组成

功能模块	功能需求
远程支援客户端	在线远程支援请求处理
	离线状态下创建远程支援请求
Wiki 数据知识管理	联网状态下“离线支援请求信息”同步到远程服务器
	联网状态下远程服务器,“支援请求信息”下载更新到本地数据库
用户管理	基于终端本地数据的 Wiki 离线浏览
	Wiki 离线状态下,数据维护,贡献,交流
	联网状态下,Wiki 远程服务器数据与本地数据同步(远程增量下载,本地增量上传)
	远程服务器 Wiki 数据库 xml 格式转换
用户管理	远程支援服务器登陆身份认证
	本地离线 Wiki 数据加密
	终端系统客户端用户管理
用户管理	终端系统客户端权限设置
	本地验证和远程服务器认证

3.4 系统实现

主要软件模块包括:创建离线支援请求、浏览离线知识库、同步数据、管理用户。

(1)创建离线支援请求

①功能说明

舰员在离线状态下,通过部署在终端中的远程维修支援客户端用户界面创建一个支援请求,并保存到本地。

②用户界面设计(见图 6)

③实现逻辑

用户打开离线支援请求的界面时,自动填充当前登录账户的名字、单位、联系电话。其中,单位名称不能修改,而请求人的姓名可以修改。

紧急程度不同,会有不同的默认解决时限。在现阶段的设置中,“一般”对应 72 小时,“紧急”对应 24 小时,“加急”对应 2 小时。选择了紧急程度后,自动填好解决时限的默认值;

用户仍然可以手动调整解决时限。

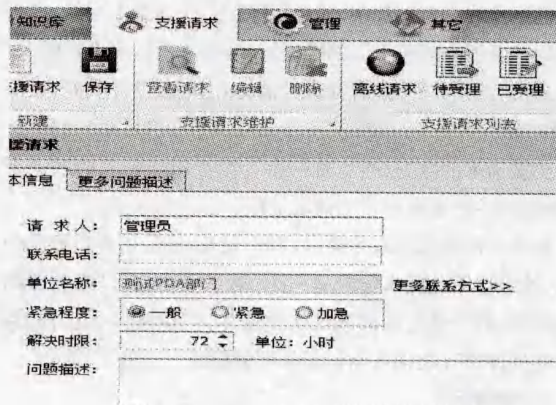


图 6 离线支援请求界面

(2)浏览离线知识库

①功能说明

舰员在离线状态下,通过部署在终端中的远程维修支援客户端浏览保存在本地知识库中的词条。

②用户界面设计(见图 7)

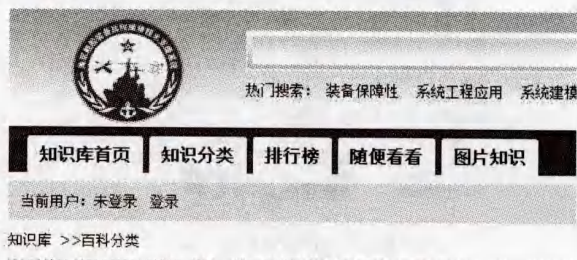


图 7 离线知识库界面

(3)同步数据

①功能说明

在联网状态下,舰员通过终端连接到远程维修支援服务器上,通过同步服务,将本地的离线支援请求上传,下载服务器上最新的知识库词条和相关的支援请求处理过程。

②用户界面设计(见图 8)

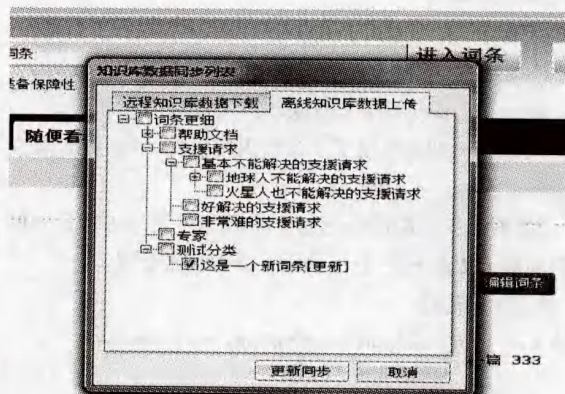


图 8 知识库同步界面

③实现逻辑设计

网络在线时启动程序,或者程序运行时连接上网络,即可使用同步功能。对于不同的数据同步,有不同的实现逻辑。

人员数据的同步是自动进行的,当出现数据冲突时,以服务器为准。

支援请求的数据,同步时将所有符合同步要求的数据下载到本地,并把本地创建的请求上传。

知识库的数据出现冲突时,对于词条,提示用户选择哪个版本,选定后用此版本覆盖另一个;对于分类,要以服务器的数据为准;对于评论和词条的历史版本,可以按照先远程、后本地的顺序来互相插入新记录从而调整成一致的。

(4)管理用户

①功能说明

管理员是一个内置用户,以管理员身份的舰员登录到远程维修支援客户端之后,可以为其他舰员创建登录账号,舰员需要持此登录账号和密码登录系统。

②用户界面设计(见图9)

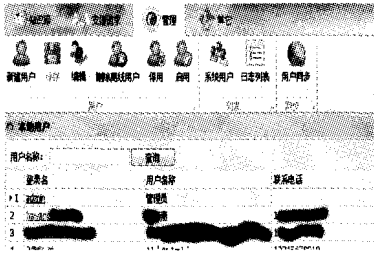


图9 用户管理界面

③实现逻辑

登录的用户如果是本终端的管理员,那么可以进行用户管理。

用户管理的数据,仅限于终端所属的单位(舰船)。

没有同步到服务端的本地账户,可以删除。

3.5 关键技术

(1)安全管理技术

①本地存储的离线知识库内容的安全性,主要依赖于数据库管理系统自身的安全机制,通过 RBAC 模式控制数据的访问权限。

根据舰员终端系统的特性,按照责权清晰的原则,需要建立两个不同的账号:

- Updater,此账号用于用户对知识库内容进行更新操作,且只对指定的 wiki_attachment、wiki_block、wiki_category、wiki_doc、wiki_datacall 有写权限,对其他表无操作权限。

- Selector,此账号用于查询知识库操作,对所有表有查询权限,但没有写入权限。

②数据同步方式的安全性,对需要通过网络进行传输的数据(包括上传和下载)进行 AES 128 位加密处理。

核心加密代码:

```
public static byte[] encrypt(String content,String password)
{ try {
    KeyGenerator kgen=KeyGenerator.getInstance("AES");
    kgen.init(128,new SecureRandom(password.getBytes()));
```

```
SecretKey secretKey=kgen.generateKey();
byte[] enCodeFormat=secretKey.getEncoded();
SecretKeySpec key=new SecretKeySpec(enCodeFormat,"AES");
Cipher cipher=Cipher.getInstance("AES");
byte[] byteContent=content.getBytes("utf-8");
cipher.init(Cipher.ENCRYPT_MODE,key);
byte[] result=cipher.doFinal(byteContent);
return result;
} catch(NoSuchAlgorithmException e) {
    e.printStackTrace();
} catch(NoSuchPaddingException e) {
    e.printStackTrace();
} catch(InvalidKeyException e) {
    e.printStackTrace();
} catch(UnsupportedEncodingException e) {
    e.printStackTrace();
} catch(IllegalBlockSizeException e) {
    e.printStackTrace();
} catch(BadPaddingException e) {
    e.printStackTrace();
}
return null;
```

(2)文件型附件的传输技术

多数情况下,数据是通过 Web Service 来传递的,但是由于 PHP 的 Web Service 库不支持 MTOM,而多个子系统集成时又需要 PHP 对 Web Service 的调用,因此远程支援系统中的 Web Service 一般不能打开 MTOM,也就不能直接在 Web Service 调用中传输附件。

解决的办法要区分各种不同的情况。

在向本地同步(即下载)文件型附件时,根据远程支援系统为浏览器提供的下载 url 来进行下载,这需要启动一个额外的后台线程,让这个下载过程不影响其他的同步行为的继续进行。例如支援请求中的附件向本地同步时,文件同步程序可以根据附件下载地址加上附件 ID 来发起 http 的 get 请求,以下载这个文件,并将其保存到本地,但是由于这个请求获得的文件名是名义上的文件名,是有可能重名的,因此还要根据数据库中附件表的内容来确定实际保存到磁盘中的文件名。

hdwiki 对于附件的设计和实现与远程支援系统完全一样,所不同的只是 url 和数据库字段,因此处理方式完全一样。

在向服务器同步(即上传)文件型附件时,要区分不同的情况。

对于支援请求,要先上传附件,获取 ID 后再同步本地的支援请求,而后把服务端的附件表同步到本地。这是因为远程支援系统对附件的处理是独立的,不论支援请求还是公告等的附件,都是同一种上传方式。

对于词条,要在服务器端编写一个 php 来接收附件并将其保存到附件目录中;本地程序在上传完附件后,再向服务端同步附件表即可。

(3)数据同步的冲突问题的处理技术

针对数据同步的冲突问题,尤其是知识库中知识条目正文的冲突问题,便携终端系统实现了正文冲突自动合并算法。算法在进行合并时分别考虑时间因素和文本差异因素,将时间较旧的版本作为基础,逐行扫描,和较新的版本进行比较,如果较新版本中当前和基础版本中的当前行不一致,则遍历基础版本,查找是否存在相同的行,如果不存在,则在基础版本中插入,如果存在则跳过,以此类推。

核心代码:

```
Public List<Attachment> GetLocalAttachments(List<string> array)
{
    List<Attachment> files=new List<Attachment>();
    StringBuilder ids=new StringBuilder("-1");
    array.ForEach(m => ids.Append(", " + m));
    DataTable dt=DataAccess. MySQLInstance. Query(string. Format
    ("select * from wiki_attachment where did in({0})",ids. ToString
    ())). Tables[0];
    Hyperats. Framework. Web. Net. HttpClient client; Attachment att
    =null;
    for(int i=0;i < dt. Rows. Count;i++)
        client = new Hyperats. Framework. Web. Net. HttpClient
    (string. Format("{0}{1}", domain,att. attachment));
    try
    {att. content=client. GetBytes();}
    catch(System. Net. WebException ex)
    {continue;}
    att. attachment2= string. Format("{0}_s{1}", att. attach-
    ment. Substring(0, att. attachment. LastIndexOf(". ")), System.
    IO. Path. GetExtension(att. attachment));
    client = new Hyperats. Framework. Web. Net. HttpClient
    (string. Format("{0}{1}", domain,att. attachment2));
    try
    {att. s_content=client. GetBytes(); }
    catch(System. Net. WebException ex)
    {continue;}
    files. Add(att);
    att. attachment3= string. Format("{0}_140{1}", att. attach-
    ment. Substring(0, att. attachment. LastIndexOf(". ")), System.
    IO. Path. GetExtension(att. attachment));
    client = new Hyperats. Framework. Web. Net. HttpClient
```

```
(string. Format("{0}{1}", domain,att. attachment2));
    try
    {att. content140=client. GetBytes();}
    catch(System. Net. WebException ex)
    {continue;}
    files. Add(att);}
    return files;
}
```

4 结果分析

便携终端系统实现了如下功能:

1)浏览离线知识库。在离线状态下,舰员可以通过客户端浏览在本地存储的知识库数据,本地知识库中的每一个具体词条信息都与服务器中的词条信息保持一致。

2)创建离线支援请求。在创建支援请求时可以实时采集视频、音频、图像,在互联网状态下,可以把离线支援请求上传到远程维修支援服务器,生成一条在线的支援请求,进入到支援请求的处理过程,同时还可以下载已上传离线支援请求的处理过程信息和对应虚拟工作组的人员信息。

3)同步数据。在互联网状态下,舰员可以通过客户端将存储在终端中的支援请求信息同步到服务器上,也可以把已经同步过的支援请求的处理信息下载到本地以供浏览;同时,还可以从服务器上下载最新的词条信息并保存到本地以供浏览。

总的说来,国外使用的软件有些类似 IETM,可以对维修诊断进行过程式处理。整个终端还是有些类似一个 PMA,并没有知识积累和知识学习的作用,同时应用的笔记本还是属于平板出现前的技术。而此次研制的便携终端系统则具有了知识管理的功能,以最大程度地方便了用户。

参考文献

- [1] 基于语义 Web 技术的 Wiki 系统研究与设计[J]. 首都师范大学学报:自然科学版,2008(01)
- [2] 闫晓妍. 基于语义 Wiki 的知识检索研究[J]. 图书馆学研究, 2010(07):75-80
- [3] Borja, Rhea R. Educators Experiment With Student-Written 'Wikis'[J]. Education Week,2006,25(30)

(上接第 279 页)

- [3] Datla D, Wyglinski A M, Minden G J. A spectrum surveying framework for dynamic spectrum access networks [J]. IEEE Trans. Veh. Technol,2009,58(8):4158-4168
- [4] Mitola J, Maguire G Q. Cognitive radio: Making software radios more personal [J]. IEEE Pers. Commun,1999,6(4):13-18
- [5] Mitola J. Cognitive radio—An integrated agent architecture for software-defined radio [D]. Sweden: Royal Institute of Technology,2000
- [6] Haykin S, Thomson D, Reed J. Spectrum sensing for cognitive radio [J]. Proceedings of the IEEE,2009,97(8):849-877
- [7] Dobre, Rajan S, Inkol R. Joint Signal Detection and Classification

- Based on First Order Cyclostationarity for Cognitive Radios [J]. EURASIP Journal on Advances in Signal Processing-special issue on dynamic,2009,2009
- [8] Porat B. Digital Processing of Random Signals: Theory and Methods [M]. New Jersey: Dover Publications,2008
- [9] Proakis J G. Digital Communications(4th edition)[M]. Beijing: Publishing House of Electronics Industry,2001
- [10] 徐以涛. 数字信号处理 [M]. 西安:西安电子科技大学出版社,2009
- [11] 盛骤,谢式千. 概率论与数理统计(第三版) [M]. 杭州:高等教育出版社,1989