

基于复杂网络的 Java 程序分析工具设计与实现

陈 焘 李孔文 王树森 顾 庆 陈道蓄

(南京大学计算机科学与技术系 计算机软件新技术国家重点实验室 南京 210093)

摘 要 随着软件产业的迅速发展,软件系统的功能和结构日益复杂,有必要有效地理解复杂软件系统的结构,理解其在生命周期里的变化规律。开发了一个基于复杂网络的 Java 程序分析工具,分析 Java 代码,为 Java 软件系统结构构建复杂网络,并进行复杂网络特征的统计和分析,为更好地研究和理解复杂软件系统的结构及其演化规律提供了有力的支持。

关键词 软件系统结构,程序分析,复杂网络

Design and Implementation of a Tool of Java Program Analysis Based on Complex Networks

CHEN Tao LI Kong-wen WANG Shu-sen GU Qing CHEN Dao-xu

(State Key Laboratory for Novel Software Technology, Department of Computer Science and Technology,
Nanjing University, Nanjing 210093, China)

Abstract With the rapid progress of software industry, large-scale software systems become dominant, and their functions and structures are usually complicated. There is a need to better understand the structures and evolutions of these systems. This paper developed a tool of Java program analysis based on complex networks. This tool builds complex networks for Java programs, and calculates the complex networks' features existed in Java programs' structures. It provides an effective support for researching and understanding the structures and evolutions of complex software systems.

Keywords Software systems structures, Program analysis, Complex networks

1 引言

随着计算机技术的迅速发展,软件的价值日益增长,软件企业希望能够加速软件的开发,不断寻求提高软件质量、降低软件开发成本和时间的办法,也就需要能更好地了解和把握软件系统的结构。所以运用成熟的技术对软件系统进行可视化建模,详细规划系统的设计,组织和编辑系统开发各阶段的信息,将有利于维护和管理软件系统,有利于开发工作的延续。

但随着软件系统应用领域和规模的不断扩大,现代软件系统越来越复杂,其结构越来越难以理解:一方面,组成系统的单元数量的急剧增加,使得对软件系统建模难度加大,软件系统开发、测试以及维护工作也变得困难;另一方面,尽管有对软件系统良好的设计,也有成熟的技术明确了各个组成单元之间的交互关系,但是受到不断变化因素的影响(如需求、环境的改变),各个单元常常需要频繁地进行扩展、组合、重构等,导致最后形成的系统结构往往比较混乱、复杂。因此,我们需要探索研究大规模软件系统结构复杂性,解决不断出现的系统结构难题,了解软件系统在其生命周期的变化规律,再与已有技术结合,从而加速软件的开发。

近年来,复杂网络受到越来越多人的关注,人们对复杂网络的研究也越来越多,并且在生态学、统计物理学、社会学等

领域得到了巨大的发展和延伸。研究表明^[3],真实世界中的复杂系统,比如网页中超链接之间的引用关系、人与人之间的社会关系、蛋白质之间的相互作用关系、计算机网络等,不能用规则网络和随机网络来描述,但可以由性质介于规则网络和随机网络之间的网络来描述,这种网络被称为复杂网络。复杂网络通常具有小平均路径长度和大平均聚集系数的小世界效应(small-world effect),以及幂律度分布的无标度特性(scale-free nature):Watts 和 Strogatz^[1]提出了小世界网络模型,构造了这种介于规则网络和随机网络之间的复杂网络;Barabási 和 Alber^[2]进一步指出了复杂系统中的无标度特性,提出了更符合真实网络的基于增长和优先连接机制的 BA 模型,描述了复杂网络的形成过程。

复杂网络为描述和研究复杂系统的整体行为和演化规律提供了有力的工具。而软件系统是由大量不同粒度的单元,以及这些单元之间的交互关系组成,这说明复杂软件系统也可以表示为复杂网络,因此复杂网络也为研究复杂软件系统提供了全新的视角。近年,大量的研究者开始基于复杂网络对软件系统结构进行研究。Myers^[4]指出软件系统由许多协作的单元组成,研究了软件系统结构的复杂网络特性;Wheeldon 和 Counsell^[5]研究了面向对象的一些度量的幂律关系等等。

但复杂网络的“复杂性”,使得人们需要开发辅助复杂网

到稿日期:2008-05-05 本文得到国家“863”项目(编号:2006AA01Z177),江苏省自然科学基金基础研究项目(编号:BK2006115)支持。

陈 焘 硕士研究生,研究方向为分布式计算与并行处理,E-mail: andreict@dislab.nju.edu.cn;顾 庆 副教授,研究方向为分布式计算与并行处理;陈道蓄 教授,博士生导师,研究方向为分布式计算与并行处理。

络研究的工具,进而节省研究者大量的精力,更能科学和高效地反映、揭示复杂网络的各种特征和变化规律。本文设计与实现一个基于复杂网络的 Java 程序分析工具(Java Program Analysis based on Complex networks,简记为 JPAC)来研究复杂软件系统的结构:分析 Java 程序的代码,为 Java 软件系统结构构建复杂网络,并统计其复杂网络特征,研究软件系统结构的演化规律。JPAC 为 Java 软件系统的复杂网络研究提供了有力的支持。

本文第 2 节概述了 JPAC 工具的基本原理和总体结构。第 3 节详细描述了 JPAC 工具各个层次的设计与实现。在此基础上,第 4 节中给出使用 JPAC 工具分析 JDK(Java Development Kits)的实例。第 5 节对已有工具做了介绍和对比。最后是对本文的总结。

2 JPAC 工具基本原理和总体结构

2.1 软件系统结构

JPAC 工具的主要目的是为 Java 软件系统结构构建复杂网络。Myers^[4]定义了一个能够表示软件系统结构的网络图——软件协作图(Software Collaboration Graph,简记为 SCG),它是类似于 UML 中类图的一种结构。文献[7]中,进一步将 SCG 细化为能够描述 Java 软件系统结构的 Java SCG,图 1 显示了一个简单的 Java SCG 的构建实例。

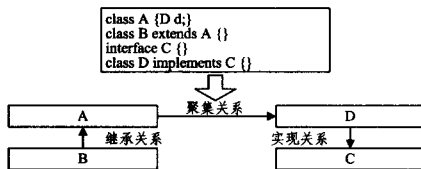


图 1 一个简单的 Java SCG 的构建实例^[7]

将构建的 Java SCG 看作是能够表示 Java 软件系统结构的网络,可以对该网络的特征进行分析。如图 2 所示,本文主要从两个角度进行了统计分析:元素级特征和网络级特征。元素级的特征主要是指构成网络的最基本的单元的特征,包括节点和边。网络级特征是指全局的特征。正是这些特征反映了 Java 复杂软件系统中的复杂网络现象。

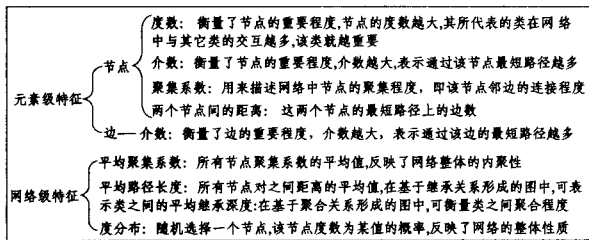


图 2 网络的统计特征

2.2 软件系统结构的演化

复杂网络又具有演化的性质,按照复杂网络演化模型演化的网络,在演化后依然具有复杂网络的特征。

文献[7]中把复杂网络演化的概念运用到软件系统结构当中,提出了一种基于模块的软件演化模型。此演化模型模拟了对软件系统结构的复杂网络特征的形成过程,与复杂网络的演化一样,软件系统结构按照此模型演化后得到的网络依然具有复杂网络的特征。此模型的具体流程如图 3 所示,其中 p_1 , p_2 和 p_3 表示了相应过程被执行的概率,而在连接新

边时考虑了软件系统结构模块的存在。

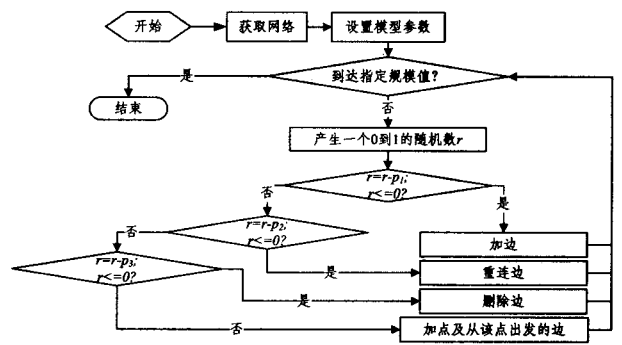


图 3 基于模块的软件演化模型演化过程

2.3 工具总体结构

基于前两个小节的考虑,JPAC 工具需要实现对 Java 程序的代码进行分析,为 Java 软件系统结构构建复杂网络功能。并对构建好的复杂网络,进行复杂网络特征分析,显示软件系统结构的复杂网络特征,模拟软件系统结构的演化规律。同时 JPAC 工具也提供了抽象的复杂网络,使得人们可以对其特征和演化规律进行模拟研究。因此,我们可以将 JPAC 工具功能按层次进行划分,其总体结构如图 4 所示,主要分为三个层次模块:

(1)数据分析层。作为工具的最底层,它主要与 Java 代码文件或者 XML 文档进行交互,封装了 Java 代码和 XML 文档到网络计算机表示的构造过程。它可以不依赖于高层(数据运算层和用户界面层)的存在,因此用户可以根据需要增加基于数据分析层的各种功能。

(2)数据运算层。它实现了基于数据分析层提供的网络的功能运算,比如统计分析、演化模拟等运算。这里运算是指对网络计算机表示进行的操作和计算。数据运算层是可扩展的层次。

(3)用户界面层。它处于最上层,它负责将网络计算机表示可视化,将运算结果可视化,为用户提供操作等。

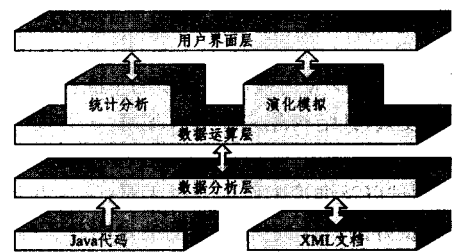


图 4 JPAC 工具总体结构

3 JPAC 工具详细设计与实现

3.1 数据分析层

3.1.1 主要结构

JPAC 工具的数据分析层流程如图 5 所示,它主要包括了对 Java 代码的分析、对 XML 文档的存取、生成网络的计算机内部表示三个功能。数据分析层读入 Java 代码或者 XML 文档,如果读入 Java 文件,判断是源文件还是字节码文件,如果是字节码文件则直接用 BCEL 技术进行分析,如果是源文件,先将其编译成字节码文件再用 BCEL 技术进行分析。

然后根据 BCEL 技术获取的类信息构建网络图。如果读入的是 XML 文档,则采用 DOM 技术进行分析,进而构建网络。构建的网络可以转换为 XML 文档进行存储,也可提供给系统结构中的高层使用。BCEL 和 DOM 技术将在稍后介绍。

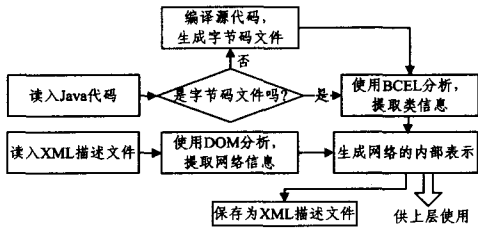


图 5 JPAC 工具数据分析层流程图

数据分析层的主要类图如图 6 所示,每个类的含义如下:

- ProgInput 是读入并解析 Java 文件的抽象类,提供了读入文件的抽象方法,被具体实现时,实现了对 Java 文件的解析。
- XMLParse 给出了读入和保存 XML 文档的实现,基于 DOM 技术对 XML 文档进行解析。
- DataLayer 是保存构建的网络的抽象类。
- JavaBinDataLayer 是保存基于 Java 系统结构构建网络的具体类。
- RandomDataLayer 是保存随机网络的具体类。

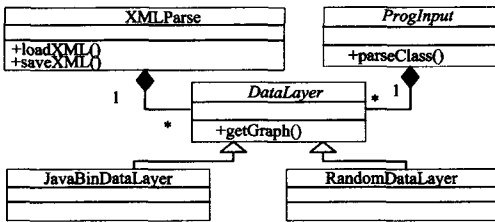


图 6 数据分析层主要类图

其中读入 Java 代码文件的主要类图如图 7 所示,每个类的含义如下:

- CustomVMConst 给出了基于 Java 虚拟机读取输入的具体实现。
- CustomJavaClassConst 给出了由 Java 字节码文件读取输入的具体实现。

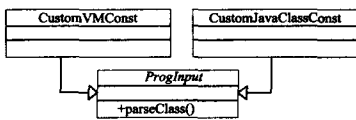


图 7 Java 文件输入的主要类图

读入和保存为 XML 文档的主要类图如图 8 所示,每个类含义如下:

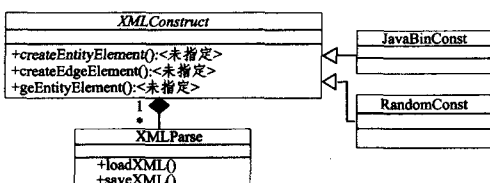


图 8 XML 文档输入输出的主要类图

- XMLConstruct 是构造 Java SCG 的抽象类。
- JavaBinConst 具体实现了基于 Java 程序的 XML 文档的 Java SCG 的构造。
- RandomConst 具体实现了基于随机网络的 XML 文档的 Java SCG 的构造。

3.1.2 主要技术

数据分析层主要包括了对 Java 代码的分析、对 XML 文档的存取、生成网络的内部表示三个主要功能:

(1) 对 Java 代码的分析

从图 5 中的流程可以看出,JPAC 工具实际上只针对 Java 的字节码文件进行分析,处理 Java 源代码文件时,先进行了编译过程。这样做的主要原因是:Java 源代码需要成为 Java 平台自带的可执行代码,这个过程一般包含了两个步骤,开发人员将源代码编译成 Java 字节码(*.class 文件),然后 Java 虚拟机将其转换为主机平台的本地代码。所以,字节码文件是编译源文件得到的,既包含了一定的 Java 代码的信息,无需从源代码出发再做重复的工作,又因一个字节码文件只对应到一个类,方便在程序中对大量的类进行分析和处理。

对 Java 字节码文件的两种分析方式:通过 Java 虚拟机进行分析和直接对 Java 字节码文件进行分析。前一种方式受到 Java 虚拟机性能的限制,因此 JPAC 工具采用后一种方式,直接对 Java 字节码文件分析,使用了 BCEL (Byte Code Engineering Library) 技术。

BCEL 是 Apache Software Foundation 的 Jakarta 项目的一部分,已经加入到 Java5.0 当中。因为 BCEL 可以深入 Java 类字节码,转换现有的类表示或者构建新的类,可以在单独的 Java 虚拟机指令级别上进行操作,所以对代码有最强大的控制。

BCEL 分析的起点是 JavaClass 类。JavaClass 封装了得到被分析的字节码对应的类的字段和方法信息,以及获取父类和接口的结构信息的方法。还提供了对类的内部信息的访问,包括常量池和属性,以及作为字节流的完整二进制类表示。

根据 BCEL 提取到的 Java 软件系统结构中各个类的信息,可以为 Java 软件系统构建 Java SCG (见第 2 节)。

(2) 对 XML 文档的存取

XML (eXtensible Markup Language, 扩展标记语言),是一种简单的数据存储语言,它使用一系列简单的标记来描述数据,尽管占用的空间比二进制数据占用的要多,但其简单性使得其易于在任何应用程序中读写数据,成为了一种跨平台的数据组织形式。因此,JPAC 工具选择了 XML 文档作为 Java SCG 的持久化存储形式。

JPAC 工具为 Java SCG 定义的 XML 文档格式,如图 9 所示。XML 文档按照 Java SCG 邻接表的形式主要描述了所有节点,包括每个节点(类)的名字、继承的父类、实现的接口、包含的域和方法、图形显示时的位置,以及从该节点出发的边的信息。边的信息包括了出发节点名和到达节点名。

```
<? xml version="1.0" encoding="iso-8859-1"?>
<cnml >
  <dnode name="A">
    <entity name="A" type="Class">
      <superclass>.....</superclass>
      <interface>.....</interface>
```

```

    <method>.....</method>
    <field>.....</field>
  </entity>
  <position>.....</position>
  <relation name="A_B">.....</relation>
</cnode>
<cnode name="B">
  .....
</cnode>
.....
</cnml>

```

图9 Java SCG的XML文档

JPAC工具根据XML文档建立Java SCG,往往需要一次性读入整个XML文档。因此JPAC工具使用DOM(Document Object Model)技术来解析XML文档。DOM是将XML文档一次性解析并读入内存,保存为一个文档对象供用户访问,为每个元素、文本、属性等都创建了对象。当XML文档对象转化为Java SCG后,则释放XML文档对象所占空间,以备他用。

JPAC工具遵循了XML的规范,以XML文档作为输入输出,实现了网络的持久化存储,便于网络的再次使用和研究。与对Java代码的分析结果一样,对XML文档的分析结果也是构建Java SCG。

(3)生成网络在计算机内部的表示

前述两个功能最终都是构建Java SCG,实际上JPAC工具正是将Java SCG作为网络在计算机内部的表示,具体如图10所示。Java SCG用Hashmap保存大量的节点,每个节点也用Hashmap保存从该节点出发边,因此Java SCG是一个邻接表的形式。

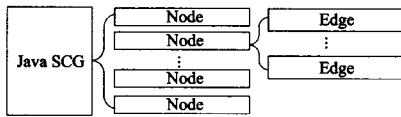


图10 Java SCG的计算机内部表示

人们也需要建立一个完全抽象的网络来进行比较分析,如果抽去Java SCG包含的软件系统结构的含义,就可以形成这种抽象网络。

JPAC工具在数据分析层实现了随机网络的构造,它是指根据ER模型^[6]来构造随机网络,即用户给定节点数 N 、边连接概率 p (任两节点之间以概率 p 决定是否相连),生成网络。这部分功能包含在RandomDataLayer类中,如图6所示。图11和图12显示了JPAC工具的随机网络的构造功能。

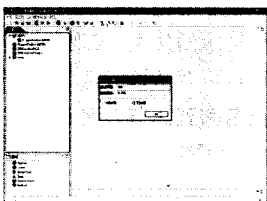


图11 随机网络的参数设置(节点数和连接概率)

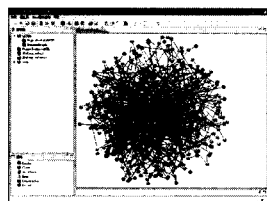


图12 节点数为500,连接概率为0.001随机网络

3.2 数据运算层

由上一小节可知,数据分析层是一个独立的层次,它为上层提供了可分析的网络Java SCG。基于Java SCG,数据运算层可进行各种运算,同时它又是一个可扩充的层次,用户可以根据需要进行功能的扩充。目前JPAC工具实现了统计分析和演化模拟运算。

(1)统计分析运算:JPAC工具主要从两个方面计算了Java SCG的网络统计特征:元素级特征和网络级特征,如图2所示。图13显示了这一运算的主要类图。

- DegreeData 实现度分布的统计,包括出度、入度和总的度分布。
- StaGraph 实现了网络级特征的统计,包括平均聚集系数和平均路径长度。
- VertexSta 实现了元素级中有关节点的特征的统计,包括介数、聚集系数、所有节点对的距离。
- EdgeSta 实现了元素级中有关边的特征的统计,主要包括介数。

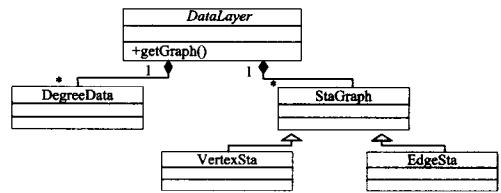


图13 统计分析运算的主要类图

(2)演化模拟运算:JPAC工具中实现了对文献[7]中提出的一种基于模块的软件演化模型的演化模拟。这一运算的流程如图3所示。

3.3 用户界面层

JPAC工具的界面使用Eclipse RCP框架。RCP(Rich Client Platform)是基于Eclipse的一个开发富客户端的应用平台,它为开发人员提供了一个功能强大的、快速的、扩展的应用平台。使用Eclipse RCP开发,可以使我们的工具具有以下优点^[9]:

- 基于插件的开发,使得我们的工具可以通过扩展点进行配置,用户可以方便地对工具进行扩展。
- Eclipse和Java的跨平台性,可以让我们的工具在各种平台下,甚至可以在嵌入式设备、掌上电脑上运行。并且Eclipse为各种操作系统提供了本地图形接口包。当RCP运行时,Eclipse首先直接调用本机窗口组件,只有没有本机所需组件时才进行模拟,具有本地观感。
- Eclipse的不断发展,基于Eclipse RCP的插件越来越多,将有利于我们工具的扩展和性能的提高。

因此,使用Eclipse RCP框架来开发JPAC工具的界面,有利于对JPAC工具的拓展,便于研究工作的延续。图14展示了JPAC工具的主要功能界面,它显示了网络的节点信

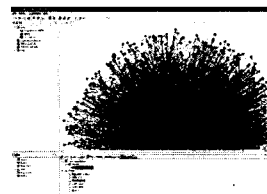


图14 显示节点信息的界面

息。在下节中还可以看到,图 15 是 Java SCG 的显示界面,图 16 是 Java SCG 对应度分布图的直角坐标系图,以及网络级的统计特征,图 17 是对应度分布的双对数坐标系图。

4 实例研究

本节使用 JPAC 工具,以 JDK 的 1.5.0 的版本为例进行了分析,构建其 Java SCG,计算其复杂网络的统计特征:平均路径长度、平均聚集系数,展示了其度分布的无标度特性。图 15 显示了对 JDK1.5.0 中 java 包分析得到的 Java SCG,可以看到网络的中心是一个度数非常大的节点,实际对应为 java.lang.Object 类,靠近中心还有少量度数较大的节点,网络周围则存在着大量度数非常小的节点,这些都是无标度特性的迹象。图 16 是图 15 中 Java SCG 的度分布的直角坐标系图,而图 17 给出了图 16 中度分布的双对数坐标图,其中展示了出度分布、入度分布、总度分布三个度分布,可以清晰地看到三个度分布都近似一条下降直线,即服从衰减的幂律分布。而相近规模的随机网络的度分布(如图 18 所示)则是一个泊松分布。表 1 显示了对 JDK1.5.0 java 包的 JSCG 的出度标度指数 γ_{out} 、入度标度指数 γ_{in} 、总度标度指数 γ_{all} 、平均路径长度和平均聚集系数等特征值的统计结果。

表 1 JDK1.5.0 的 java 包的 JSCG 特征数据

JDK1.5.0	package java
节点数	2138
边数	4708
平均度数	4.4041
平均路径长度	3.0259
平均聚集系数	0.5695
λ_{out}	2.6331
λ_{in}	1.0721
λ_{all}	1.3339

表 2 JDK java 包的 JSCG 演化模拟的特征数据

JDK 模拟	1.2.2	1.3.0	1.4.2	1.5.0	1.6.0
节点数	1128	1236	1777	2138	2363
边数	2496	2750	3984	4702	5257
平均度数	4.4255	4.4498	4.48396	4.3985	4.4494
平均路径长度	2.9225	2.9932	3.1393	3.3428	3.4426
平均聚集系数	0.5803	0.5606	0.4957	0.5108	0.4704
λ_{out}	2.4154	2.4562	2.5857	2.7168	2.7556
λ_{in}	1.0374	1.0402	1.0113	0.9042	0.923
λ_{all}	1.3139	1.337	1.2748	1.27	1.3166

5 相关工具

复杂网络研究中最常使用的是 Pajek 工具^[8],它是由卢布尔雅那大学于 1997 年正式发布的,基于 Windows 平台开发的复杂网络分析软件。它能够处理大型数据集,并支持网络分析和可视化的。

Pajek 工具支持 4 种生成网络的方式:自动生成指定模型的网络;从文件中读入 ASCII 网络数据;以其他格式从软件包中读入数据;打开 Pajek 项目文件(PAJ),它把所有不同的数据结构合并成一个文件。

Pajek 工具可以对生成的网络结构进行操作,比如分区、转置、层级和向量等。同时又有很强的可视化功能,比如,网络由有向图变成无向图,可以添加或删除边,通过减少节点或删除节点使网络简化,演示网络演化等,这些过程都可以可视化,它支持二维和三维网络的可视化,如图 19 所示。

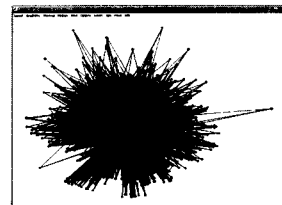


图 19 Pajek 工具截图

Pajek 工具具有强大的图像处理能力,绘图窗口给用户提供了很多选项来处理图表。它也支持 3D 可视化。这种可视化能使用多种格式存储:EPS,SVG,KIN,BMP 以及 VRML。Pajek 工具中的每个数据对象都拥有它自己的描述方法,很多方法可以用于描述网络、实例、度的计算、深度、核心或类、中心度、发现关系类型(强、弱、连接、对称)、路径或流、结构空洞和对两个网络的一些二元操作。

Pajek 工具主要针对的是抽象网络的分析,不考虑网络的实际意义,不能用来解析软件系统的结构。同时,我们也搜寻了其他的复杂网络工具,也不能够解析软件系统结构。因此有必要开发一个能够分析软件系统结构的复杂网络的工具,并进行复杂网络特征和演化方面的分析。

本文设计与实现的 JPAC 工具正是实现了解析软件系统

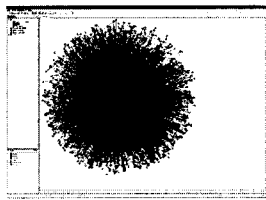


图 15 JDK1.5.0 中 java 包的 Java SCG

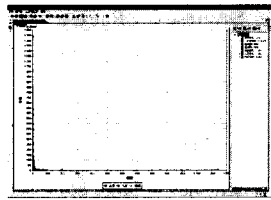


图 16 JDK1.5.0 中 java 包的 Java SCG 的度分布(直角坐标)

这些实验结果都验证了 JDK 系统结构的网络既具有小世界效应,也具有幂律的度分布特征,即说明 JDK 系统具有复杂网络的特征。而 JDK 系统是一种常见的大规模 Java 软件系统,因此在一定程度上可以说,复杂软件系统中存在着复杂网络现象。

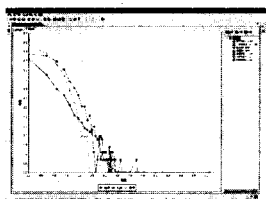


图 17 JDK1.5.0 中 java 包的 Java SCG 的度分布(双对数坐标)

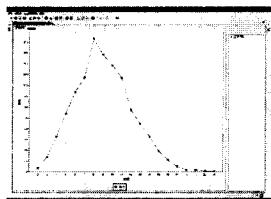


图 18 与 JDK1.5.0 中 java 包的 Java SCG 相近规模的随机网络的度分布

而软件系统结构的复杂网络也应具有演化的特性。由 JDK1.2.2 版本开始,按照基于模块的软件演化模型^[7]进行演化模拟,当达到与 JDK1.6.0 相近规模(平均度数相近)时停止。表 2 显示了模拟演化过程中记录的与各个 JDK 版本相近规模时的复杂网络特征值。可以看出演化后的网络仍然是一个复杂网络。

结构,构建复杂网络的功能。但 JPAC 工具对抽象网络本身的分析深度和广度有待完善和扩充。

结束语 本文描述了基于复杂网络的 Java 程序分析工具(JPAC)的基本原理、总体结构,以及更个层次模块的详细设计与实现。JPAC 工具能够分析 Java 程序的代码,为 Java 软件系统结构构建复杂网络,统计并展示其复杂网络特征,模拟软件系统结构的演化规律。同时,JPAC 工具也提供了抽象的复杂网络,使得人们可以对其特征和演化规律进行模拟研究。将来的工作将会进一步完善 JPAC 工具,不断扩充其功能,增加其对复杂网络的分析深度和广度。

参考文献

[1] Watts D J, Strogatz S H. Collective dynamics of 'small-world' networks. *Nature*, 1998, 393: 440-442
 [2] Barabási A-L, Albert R. Emergence of scaling in random networks. *Science*, 1999, 286 :509-512

[3] Albert R, Barabási A-L. Statistical mechanics of complex networks. *Rev. Mod. Phys.*, 2002, 74: 47-97
 [4] Myers C R. Software systems as complex networks: Structure, function, and evolvability of software collaboration graphs. *Phys. Rev.*, 2003, E 68, 046116
 [5] Wheeldon R, Counsell S. Power Law Distribution in Class//Proc. Third IEEE Int'l Workshop Source Code Analysis and Manipulation. 2003
 [6] Erdős P, Rényi A. On random graphs. *Publicationes Mathematicae*, 1959, 6: 290-297
 [7] 陈焘, 王树森, 顾庆, 等. 基于复杂网络的 JDK 代码结构演化研究[J]. *电子学报*, 2007(12A): 47-53
 [8] 颜端武, 王曰芬, 李飞. 国际人际网络分析的典型软件工具[J]. *现代图书情报技术*, 2007, 9
 [9] 陈冈. Eclipse RCP 应用系统开发方法与实践[M]. 北京: 电子工业出版社, 2007

(上接第 120 页)

形定位算法的性能,利用 Matlab 建立了一个在 50m×50m 范围内统一的仿真环境。锚节点均匀地分布在这个范围内,每一个未知节点都可以接收到全部锚节点的信息。加入干扰的高斯分布噪声,锚节点周期发送自身信息。仿真采取对 100 个在测试范围内平均分布的未知节点的结果取算术平均作为最后的结果。仿真结果分别如图 2、图 3 所示。

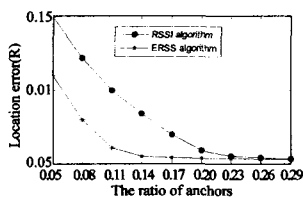


图 2 算法定位精度比较

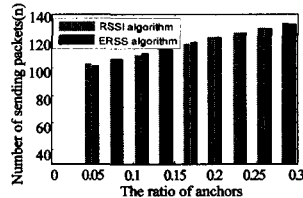


图 3 算法通信开销比较

图 2 显示,在锚节点比例小于 0.17% 时,ERSS 算法定位精度明显高于 RSSI 算法。随着锚节点的增加,两种算法的定位精度都不断得到提高。当锚节点比例趋近 0.3,定位精度趋近恒定值。这是因为这部分距离误差是由路径损耗传播模型决定的。从图中可以得出,对于相同定位精度,RSSI 是通过多次迭代计算得到的,相应也就延长了系统的定位时间和增加能量消耗,RSSI 是通过牺牲时间和能量来提高系统定位精度的。图 3 为 ERSS 测距算法和 RSSI 测距算法通信量比较。图中显示 ERSS 算法不增加通信开销,这是因为 ERSS 算法主要通过判断锚节点和未知节点的位置关系来提高定位精度,与 RSSI 比较没有额外通信开销。

结束语 本文提出了一种新的定位算法,即基于测距 RSSI 的新的定位算法 ERSS,其基本思想是通过通过对 RSSI 测距过程中产生的距离误差进行分析,推导出未知节点和锚节点在某种分布关系下未知节点的距离误差最小。为了更好地适应传感器网络的不规则性,ERSS 算法加入了角度阈值条件。该算法对硬件的要求不高。仿真结果验证了在不增加网络通信量情况下 ERSS 较普通的 RSSI 测距精度有了明显的提高,适用于大多数无线传感器网络定位的测距要求。

参考文献

[1] Akyildiz I F, Su Weilian, Sankarasubramaniam Y, et al. A Sur-

vey on Sensor Networks[J]. *IEEE Communications Magazine*, 40(8):102-114
 [2] Rabay J J, Ammer M J, da Silva J L Jr, et al. Picoradio Supports Ad Hoc Ultra-low Power Wireless Networking [J]. *Computer*, 2000, 33(7): 42-48
 [3] Savarese C, Rabay J M, Beutel J. Locationing in Distributed Ad-Hoc Wireless Sensor Network[C]// Proc. of the 2001 IEEE Int'l Conf. on Acoustics, Speech, and Signal. Salt Lake: IEEE Signal Processing Society, 2001, 4: 2037-2040
 [4] He T, Huang C D, Blum B M, et al. Range-free localization schemes in large scale sensor networks//Proc. of the 9th Annual Int'l Conf. on Mobile Computing and Networking. San Diego: ACM Press, 2003: 81-95. http://www.cs.virginia.edu/~th7c/paper/APIT_CS-2003-06.pdf
 [5] Bahl P, Padmanabhan V N. RADAR: an in-building RF-based user location and tracking system// Proc. of the IEEE INFOCOM. Tel Aviv: IEEE Computer and Communications Societies, 2000: 775-784
 [6] Meguerdichian S, Slijepcevic S, Karayan V, et al. Localized algorithms in wireless ad-hoc networks: Location discovery and sensor exposure//Proc. of the 2nd ACM Int'l Symp. on Mobile Ad Hoc Networking & Computing. Long Beach: ACM Press, 2001. 106-116. <http://www.vs.inf.ethz.ch/edu/SS2005/DS/papers/sensormetze/meguerdichian01c.pdf>
 [7] Li D, Wong K D, Hu Y H, et al. Detection, Classification, and Tracking of Targets[J]. *IEEE Signal Processing Mag*, Mar. 2002: 17-29
 [8] Niculescu D, Nath B. Ad Hoc Positioning System. (APS) [C]// IEEE GLOBECOM 2001: 2926-2931
 [9] Bergamo P, Mazzini G. Localization in sensor networks with fading and mobility//Proc. of the 13th IEEE Int'l Symp. on Personal, Indoor and Mobile Radio Communications. Lisbon: IEEE Communications Society, 2002, 2: 750-754. <http://www.eyes.eu.org/publications/pimrc2002.pdf>
 [10] Savarese C, Rabay J, Langendoen K. Robust positioning algorithms for distributed ad-hoc wireless sensor networks//Ellis C S, ed. Proc. of the USENIX Technical Annual Conf. Monterey: USENIX Press, 2002. 317-327. <http://rama.pds.twi.tudelft.nl/~koen/papers/robust-positioning.pdf>