

基于复杂网络的软件体系结构复杂性度量方法

秦怀斌^{1,2}, 李道亮¹, 郭 理², 梁 斌², 邵明文²

(1 中国农业大学 信息与电气工程学院, 北京 100193; 2 石河子大学 信息科学与技术学院, 新疆 石河子 832003)

摘 要: 介绍了复杂网络理论及软件体系结构复杂性度量思想, 提出借助复杂网络理论来进行软件体系结构复杂性度量, 给出了具体度量过程和算法, 最后对提出的度量算法通过实例进行验证。

关键词: 复杂网络; 软件体系结构; 复杂性; 度量

中图分类号: TN402

文献标识码: A

文章编号: 1000-7180(2013)02-0005-04

Complexity Metrics Method for Software Architecture Based on Complex Networks

QIN Huai-bin^{1,2}, LI Dao-liang¹, GUO Li², LIANG Bin², SHAO Ming-wen²

(1 College of Information and Electrical Engineering, China Agricultural University, Beijing 100193, China;

2 College of Information Science and Technology, Shihezi University, Shihezi 832003, China)

Abstract: This paper introduces the theory of complex networks and the complex metrics thought of software architecture, puts forward the complexity metrics method of software architecture based on the theory of complex networks, and gives a specific metrics process and algorithm. Finally, verifies the algorithm by example.

Key words: complex networks; software architecture; complexity; metrics

1 引言

自上世纪 90 年代软件体系结构(Software Architecture, SA)提出以来, 它已成为软件工程领域的研究热点. 作为系统设计的高层抽象, SA 体现了系统设计的基本思想和系统开发中的早期决策, 对系统具有深远影响. SA 在明确了系统各个组成部分的同时, 也限制了各部分之间的交互方式. 作为系统的蓝图, SA 的复杂性直接影响到后续设计、编码、测试、维护等阶段的工作. 对所开发系统的 SA 的复杂性进行分析度量, 尽早发现与把握系统可能的复杂度, 并通过特定的方法和技术来改善与控制系统复杂性, 显得尤为重要^[1].

复杂性科学是一种新兴的边缘、交叉学科, 国内外许多学者认为复杂性科学是 21 世纪科学技术发

展的前沿^[2]. 当一个系统由大量相互作用的构件构成, 其整体的复杂性超过了对其构成构件行为的简单叠加, 那么, 这个系统就是复杂系统^[3-5]. 复杂网络打破了线性、均衡、简单还原系统的传统模式, 致力于复杂系统非线性和非均衡方面的研究. 复杂网络不仅研究构成系统各个构件的复杂性, 更重要的是研究系统整体的复杂性^[6-7].

鉴于此, 论文将复杂网络与软件体系结构复杂性度量相融合, 通过复杂网络理论与方法进行软件体系结构复杂性度量, 以此从较高的层次理解系统的复杂性.

2 复杂网络及软件体系结构复杂性

2.1 复杂网络

复杂网络借助于图论和统计物理方法来捕捉并

收稿日期: 2012-05-21; **修回日期:** 2012-06-11

基金项目: 国家自然科学基金(60963006); 教育部人文社科研究基金新疆项目(11XJJAZH001); 国家科技支撑计划项目(2007BAH12B07); 石河子大学科学技术研究发展计划(ZRKX2010YB18)

描述系统的演化机制、演化规律(结构)和整体行为(功能)^[5-6]. 在复杂网络描述中, 节点代表系统中不同的个体, 边表示个体间的特定关系. 用二元组 (V, E) 来表示复杂网络, 其中 $V = \{V_1, V_2, \dots, V_n\}$ 为节点集合, $E = \{E_1, E_2, \dots, E_n\}$ 为边集, V 中元素称为节点, E 中的元素称为边, $N = |V|$ 表示节点总数, $L = |E|$ 表示边总数. E 中的每条边 E_i 有 V 的一对节点 (u, v) 与之对应, 如果 E 中任意的节点对 (u, v) 和 (v, u) 对应同一条边, 则该网络称为无向网络, 否则为有向网络; 如果 E 中的所有边的长度均为 1, 即 $|E_i| = 1$, 则称网络为无权网络, 否则为加权网络. 在无向网络中, 一个节点对最多只能被一条边连接.

在有向复杂网络中, 节点 V_i 的度指与此节点连接的边的数量, 节点的入度为从其他节点指向该节点的边数, 节点的出度为从该节点指向其他节点的边数; 节点的度分布用函数 $P(k)$ 描述, 表示一个随机选定的节点恰好有 k 条边的概率, 即度为 k 的概率 $P(k)$ 随节点度 k 的变化规律. 统计表明, 节点的度分布主要有两种形式: 一是指数分布(泊松分布), 另一种是幂律分布.

簇系数(Clustering Coefficient)用于衡量复杂网络集团化程度, 即复杂网络中节点之间的连通程度, 或网络节点的局部传递程度. 平均路径长度(Average Path Length)指复杂网络中所有节点对之间的平均最短距离. 最短距离是指从一个节点到另一个节点所经过的边的最小数目. 大量研究表明, 真实复杂网络具有较大的簇系数和较小的平均路径长度.

2.2 软件体系结构复杂性

最近几年, 部分研究者借助复杂网络理论对大量软件系统的类图进行研究^[8-9], 发现软件的 SA 并不是随机和无序的, 大多具有“小世界”和“无标度”等复杂网络的特性, 即具有较大的簇系数和较短的平均路径, 以及节点的连接呈指数分布或幂律分布. 这一点正好与软件工程强调的“高内聚低耦合”、“软件结构通常顶层扇出较多, 中层扇出较低, 底层又高扇入到公共的实用模块中”、“在软件设计中, 结构层次不能太深, 被多个模块调用的公共模块数量不能太多”、“模块尽量调用其紧邻下层模块, 避免越级访问”等思想吻合.

根据 Garlan & Shaw 的定义: $SA = \{\text{构件}, \text{连接件}, \text{约束}\}$. 此定义认为软件体系结构超越计算过程中的算法设计和数据结构设计, 系统是由构件、连接件和约束构成. 其中, 构件(component)是一组代码

或一个独立的程序, 构件相对独立, 仅通过接口与外部相互作用; 连接件(connector)是构件的粘合剂, 把不同的构件连接起来; 约束(constrain)是构件与连接件连接时的规则, 并指明构件连接的势态和条件.

通过 SA 定义可知, SA 的复杂性与构件和连接件的连接有直接关系. 同时, 构件与连接件的连接存在方向性, 并且连接件不存在长短问题, 所以, 可通过有向无权复杂网络理论来度量 SA 的复杂性. 在建立复杂网络模型时, 构件映射为复杂网络的节点, 连接件映射为复杂网络的边^[10], 约束限定了节点与边之间的连接. 下面给出 SA 的形式化定义^[2]:

定义 1 一个三元组 $SA = (C, L; F)$ 称为一个软件体系结构, 当且仅当:

(1) $C = \{C_1, C_2, \dots, C_n\}$ 是构件的集合, $L = \{L_1, L_2, \dots, L_k\}$ 是连接件的集合, C 和 L 是有限集, 即 n, k 为任意一个有界自然数;

(2) $C \cap L = \varphi, C \cup L \neq \varphi$;

(3) $F \subseteq \{(C \times L) \cup (L \times C)\}$, 其中 $\times$ 为迪卡尔积, 表示构件与连接件之间的关系;

(4) $C, L \subseteq F, C_i \times L_j \times \dots \times L_z \times C_y \subseteq F$, 其中 i, j, z, y 为有界自然数.

3 软件体系结构复杂性度量

根据以上分析, 研究 SA 复杂性之前, 先将 SA 中的构件映射为节点, 连接件映射为边, 将 SA 转化为复杂网络模型. 这样, 对 SA 的复杂性研究就转化为对复杂网络的复杂性的研究. 复杂网络的一些重要的属性可以表示复杂网络的拓扑结构特征, 根据 SA 转换而来的复杂网络模型也具有这些性质.

由于构件与连接件及构件与构件之间的连接存在方向性, 我们将软件体系结构转化为有向复杂网络模型, 并主要通过节点的度与度分布、簇系数、平均路径长度等属性来度量 SA 的复杂性.

3.1 节点(构件)的度及度分布

节点(构件)的度用于衡量节点的重要程度. 节点的入度越大, 那这个节点所代表的构件被多个构件所调用或依赖; 节点的出度越大, 那么这个节点所代表的构件要调用或依赖多个其他的构件. 节点(构件)的入度和出度通称为节点的度.

节点(构件)的度分布用函数 $P(k)$ 表示, 表示复杂网络模型中度为 k 的节点的个数占网络节点总数的比值. 在随机网络中, 节点的度分布服从 Poisson 分布(泊松分布). 在实际 SA 结构中, 由于

SA 由人工建立设计,研究表明,实际系统的度分布服从幂律分布 $P(k) \sim k^{-\gamma}$,即无标度特性,并且 γ 的取值为(1.0,3.0)之间。

3.2 软件体系结构簇系数

簇系数表示复杂网络模型的群组结构,即,复杂网络模型中的节点按照群组分布,群组内的节点之间有密集的边相连,而群组之间的关联的边较少。按照软件工程思想,将 SA 中的构件、类等按照包的语义划分为多个组,组内的构件或类之间的联系紧密,而包之间的关系较松散(即用较少的、简单的连接件进行连接)。

如果网络中的一个节点 i 有 n_i 条边将它与其他节点相连接,这 n_i 条边连接的 n_i 个节点之间最多有 $n_i(n_i - 1)/2$ 条边。将这 n_i 个节点之间实际存在的边数 E_i 和总的可能的边数 $n_i(n_i - 1)/2$ 之比定义为节点 i 的簇系数 C_i ,即 $C_i = 2E_i / (n_i(n_i - 1))$ 。整个网络的簇系数则就是网络中所有节点的簇系数的平均值。

定义2 软件体系结构的簇系数

$$C = \frac{1}{N} \sum_{i=1}^N C_i,$$

$$\text{其中 } C_i = \frac{2E_i}{n_i(n_i - 1)}.$$

在定义中, N 为 SA 转化而来的复杂网络中节点(构件)的个数。显然, $C \in [0, 1]$, $C = 0$ 代表所有节点均为孤立节点,即没有任何连接边(连接件); $C = 1$ 代表整个网络是全连通的,而且任意两个节点都直接连接。

3.3 软件体系结构的平均路径长度

某个节点的平均路径长度是指该节点到其他所有节点的最短路径的平均,网络的平均路径长度是指网络中所有节点对之间最短路径的平均值,即任意两个节点之间要经过的边的数目的平均值。

定义3 软件体系结构的平均路径长度

$$L = \frac{1}{N(N-1)} \sum_{u \in V} \sum_{v \in V, v \neq u} d(u, v) \text{ 或}$$

$$L = \frac{1}{\frac{1}{2}N(N-1)} \sum_{u \geq v} d(u, v)$$

其中, V 表示节点(构件)集合, $N = |V|$ 为节点总数, $d(u, v)$ 表示节点 u 到节点 v 的最短距离。

平均路径长度表明了节点所代表的构件在 SA 中的中心度。节点的平均距离越短,则该节点所代表的构件与其他节点所代表的构件所需要经过的连接件越少,说明该节点所代表的构件的中心度越高。根据软件工程思想,系统结构层次不能太深,构件尽量调用其紧

邻下层构件,避免越级访问,所以,SA 所转化的有向复杂网络模型应该具有较小的平均路径长度。

4 实例分析

随着计算机技术及网络技术的发展,新疆生产建设兵团(以下简称“兵团”)开发了兵团空间信息系统,分别服务于兵团的各职能部门,为兵团屯垦戍边提供支持。本系统包括农业、水利、土地、生态环境、公共安全等 5 个子系统。

由于本系统较大,故本文仅选择水利子系统内的基于遥感(Remote Sensing, RS)和地理信息系统(Geographical Information System, GIS)的洪灾检测评估系统进行验证。该子系统主要实现在对洪灾区调查的基础上,建立洪灾遥感监测快速识别模型,实现对不同的土地利用类型其淹没情况进行实时监测,为汛情的动态监测和评估提供迅速直观及可靠的现状和变化信息。

基于 RS 和 GIS 的洪灾检测评估系统体系结构如图 1 所示。

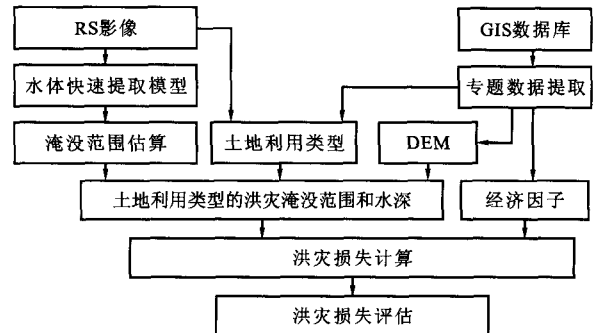


图1 系统体系结构

将图 1 的系统体系结构转换为图 2 所示的形式化体系结构图形式表达: $SA = (C, L; F)$ 。其中, 构件集合 $C = \{C_1, C_2, C_3, C_4, C_5, C_6, C_7, C_8, C_9, C_{10}, C_{11}\}$ 。由于本系统度量中没有涉及到连接件集合 L 和约束 F , 故不对 L 和 F 进行描述, 只关注构件的连接情况。

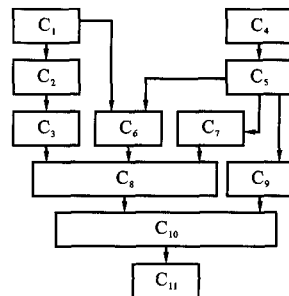


图2 形式化体系结构图

根据上节的研究,在度量系统的软件体系结构复杂度时,先把软件体系结构转化为网络图,将构件映射为节点,将连接件映射为边.将图 2 的体系结构图映射为网络图如图 3 所示.映射后的网络用二元组 $G = (V, E)$ 表示,其中节点集合 $V = \{V_1, V_2, V_3, V_4, V_5, V_6, V_7, V_8, V_9, V_{10}, V_{11}\}$. 只关注节点,不对边集 E 进行描述.

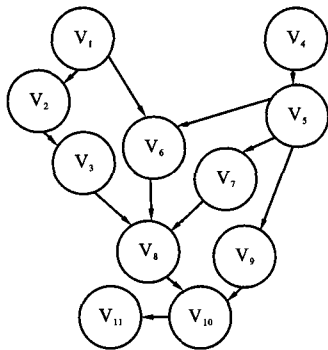


图 3 系统网络图

通过分析,图 3 中 11 个节点的度(入度与出度的和)分别为: $|V_1| = 2$, $|V_2| = 2$, $|V_3| = 2$, $|V_4| = 1$, $|V_5| = 4$, $|V_6| = 3$, $|V_7| = 2$, $|V_8| = 4$, $|V_9| = 2$, $|V_{10}| = 3$, $|V_{11}| = 1$. 同理,每个节点的入度与出度分别可得出. 研究发现,有 2 个节点的度为 4,有 2 个节点的度为 1,其余节点的度为 2 或 3. 节点的度分布 $P(k)$ 服从 Poisson 分布(钟型分布).

根据平均路径长度公式,考虑该系统网络路径的方向性因素(即有向网络),计算得到本系统的平均路径长度为 1.288. 类似地,可计算系统的簇系数和其他属性(均需考虑方向性).

通过对兵团空间信息系统水利子系统基于 RS 和 GIS 的洪灾检测评估系统进行软件体系结构复杂度度量验证,发现该子系统复杂性与一般复杂网络的特性相似. 同时,该子系统具有较合适的簇系数和较短的平均路径,系统度分布比较合理,符合软件工程开发思想.

5 结束语

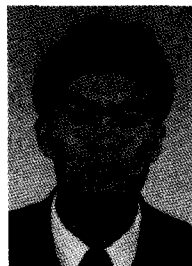
本文基于复杂网络理论和技术,将软件体系结构转化为有向复杂网络模型,并通过节点的度与度分布、簇系数、平均路径长度等属性来度量 SA 的复杂性. 最后,通过兵团空间信息系统的软件体系结构

作为实例,验证了其 SA 的复杂性. 实践表明,此方法能较好地度量所开发系统的软件体系结构的复杂度,有效地提高所开发系统的质量及可靠性.

参考文献:

- [1] 秦怀斌,梁斌,邵明文. 软件体系结构的复杂网络社团特性研究[J]. 微电子学与计算机,2012,29(4):6-9.
- [2] 张友生,李雄. 基于 Petri 网的软件体系结构可靠性分析[J]. 计算机工程与应用,2006(25):69-73.
- [3] 王树林,顾庆,陈焘,等. 基于复杂网络的大型软件系统度量[J]. 计算机科学,2009,36(2):287-290.
- [4] 韩定定. 复杂网络的拓扑、动力学行为及其实证研究[D]. 上海:华东师范大学,2008.
- [5] Concas G M. Power-laws in a large object-oriented software system[J]. IEEE Trans. on Soft. Eng. 2007,33(10):687-708.
- [6] Newman M E J. Modularity and community structure in networks[J]. Natl. Acad. Sci. USA,2006(103):8577-8582.
- [7] 李兵,马于涛,刘婧,等. 软件系统的复杂网络研究进展[J]. 力学进展,2008,38(6):805-814.
- [8] 李丽萍,缪淮扣,钱忠胜. 基于复杂网络面向对象集成测试的研究[J]. 计算机科学,2008,35(12):254-257.
- [9] 张大陆,王志晓,刘雯. 基于复杂网络的本体结构分析[J]. 同济大学学报,2009,37(2):258-261.
- [10] 郑伟才. 基于复杂网络的软件模型测度研究[D]. 南京:南京信息工程大学,2010.

作者简介:



秦怀斌 男,(1980-),博士研究生,副教授. 研究方向为软件工程、复杂网络等.



李道亮 男,(1971-),博士,教授,博士生导师. 研究方向为数字农业与农业信息化.