

基于复合信息矩阵的软件体系结构演化波及效应分析^{*}

黄翰¹ 郝志峰^{1,2} 陈明² 韩涛¹

(华南理工大学计算机科学与工程学院 广州 510640)¹

(东南大学移动通信国家重点实验室 南京 210096)²

摘要 演化性是软件的基本特性之一。软件演化由一系列复杂的变化活动组成。本文从宏观层面入手,以软件体系结构(Software Architecture, SA)作为软件的蓝图和支撑,设计了复合信息矩阵模型,作为软件体系结构演化波及效应分析的新工具。结合矩阵特性、图论知识和算法设计,新模型可以对一般 SA 静态和动态演化中的波及图表 1 效应进行理论分析和量化界定,弥补了现有研究模型需要简化连接器和无法评估连接器影响的不足,从而为 SA 演化的利用和评价提供了更为全面的依据,并为基于矩阵变化的 SA 演化计算机自动处理提供了新的理论支持。

关键词 软件体系结构,软件演化,波及效应,复合信息矩阵

Web Service Evaluation Algorithm Based on QoWS

HUANG Han¹ HAO Zhi-Feng¹ CHEN Ming² HAO Tao¹

(Dept. of Computer Science and Engineering, South China. of Tech., Guangzhou 510640)¹

(National Mobile Communications Research Laboratory, Southeast University, Nanjing 210096)²

Abstract Construction and evolution are two basic properties of software. Software evolution consists of a series of complex change activities. In the present paper, a multiple information matrix (MIM) model is designed to analyze the ripple-effect of software architecture (SA) evolution, based on macrostructure of SA. The novel model can display the characters for kinds of SA in the static and dynamic evolutionary process with the matrix and special algorithm. As a result, MIM can overcome the shortcoming of models in previous literature that the effect of connector cannot be analyzed. Therefore, SA can be evaluated with enough information to support the automatic computing with the new theory.

Keywords Software architecture (SA), Evolution, Ripple effect, Multiple information matrix

演化性与构造性是软件的两个基本特性^[1]。软件进行渐变并达到所希望的形态就是软件演化^[2]。软件演化由一系列复杂的变化活动组成,且引起变化的原因很多:基础设施的改变、功能需求的增加、高性能算法的发现和 技术环境因素变化等等。因此,对软件变化甚至演化进行理解和控制显得复杂困难。目前对软件演化的研究都流行从宏观层面入手,原因是可以避免过早陷入软件演化研究的复杂细节,可以在结构上确定每一个变化所影响的范围。

近年来,软件体系结构 SA(Software Architecture)已成为软件研究的热点之一。显然,要从宏观的角度来刻画软件演化,自然应从 SA 演化研究开始^[2]。另外,SA 是软件周期的早期产品,着重解决软件系统的结构和需求向实现平坦过渡的问题,是软件生命周期中开发、集成、测试和更改阶段的基础,加之对 SA 检测和修改的相对低代价性^[3]。这些都体现了研究 SA 演化的必要性。

在 SA 演化研究方面,形式化描述方法^[4,5]曾被用于对软件构造性的描述,而隐藏了 SA 的演化性,其缺点是不能反映 SA 演化活动的波及效应。后来,UML^[6]方法实现了对 SA 基于不同视图的描述和对 SA 结构的可视性,但未涉及 SA 演化活动中波及效应的确定方法。其他还有一些定性分析的研

究^[7-9]。Bohner^[9,10]提出了“波及效应”来描述软件变换的影响,并用可达矩阵的概念对软件变化进行了简单阐述。最近,一种基于可达矩阵的软件体系结构演化波及效应分析^[2],是对 Bohner 工作的一个发展。该研究建立了基于 SA 的构件-连接件模型,建立了 SA 关系矩阵和可达矩阵,实现了对 SA 静态和动态演化的定量分析。综上所述,以往研究成果存在以下局限:①模型只考虑了连接件关系为 1:1 的情形,并没有分析一般化的 SA 结构;②简化模型时忽略了连接件,无法量化分析连接件在 SA 演化中的影响;③只有构件贡献而无连接件贡献,得出的评价并不全面;④静态演化分析只考虑了构件的增减变化,并未考虑连接件的变化影响。

构件(Component)与连接件(Connector)被公认为 SA 的两大类构成部分,连接件的重要性在以往研究已经有所描述^[11-13]。因此,对于 SA 演化波及效应分析必须综合考虑这两方面的内容,才能得出较为全面合理的结论。本文同样从宏观层面入手,以 SA 演化为核心,建立了一种新型的模型——复合信息矩阵。该模型高度浓缩了 SA 结构中构件与连接件信息,刻画了构件之间、构件与连接件、连接件之间的关系,准确地对应一般化的 SA 结构。结合特殊算法,可以用复合信息矩阵计算构件贡献、连接件贡献和综合贡献,从而对

^{*}国家自然科学基金资助项目(60433020,10471045)、东南大学移动通信国家重点实验室开放研究基金资助课题(编号:N200605)。黄翰 博士研究生,主要研究领域为软件体系结构评估数学模型、软件研发、算法设计与分析;郝志峰 博士,教授,博士生导师,主要研究领域为代数学及其应用、组合优化与算法研究、仿生算法的数学基础研究。

与 SA 结构图边——对应。因此,命题前部分成立。

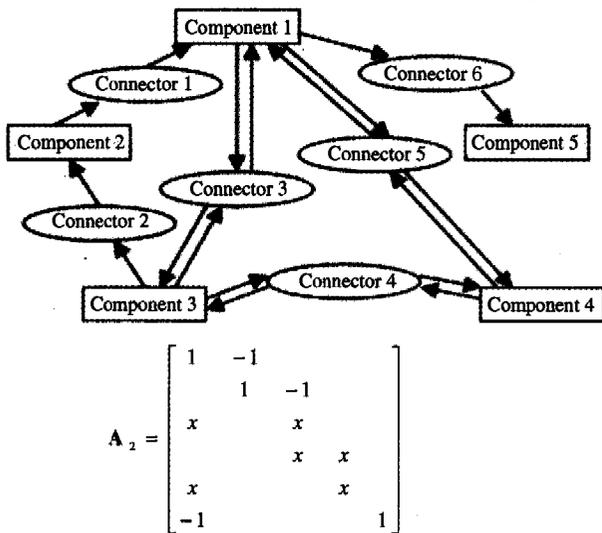


图 5 含有双向连接例子

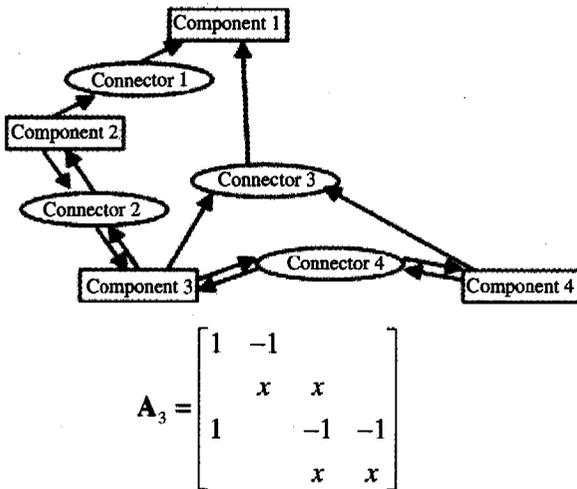


图 6 含有多对多连接件例子

因为构件和连接件之间是通过活动接口 (Active Interface) 进行通讯联系 (单向或双向); 所以, SA 结构图中结点的度数 (含出度和入度) 对应 SA 中构件/连接件的活动接口 (Active Interface) 个数。根据图论中定理: 每个图结点数度的总和等于边数的两倍^[17], 且 SA 结构图边对应复合信息矩阵中非零元素; 因此, 复合信息矩阵中非零元素个数等于活动接口 (Active Interface) 数目的一半。

命题 2 复合信息矩阵 A 对应的 SA 结构图最大度 (含出度和入度) 等于 $\max_{i=1, \dots, m; j=1, \dots, m} \{nr(i), nc(j)\}$; 其中, $nr(i)$ 为 A 第 i 行非零元素的数目, $nc(j)$ 为第 j 列非零元素的数目; 同理, 最小度 (含出度和入度) 等于 $\min_{i=1, \dots, m; j=1, \dots, m} \{nr(i), nc(j)\}$ 。

证明: 根据复合信息矩阵的定义, 复合信息矩阵第 i 行 (第 j 列) 非零元素的个数对应 SA 结构图中连接件 i (构件) 邻接点的个数, 即该点度数 (含出度和入度)。显然, 命题 2 成立。

根据命题 2, 复合信息矩阵某一行/列非零元素越多表明某个构件/连接件的邻接部件越多, 在 SA 图中的关联就越复杂。 $\max_{i=1, \dots, m; j=1, \dots, m} \{nr(i), nc(j)\}$ 对行/列揭示了枢纽型的连接件/构件 (如图 6 中 Component 3 和 Connector 3); 反之,

$\min_{i=1, \dots, m; j=1, \dots, m} \{nr(i), nc(j)\}$ 对应互连关系最简单的连接件/构件。 $nr(i)$ 与 $nc(j)$ 可以作为评价连接件/构件在 SA 图中关联重要性的指标之一。当 $\max_{i=1, \dots, m; j=1, \dots, m} \{nr(i), nc(j)\}$ 与 $\min_{i=1, \dots, m; j=1, \dots, m} \{nr(i), nr(j)\}$ 很接近时, 表明 SA 中各个部件互连关系复杂性分布均衡, 命题 3 阐述的就是这类特例。

命题 3 仅有 1 对 1 连接件 SA 结构图, 对应的复合信息矩阵任意行非零元素的个数为 2。

命题 4 若复合信息矩阵存在行满足非零个数大于 2, 则其对应的 SA 结构图中必存在 1 对多或者多对多的连接件。

证明: 根据复合信息矩阵的定义, 命题 3 和 4 易得证。

命题 3 和命题 4 表明, 可以从复合信息矩阵得出 SA 中连接件的类型。连接件承担了实现部件间信息和行为关联的作用, 是系统复杂性的来源, 对系统各种性能有着重要的影响。可以结合以上的命题, 描述复合信息矩阵所包含的 SA 复杂性信息。

3 SA 演化中的波及效应分析

SA 的静态演化是指在系统非运行状态下, 软件功能的变更和环境因素的变化等对组成 SA 的构件进行的增加、替换、删除, 但这种变化蕴含着波及效应, 更多地表现连接件的变化, 包括接口变化和其联系的构件的重新组合。本文给出基于复合信息矩阵的构件和连接件可达域, 以刻画构件和连接件的波及范围; 同时给出了可达域的求解算法, 以及构件贡献^[2]、连接件贡献和综合贡献的计算方法, 用以确定构件和连接件对 SA 的影响。

3.1 构件/连接件可达域

定义 2 (构件可达域) 在 SA 结构图中, 与构件/连接件 a 连通的构件组成的集合 W 为 a 的构件可达域, 记为 $W(a)$ 。

例如: 在图 4 中, 构件 4 的构件可达域 $W(\text{Component } 4) = \{\text{Component } 1, \text{Component } 5\}$ 。

定义 3 (连接件可达域) 在 SA 结构图中, 与连接件/连接件 a 连通的连接件组成的集合 Q 为 a 连接件可达域, 记为 $Q(a)$ 。

例如: 在图 4 中, 构件 4 的连接件可达域 $Q(\text{Component } 4) = \{\text{Connector } 1, \text{Connector } 5\}$ 。

定义 4 (可达域) 连接件/连接件 a 的可达域 $S(a)$ 为 a 的构件可达域和连接件可达域的并集, 即 $S(a) = W(a) \cup Q(a)$ 。

可达域描述的是构件/连接件变化所影响的其他构件和连接件, 当某一组构件/连接件发生变化时, 可以确定被影响或者波及的范围。

以下给出一种算法, 当给出 $n \times m$ 复合信息矩阵 A, 就可求得某个构件/连接件 a_0 的构件可达域 $W(a_0)$ 和连接件可达域 $Q(a_0)$ 。算法框架为:

步骤 1 $W(a_0) = \emptyset, Q(a_0) = \emptyset$, 清空临时栈 RS, $t = a_0$ 。

步骤 2 如果 t 是代表构件, 则当前列 $c = t$ 对应的列号, 转 3; 否则, 当前行 $r = t$ 对应的行号, 转 4。

步骤 3 如果 $a_k = -1$ 或 $a_k = x$, 且 Connector i $\notin Q(a_0)$, 则 $Q(t) = Q(t) \cup \{\text{Connector } i\}$, Connector i 入栈 RS ($i = 1, \dots, n$)。转 5。

步骤 4 如果 $a_r = 1$ 或 $a_r = x$, 且 Component j $\notin W(a_0)$, 则 $W(t) = W(t) \cup \{\text{Component } j\}$, Component j 入栈 RS ($j = 1, \dots, m$)。转 5。

可将其加入任务流图模型库中以供使用。

2.2 重用任务流图模型模板

下面以带有报警功能的农业环境无线远程监控系统设计为背景,简要说明重用任务流图模型模板的方法。图3是带有报警功能的农业环境无线远程监控系统的功能结构示意图,与

上面定义的普通农业环境无线远程监控系统任务流图模型模板接口相比,增加了对报警功能的支持。因此,在进行系统任务流图建模时,可以对前面建立的任务流图模型模板进行重用,并在其中增加报警任务。这样,在系统建模过程中,主要工作就是添加实现报警任务,从而提高建模效率与质量。

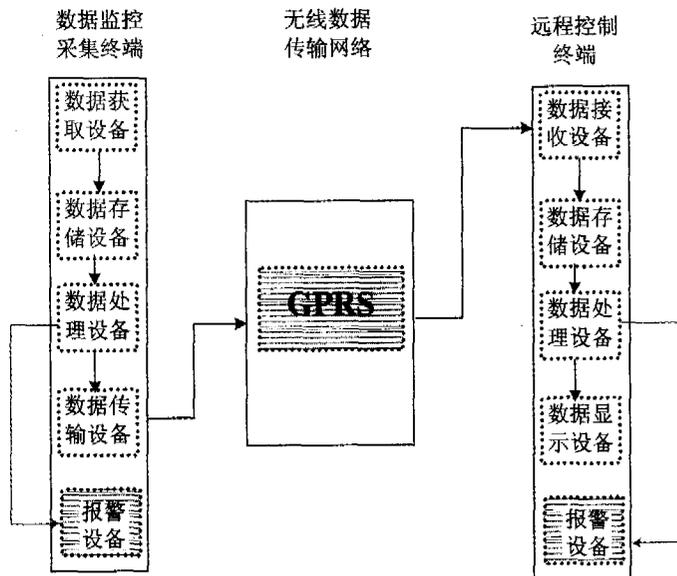


图3 任务流图模型模板的重用过程

2.3 验证任务流图模型模板

在GPRS行业应用系统建模完成之后,首先进行映射,将农业环境无线远程监控系统任务流图模型中的每个任务映射到相应执行部件,得到与农业环境无线远程监控系统功能相对应的执行流。

接下来,接入计算机模拟的输入信号,运行农业环境无线远程监控系统功能相对应的执行流,得到执行结果,将系统的输出与输入信号比较,看其功能是否正确。如果功能不正确,则需对系统模型进行调试;功能正确,则统计性能参数对系统模型进行仿真,得到农业环境无线远程监控系统的性能参数。

结论 本文提出了基于平台进行GPRS行业应用开发的思路。首先,对现有的GPRS行业应用开发方法进行分析,针对现存问题提出了基于平台的GPRS行业应用开发方法;然后,详细研究了基于平台的开发方法;最后,以农业环境无线远程监控系统为例子,进一步阐述了基于平台的GPRS行业

应用开发方法。基于平台的农业环境无线远程监控系统开发实例表明,这种GPRS行业应用开发技术保证了开发正确性,增强了通用性,缩短了开发周期,取得了良好效果。

(注:本文是山东省重点技改科研课题“DYX-1型电表表遥测遥控系统”的后期阶段性研究成果。)

参考文献

- 徐济仁,崔剑,董红星. GPRS的技术与应用. 四川通信技术, 2001(5)
- 杨菁,余成波,胡晓倩. GPRS技术及其应用探析. 重庆工学院学报, 2004(1)
- 吴淑健. GPRS技术及其行业应用. 移动通信, 2003(10)
- 蔡锐丹,许少云,甘义成. GPRS无线数据传输系统的设计与应用. 电子质量, 2004(1)
- 吴红云. 移动通信前沿——GPRS技术应用. 中山大学学报(自然科学版), 2003(S2)
- 郑家莉,黄炜. 无线远程监控系统的核心技术研究. 单片机与嵌入式系统应用, 2004(6)

(上接第263页)

- Wang Y H, Zhang S K, Liu Y, et al. Ripple-Effect analysis of software architecture evolution based on reach ability matrix. Journal of Software, 2004, 15(8): 1107~1115(in Chinese with English Abstract)
- Bass L, Clements P C, Kazman R. Software Architecture in Practice. Aonon; Addison-Wesley, 1998
- Medvidovic N, Taylor R N. A classification and comparison framework for software architecture description languages. IEEE Trans on Software Engineering, 2000, 26(1): 70~93
- Luo H J, Tang Z S, Zheng J D. Visual Architecture Description Language XYZ/ADL. Journal of Software, 2000, 11(8): 1024~1029 (in Chinese with English Abstract)
- Rational Rose Corporation. UML notation guide. 2003. http://www.rational.com/uml
- Sun C A, Jin M Z, Liu C. Overviews on software architecture research. Journal of Software, 2002, 13(7): 1228~1237
- Bohner S A. Impact analysis in the software change process: A year 2000 perspective. In: Proc. of the Int'l Conf on Software Maintenance (ICSM'96). Washington; IEEE, 1996. 42~51
- Ryder B G, Tip F. Change impact analysis for object-oriented programs. In: Proc. of 2001 ACM SIGPLAN-SIGSOFT Workshop on Program Analysis for Software Tools and Engineering.

- New York; ACM Press, 2001. 46~53
- Bohner S A. Software change impacts: An evolving perspective. In: Proc. of the Int'l Conf of Software Maintenance (ICSM 2002). Washington; IEEE, 2002. 263~272
- Jansen A, Bosch J. Evaluation of tool support for architectural evolution. In: Automated Software Engineering, 2004. Proceedings. 19th International Conference on 2004. 375~378
- Gomaa H, Hussein M. Software reconfiguration patterns for dynamic evolution of software architecture. In: Software Architecture, 2004. WICSA 2004. Proceedings. Fourth Working IEEE/IFIP. June 2004. 79~88
- Tingting H, Taolue C, Jian L. Structure Analysis for Dynamic Software Architecture Based on Spatial Logic. In: Computer Software and Applications Conference, 2005. COMPSAC 2005. 29th Annual International Vol1, July 2005. 71~76
- Mei H, Chen F, Feng YD, et al. ABC: An architecture based, component oriented approach to software development. Journal of Software, (in Chinese with English Abstract), 2003, 14(4): 721~732
- Garlan D, Shaw M. An introduction to software architecture. In: Ambriola V, Tortora G, eds. Advances in Software Engineering and Knowledge Engineering, Vol II. Hackensack; World Scientific Publishing, Co, 1993