

# 一种通过量化分析实现非功能需求的方法

刘小丽, 毋国庆, 江敏

(武汉大学计算机学院, 武汉 430072)

**摘要:** 提供了一种面向过程非功能需求处理方法, 通过量化需求领域信息实现; 对非功能需求目标进行分析得到目标优化树, 在目标优化树中加入领域信息产生设计分析图, 然后对图中领域信息结点进行量化赋以特定的量值, 权衡各量值来选择合适的解决方案, 还提供了简便的方法来支持系统升级。

**关键词:** 非功能需求; 量化; 目标优化; 领域信息

## Quantifying Method for Dealing with Non-functional Requirements

LIU Xiaoli, WU Guoqing, JIANG Min

(School of Computer, Wuhan University, Wuhan 430072)

**【Abstract】** The quantifying method proposed is process-oriented, it manages non-functional requirements using quantifying the domain information. A non-functional goal refinement tree is generated during the goal refinement process, and then the domain information is incorporated into the tree. After quantifying the domain node, the best solution is selected through the analysis of the value of many different domain nodes. This method also supports system change.

**【Key words】** Non-functional requirements; Quantifying; Goal-refinement; Domain information

非功能需求指对系统提供的服务或功能的约束, 包括时间约束、开发过程约束、标准等, 分为可用性、可靠性、安全性、标准、可移植性等。非功能需求关心的是整个系统的特性, 比功能需求更关键, 一个功能需求没有实现可能降低系统的可用性, 但一个非功能需求没有满足, 就会导致系统无法使用<sup>[5]</sup>。近年来出现了很多非功能需求的处理方法, 但往往由于模糊的需求表示或分析造成了在需求优化的时候出现了不合理的需求<sup>[3]</sup>。本文给出了一种面向过程的量化非功能需求的方法, 通过精确分析领域信息和非功能需求获得合理的设计决策, 该方法也描述了如何根据需求的变动修改系统的设计。

### 1 量化非功能需求的方法

本方法涵盖了一系列的分析设计过程, 通过领域分析和目标抽取获取必要的领域信息和非功能需求, 对非功能需求按面向目标方法优化分解得到目标优化树, 在目标优化树中加入领域信息得到设计分析图, 然后对特定的领域信息进行量化获取合理的设计决策。

#### 1.1 获取领域信息以及非功能需求

这里所要获取的领域信息指应用领域和与其相关的执行领域、体系结构领域和算法领域等<sup>[2,3]</sup>。领域信息是用于约束非功能需求的, 各个目标需要在领域中权衡分析来进行折中处理。例如, 要设计一个安全性较高的IP电话系统, 要保证人所能承受的时间延迟就要考虑如下信息领域: 语音信号在什么样的网络中传输、语音信号加密的时候使用的是什么样的加密算法、网络的传播时延等; 高性能的加密算法处理时间就长, 这样处理时间和算法可靠性就可能在领域中发生冲突, 因此对领域信息进行量化分析是一个很好的解决方案。

获取领域信息的同时也要获得非功能需求, 之后就要对

需求和领域信息进行分析, 选择合适的系统解决方案。

#### 1.2 选择合适的解决方案

对上一步获取的非功能需求, 进行面向目标分析, 得到一棵关于非功能需求的目标优化树<sup>[1]</sup>, 父子结点间是“与”关联。分析树中的非功能需求获取相关的领域信息, 将非功能需求和与其相关的所有领域关联建立设计分析图。

设计分析图中的主要结点和关联关系如图1, 领域结点和量化结点就是用于加入领域信息的。领域结点存放特定的领域信息, 量化结点存放目标结点与多个领域结点关联时的量化值, 说明性目标标出目标对领域的限制, 量值结点给出了量化结点或者领域结点的取值大小。量化结点的值由多个领域结点的值决定, 有4种取值方法: (1) +: 领域的值是各个领域结点值的和; (2) min: 取各个领域结点的最小值; (3) max: 取最大值; (4) or: 按照某种方法选择其中一个结点的值。

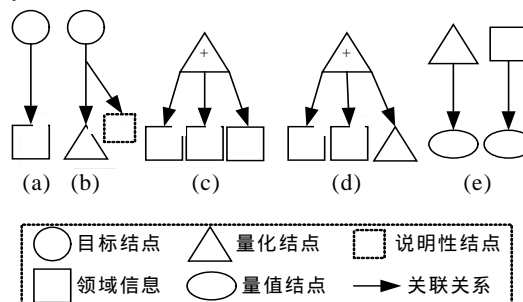


图1 结点之间的关联关系

**作者简介:** 刘小丽(1981—), 女, 硕士生, 主研方向: 软件工程, 软件基础理论; 毋国庆, 教授、博导; 江敏, 博士

**收稿日期:** 2005-11-22 **E-mail:** liuliu\_99@126.com

该方法最关键的一步就是对设计分析图进行量化分析产生设计决策,为图中的领域结点确定某一方面的数量值,根据这些量值选择合适的执行平台、适当的算法等。从叶子目标开始分析,如果一个目标实现了,就把设计图中的目标结点标记为“”,即满意目标;如果一个目标的所有子结点都被标记为“”,就把它也标记为满意目标,如此依次对目标结点处理,直到图中所有的目标都被标记就完成了整个设计决策过程。

### 1.3 支持系统改变

系统的需求或者领域信息发生改变时,首先要对目标分析图进行修改,通过加入一些新的标记来实现<sup>[4]</sup>,若某结点发生改变或者新添加/删除某结点,则从该结点出发到起对应根结点的路径上的所有结点都要重新考虑,可能会添加或者删除一些关联关系,分析图完成后重新对未处理的结点进行处理得到新的设计决策。

## 2 实例

该实例是一个安全的 IP 电话系统,此 IP 电话可能在局域网或者 WAN 中使用,要求的安全性较高需要对语音信号进行加密,同时要保证语音的传播时延在人能承受的范围内。

### 2.1 基本需求分析

这里只对系统的部分目标进行分析。该系统中语音信号的时间延迟不能超过 150ms(这是人能够承受的时间延迟),信号传播过程中以及在加密的时候都有时间延迟,因此要选择合适的加密算法、计算平台和网络环境等。

根据需求信息生成目标优化树,如图 2(a)中的目标结点:总的非功能目标(结点  $g_0$ )分为 2 种类型:性能目标( $g_1$ )和安全性目标( $g_2$ ),对于性能又分为:带宽和时间延迟( $g_4$ ),安全性目标分为保密性( $g_5$ )、完整性和可用性,保密性通过加密来实现。

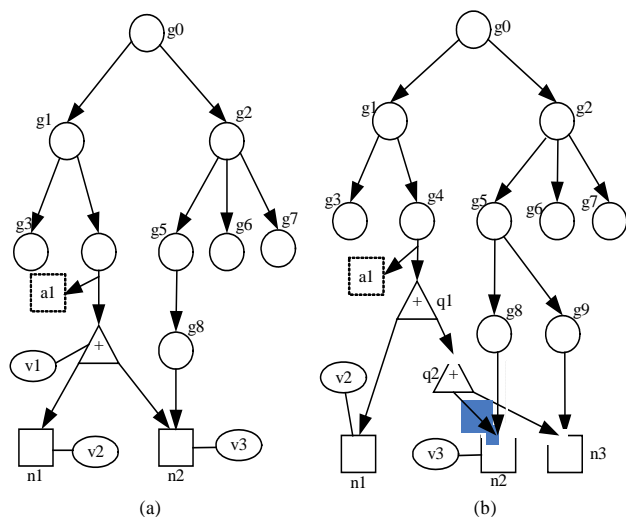


图 2 IP 电话系统设计分析图

这里只考虑时间延迟和安全性这两个目标。时间延迟在系统中表现为传播时间和加密时间两部分,就加入与加密相关的加密算法领域结点( $n_2$ )和传播时延结点( $n_1$ ),因此引入量化结点  $q_1$  与传播时间延迟和加密两个领域结点关联( $q_1$  的值等于  $n_1$  和  $n_2$  值的和);语音信号的时间延迟不超过 150ms 就加入说明性结点  $a_1$ 。这样就得到了部分目标的设计分析图。

### 2.2 获取设计决策

选择合适的加密算法、执行平台、网络平台等具体的执

行领域信息使分析图中的各个目标都得以实现,这是非功能需求分析的关键。算法的执行时间和传播时间延迟的和不大 于 150ms。该系统的传输网络可能是局域网也可能是万维网,万维网的传播时延 75ms,局域网的传播时延 1ms;可选择的算法也有很多:单 DES、三 DES 和 AES 等,执行平台的硬件配置也都不同,具体量值也不同<sup>[4]</sup>。

该系统应用领域可有很多种不同的配置,选择其中一种做一介绍:加密算法选择 PIII 系统的 AES,是一种比较高安全级别的加密算法符合机密性这个目标,加密时间是 2ms,传输网络是万维网,时间延迟是 75ms,总的时间延迟不超过 150ms。这样通过对时间的量化就很准确地选取合理的领域信息。不同平台下的算法与认证所用时间见表 1。

表 1 不同平台下的算法与认证所用的时间

| 算法         | 加密算法(b)  |            |           | 认证  |      |     |
|------------|----------|------------|-----------|-----|------|-----|
|            | DES (64) | 3DES (196) | AES (256) | 口 令 | 求摘 要 | 签 名 |
| PIII       | 2        | 4          | 2         | 30  | 28   | 38  |
| HP Jornada | 16       | 31         | 17        | 284 | 313  | 348 |

处理时间(ms)

### 2.3 根据需求的改变进行设计改变

在系统的使用过程中可能出现需求的改变或者应用领域的改变;比如在该 IP 电话系统中,要增加一个认证功能来提高 IP 电话的保密性,就要对系统的设计进行修改。要对分析图的部分结点重新分析如图 2(b)。

首先在目标优化树中要增加认证目标( $g_9$ ),加入到安全目标中,当然对应也要添加一个认证领域信息结点( $n_3$ )。增加了认证功能就需要增加系统的处理时间,也就是语音信号的总时间延迟总要增加,总时延分为传播时延和处理时延,为了简便起见加入了一个量化结点( $q_2$ )作为处理时间结点, $q_2$  的值是  $n_1$  和  $n_3$  的值的和,仍然使用“+”连接。处理时间包括加密和认证两部分;添加新结点和关联关系如图 2(b)。新添加或者改变的结点都用黑体标出。

设计图分析完成后就要进行系统设计,从新添加的结点( $n_3$ )往上到根结点的所有路径上的所有结点都重新分析,在图中用粗箭头标识的路径就是需要重新分析的路径,其上的结点的量值都要删除,比如图中  $q_1$  对应的  $v_1$  结点就被删除了,路径上的目标结点的满意标记同时也要被删除。如果要删除结点就把被删结点的父结点标记为新结点;修改结点时,结点本身就看作新结点。

通过目标优化树能很清楚地看到系统所有的非功能需求预计与它们相关的领域,改变的需求也容易在图中显现处理,需求改变时只需对相关领域做适当的调整,整个设计和修改过程很简洁。

## 3 结论

本文给出了一种面向过程的方法来处理非功能需求,通过对领域信息进行量化分析来选择满足非功能需求的决策方案。提供了支持需求改变的机制,描述了一个完整的非功能需求的处理的过程。但是该方法能否处理所有类型的需求以及在系统非常复杂的时候如何处理这些需求,都是我们要考虑的问题;并且也需要寻找一些合适的工具来支持该方法。

(下转第 48 页)