

# SPC 在软件过程度量中的应用及改进

杜庆峰, 马慧璐

(同济大学软件学院, 上海 201804)

**摘要:** 在介绍软件过程度量原理的基础上, 讨论 Shewhart 控制图的构成和分析方法。结合实例, 分析统计过程控制在软件过程度量中的作用。针对传统 Shewhart 控制图无法区分软件过程之间影响的缺陷, 借助选控图理论对现有方法在软件过程度量中的不足提出改进。有效区分软件过程的相互作用, 定性和定量地分析软件过程的稳定性和性能。

**关键词:** 软件过程度量; 统计过程控制; 选控图

## Application and Improvement of Statistical Process Control in Software Process Metrics

DU Qing-feng, MA Hui-jun

(School of Software Engineering, Tongji University, Shanghai 201804)

**【Abstract】** This paper introduces the theory of software process metrics as well as the structure and analyzes method of Shewhart control charts. It discusses the role Statistical Process Control(SPC) plays in software process metrics using an example. As traditional Shewhart control charts can not distinguish the causes from other processes, an improvement is proposed with the help of select cause control charts, which can conquer the deficiencies in the field of software process metrics. It can analyze the stability and performance characteristics of software process both qualitatively and quantitatively.

**【Key words】** software process metrics; Statistical Process Control(SPC); select cause control charts

统计过程控制<sup>[1]</sup>技术是一种广泛用于制造业的过程质量控制技术。它利用统计技术将数据转换成信息, 运用过程控制图来检测产品、过程、资源在生产过程中的质量特性, 确保过程处于统计控制之下, 为过程控制和改善提供量化基线。目前, 利用统计过程控制(Statistical Process Control, SPC)原理控制软件过程质量, 是软件界正在探讨的热门话题。其研究主要集中于如何应用传统的 Shewhart 控制图来控制软件开发过程的稳定性和过程能力<sup>[2]</sup>。本文结合实例, 对 SPC 在软件过程度量中的应用及改进作了探讨。

### 1 统计过程控制(SPC)

#### 1.1 SPC 基本原理

统计过程控制的理论基础是统计学中的中心极限定律和  $3\sigma$  原理。当过程仅受相互独立的随机因素影响时, 子样本均值将随容量的增大而近似服从正态分布, 即  $\lim_{n \rightarrow \infty} \bar{x}_i \sim N(\mu, \sigma^2)$ 。由  $3\sigma$  原理,  $P\{\mu - 3\sigma < \bar{x}_i < \mu + 3\sigma\} = 99.73\%$ , 即  $\bar{x}_i$  落在  $3\sigma$  范围内的概率是 99.73%。在有限次的观测内, 其值几乎全部落入  $3\sigma$  范围内, 过程仅受随机因素影响, 处于“统计控制状态”; 否则认为过程受到可归属原因影响, 处于“失控状态”。由于过程波动具有统计规律性, 当过程受控时, 过程特征服从稳定的随机分布; 而失控时, 过程分布将发生改变。SPC 正是应用过程波动的统计规律对过程进行分析控制的。

#### 1.2 控制图的原理与构造

SPC 利用 Shewhart 控制图来记录过程质量, 分析过程质量是否产生异常。Shewhart 控制图以中心线表示测量值的均值  $\mu$ , 以上下控制线  $\pm 3\sigma$  (为标准差) 表示测量值的正常偏差。

当测量值超出控制线或出现某些非随机分布时, 表明过程出现异常。用  $3\sigma$  作为界限避免了对基本的自然偏差分布进行假设的需要, 同时  $3\sigma$  界限对于异常偏差非常敏感而且很少出现错误的警报。

控制上限  $UCL = \mu + 3\sigma$ , 控制下限  $LCL = \mu - 3\sigma$ , 控制中心  $CL = \mu$ , 如图 1 所示。

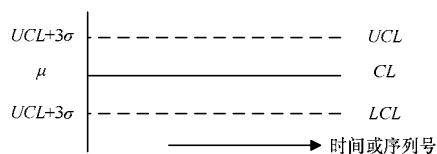


图 1 SPC 控制图结构

#### 1.3 XmR 控制图

XmR 图(单值点-移动极差图)是 SPC 控制图之一, 由 X 图和 mR 图构成。它反映了单值点度量的平均值及偏差的波动。假设样本提供  $k$  个连续度量值, 则

$$\bar{X} = \frac{1}{k} \sum_{i=1}^k X_i \quad (1)$$

第  $i$  个移动极差:

$$mR_i = |X_{i+1} - X_i| \quad (1 \leq i \leq k-1) \quad (2)$$

平均移动极差:

$$\overline{mR} = \frac{1}{r} \sum_{i=1}^r mR_i \quad (3)$$

**作者简介:** 杜庆峰(1968 - ), 男, 高级工程师, 主研方向: 软件项目管理, 软件测试; 马慧璐, 硕士研究生

**收稿日期:** 2009-07-08 **E-mail:** du\_cloud@mail.tongji.edu.cn

X 图控制限：

$$\text{控制上限 } UCL = \bar{X} + 2.660\overline{mR} \quad (4)$$

$$\text{中线 } CL = \bar{X} \quad (5)$$

$$\text{控制下限 } LCL = \bar{X} - 2.660\overline{mR} \quad (6)$$

mR 图控制限：

$$\text{控制上限 } UCL = 3.27\overline{mR} \quad (7)$$

$$\text{中线 } CL = \overline{mR} \quad (8)$$

$$\text{控制下限 } LCL = 0 \quad (9)$$

## 2 案例分析

例：某项目过程管理中，进度偏差如表 1 所示，用 SPC 的 XmR 控制图实现对数据的统计分析 & 控制。

表 1 项目各阶段的进度偏差

阶段	计划持续时间	计划结束时间	实际结束时间	进度偏差/(%)
计划	7	2008-02-01	2008-02-01	0.00
需求分析	20	2008-02-20	2008-02-21	5.00
设计	24	2008-03-15	2008-03-17	8.30
编码	100	2008-06-23	2008-06-30	7.00
单元测试	18	2008-07-11	2008-07-14	16.67
集成测试	16	2008-07-27	2008-07-30	18.75
系统测试	20	2008-08-16	2008-08-19	15.00

由式(2)得移动极差值，如表 2 所示。

表 2 移动极差值

阶段	mR/(%)
计划	-
需求分析	5.00
设计	3.30
编码	1.30
单元测试	9.67
集成测试	2.08
系统测试	3.75

由式(1)得  $\bar{X} = 10.1\%$ ；由式(2)、式(3)得  $\overline{mR} = 4.18\%$ ；由式(4)~式(6)得 X 图控制限  $UCL = 21.22\%$ ， $CL = 10.1\%$ ， $LCL = -1.02\%$ ；由式(7)~式(9)得 mR 图控制限  $UCL = 13.67\%$ ， $CL = 4.18\%$ ， $LCL = 0$ 。

由表 1 及 X 控制图的控制限绘制如图 2 所示。

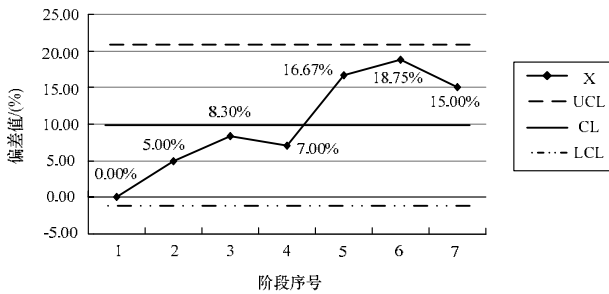


图 2 X 图

由表 2 及 mR 控制限绘制如图 3 所示。

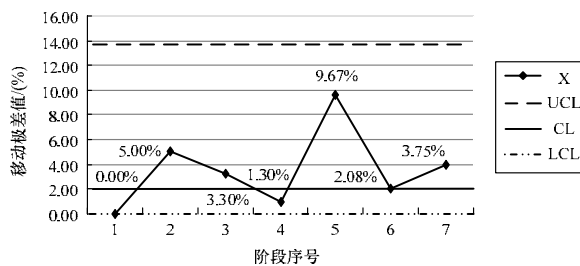


图 3 mR 图

根据 Western Electric Handbook 提出的控制图失控状态

判定条件<sup>[3]</sup>，控制图中没有符合判定条件的点，该过程稳定。

## 3 改进

通过上述案例分析，知该项目在单元测试及集成测试处进度出现异常，过程不稳定。但分析表 1 中各阶段项目进度偏差易发现，编码阶段出现很大的进度偏差，项目进度延迟，导致后续的单元测试与集成测试进度偏差出现大的波动。而单元测试与集成测试阶段的实际持续时间并未超出计划持续时间。

从而可见，在复杂的软件开发全过程管理过程中，Shewhart 控制图只能用于评判软件开发的总体性能，其中涉及到各种子过程的相互作用<sup>[4-5]</sup>，而不适合度量和分析一个特定的软件过程。

一个软件项目的开发活动涉及到很多子过程，每个子过程实现软件产品的一个工序，这些过程是相互关联的。传统的 Shewhart 控制图是全控图，只能区分偶然因素(软件过程内在的、难以消除的过程因素)和系统因素(软件过程外在的、应该及时消除的过程因素)，无法把握和处理过程之间的相互作用。在过程控制中要合理分辨异常是由该目标过程引起的还是前序过程引起的，即“目标过程因素”和“前序过程因素”，需要借助文献[5]提出的选控图。

由于后序过程要受到前序过程的影响，因此过程控制的基本单元应由相邻的 2 个过程组成，如图 4 所示(图中以后序过程作为目标过程进行诊断)。图中前序过程与目标过程的接口处是前序过程的总质量。由于受到前序过程总质量的影响，因此目标过程存在 2 种质量，即总质量与分质量。总质量由 Shewhart 控制图度量，分质量由选控图度量，这样得到了三图诊断系统，它是过程控制的基本单元。表 3 分析了 8 种状态的诊断。

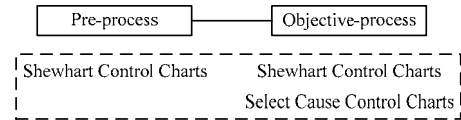


图 4 2 种控制图

表 3 基于 2 种控制图的过程控制分析

前序过程控制图	目标过程控制图	目标过程选控图	分析				补充
			目标过程过程	目标过程系统因素	前序过程过程	前序过程系统因素	
异常	异常	异常	异常	存在	异常	存在	-
异常	异常	正常	正常	无	异常	存在	-
异常	正常	异常	异常	存在	异常	存在	两者作用相反而抵消，目标过程控制图正常
异常	正常	正常	正常	无	异常	存在	两者作用相反而抵消，目标过程正常
正常	异常	异常	异常	存在	正常	无	-
正常	异常	正常	正常	无	正常	无	两者作用方向相同而叠加，目标过程的控制图异常
正常	正常	异常	异常	存在	正常	无	两者作用相反而抵消，目标过程的控制图正常
正常	正常	正常	正常	无	正常	无	过程稳定

## 4 结束语

SPC 为软件过程的监控提供了诊断依据，使得过程管理工作目标清晰，过程控制高效有序。借助三图诊断系统可以弥补传统 Shewhart 控制图的不足，区分和定义过程间的相互作用。和选控图的结合有效地提高了 SPC 在软件度量的应用效果。

(下转第 107 页)