

# 基于云模型的程序时间性能评价方法

刘 玮<sup>1</sup>, 李广力<sup>2</sup>, 王冠成<sup>2</sup>, 崔天明<sup>1+</sup>

(1. 长春大学 旅游学院 基础部, 吉林 长春 130607;

2. 吉林大学 计算机科学与技术学院, 吉林 长春 130012)

**摘 要:** 提出一种基于云模型的程序时间性能评价方法。利用云模型在不确定性处理中的优势, 对程序时间性能进行定量评价与定性评价之间的转化, 改善传统的程序时间性能评价方法在复杂运行环境中程序规模较大时存在的不足, 能够很好地适用于各类程序的程序时间性能评价。使用 LeNet 和 Sharpness 等程序进行测试和实例验证, 实验结果表明, 所提方法充分利用了云模型的特性, 能够高效、准确地对程序的程序时间性能进行评价。

**关键词:** 云模型; 不确定性; 时间性能评价; 时间性能比较; 程序测试

**中图分类号:** TP302 **文献标识码:** A **文章编号:** 1000-7024 (2018) 02-0435-06

**doi:** 10.16208/j.issn1000-7024.2018.02.025

## Program running time performance evaluation method based on cloud model

LIU Wei<sup>1</sup>, LI Guang-li<sup>2</sup>, WANG Guan-cheng<sup>2</sup>, CUI Tian-ming<sup>1+</sup>

(1. Foundation Department, Tourism College, Changchun University, Changchun 130607, China;

2. College of Computer Science and Technology, Jilin University, Changchun 130012, China)

**Abstract:** A method for evaluating program running time performance based on cloud model was proposed. The superiorities of cloud model processing uncertainty were utilized to complete a conversion between quantitative and qualitative evaluation and disadvantages of the traditional program timing evaluation methods were made up, especially when running environments were complicated and the scales of the programs were enormous. The method proposed can be well applied to timing performance evaluation of any programs. The method was empirically validated with programs like LeNet and Sharpness. Experimental results show that the proposed method based on cloud model takes full advantage of cloud model. It can evaluate program timing performance effectively and accurately.

**Key words:** cloud model; uncertainty; timing performance evaluation; timing performance comparison; program test

## 0 引 言

程序时间性能评价是软件测试中非常重要的一个组成部分, 对程序时间性能的评测有助于进一步提高软件产品的质量。在程序执行过程中, 运行环境往往会受到多方面因素的影响, 导致程序的执行时间具有不确定性。同时, 对于大型程序也很难采用多次执行计算平均值的方式来合理衡量程序的运行时间。云模型<sup>[1,2]</sup>是一种度量不确定性的定量-定性转换模型, 近年来被广泛地应用在系统信任评估<sup>[3]</sup>、服务选择<sup>[4]</sup>、不确定性语言和数值之间的转换<sup>[5]</sup>、

群体决策<sup>[6]</sup>、系统评价<sup>[7]</sup>、聚类分析<sup>[8]</sup>等诸多领域, 利用云模型来对计算机程序时间性能进行评价是解决程序时间性能不确定性的一种新方法。

## 1 云模型理论

云模型理论<sup>[9,10]</sup>融合了模糊集和概率论的观点, 能够很好地描述自然界中各种概念的不确定性, 同时也体现了模糊性和随机性之间存在的关联。云模型是由李德毅院士在结合了概率论和模糊数学理论的基础上, 创立的一种通过特定结构算法构成的定性与定量概念转换的模型。云模

收稿日期: 2016-12-06; 修订日期: 2017-02-24

基金项目: 吉林省教育厅“十二五”科学技术研究基金项目(吉教科文合字【2015】582); 吉林省教育厅“十三五”科学技术研究基金项目(吉教科文合字【2016】534)

作者简介: 刘玮(1982-), 女, 吉林长春人, 硕士, 讲师, CCF 会员, 研究方向为程序设计语言; 李广力(1992-), 男, 吉林长春人, 硕士研究生, CCF 会员, 研究方向为云计算与并行计算; 王冠成(1993-), 男, 江苏连云港人, 硕士研究生, CCF 会员, 研究方向为编译优化; +通讯作者: 崔天明(1979-), 女, 黑龙江北安人, 硕士, 副教授, 研究方向为程序设计语言。E-mail: liuwei-198202@126.com

型能够对自然界概念的不确定性进行描述,通过随机性与模糊性的关联,构成了概念在定性和定量之间的相互映射。

### 1.1 云的定义

“云”或“云滴”是云模型中的基本单元。“云”是指其在论域  $U$  上的一个分布,用联合概率  $(x, \mu)$  来表示。 $x$  是论域中的一个定量值,  $\mu$  表示该值的隶属度或确定度,用来对某种倾向的稳定程度进行度量。

### 1.2 云的数字特征

在云模型中,用 3 个数据来对其特征进行表示<sup>[1]</sup>。

**期望:**期望是云在论域空间分布的期望,它是概念在整个论域空间中的中间值,是该概念中最具有代表性的一个样本,一般使用符号  $Ex$  表示。

**熵:**熵用来表示概念的不确定程度,由模糊性和随机性共同决定。随机性的程度反映了云的离散程度,模糊性的程度反映了概念能够接受的云取值的范围。熵一般使用符号  $En$  表示。

**超熵:**超熵用来对熵的不确定性进行度量,它是熵的熵,体现了一个概念在论域空间中不确定程度的凝聚性,间接反映了云的厚度。超熵由熵的模糊性和随机性共同决定,一般使用符号  $He$  表示。

### 1.3 云发生器

云发生器分为正向云发生器(CG)和逆向云发生器( $CG^{-1}$ )两种,分别用来生成云滴和计算数字特征。正向云发生器根据云的 3 个数字特征(期望、熵、超熵)不断产生新的云滴,它实现了概念从定性表示到一定范围和分布的定量表示的过程。逆向云发生器是根据定量表示计算云模型的数字特征,从而转换为定性表示的过程。

## 2 基于云模型的程序时间性能评价方法

程序时间性能评价在软件测试过程中是不可或缺的一个组成部分,通常需要获得较为准确的程序运行时间以便进一步对程序的性能进行评判。然而,由于计算机操作系统环境的复杂性,程序的运行时间往往具有不确定性,导致无法对程序时间性能进行准确的评判。针对以上问题,本文利用云模型完成程序运行时间从定量到定性评价的转换,设计了一种基于云模型的程序时间性能评价方法。接下来,本文首先对当前程序时间性能评价中普遍存在的问题进行分析和讨论,之后对基于云模型的程序时间性能评价方法进行阐述。

### 2.1 当前程序时间性能评价存在的问题

目前,对于程序时间性能的评价方法往往都是通过多次执行程序进行平均值计算从而获得时间性能指标,之后利用指标进行分析,从而完成程序时间性能的评价。一方面,对于大型程序而言,运行一次程序的所需的时间较长,若多次执行,会导致测试的耗时过长;另一方面,由于计算机操作系统的复杂性,会导致多次执行程序时,得到的

时间性能指标具有不确定性。

#### 2.1.1 大型程序测试耗时过长

在对大型程序进行测试时,程序的测试时间往往是一个需要考虑的重要因素。执行一次大型程序往往要耗费几个小时甚至几天的时间,现有的程序时间性能评价方法往往采用多次执行程序取平均值的方法获取时间性能指标,再对获取到的指标进行分析得出结论,这导致了对大型程序测试的效率十分低下,延误软件项目整体的测试效率。

#### 2.1.2 复杂运行环境导致的运行时间不确定性

在复杂的运行环境下,程序运行时间由很多因素决定,例如:数据传输、资源争夺等。复杂的运行环境导致了程序运行时间的不确定性,这种不确定性对程序时间性能的评价存在较大影响。以 LeNet 神经网络对某一特定图片数据集进行预测的程序为例,如图 1 所示,图中包含程序 10 次运行时间数据,程序运行时间的平均值在图中以虚线的方式表示出来。从图中可以看出,该程序运行 10 次的时间并不稳定,较运行时间的平均值波动较大。这种普遍存在的现象对程序的时间性能评价是一个需要考虑的重要问题。

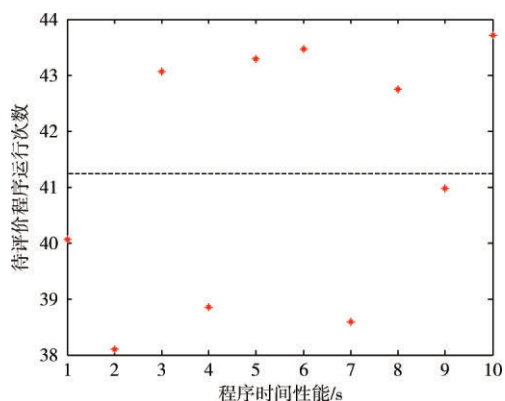


图 1 程序运行时间的不确定性

### 2.2 基于云模型的评价方法

针对以上问题,本文利用云模型,设计了一种程序时间性能评价方法。如图 2 所示,该方法中包含了一个正向云发生器和一个逆向云发生器,首先通过执行程序获得少量的程序运行时间数据传入逆向云发生器产生相应的云数字特征,之后将云数字特征传入正向云发生器生成更多的云数据,最后根据生成的云对程序时间性能进行评价。该方法的具体步骤如下,其中  $m$  和  $n$  为指定参数(本文将在实验部分对  $m, n$  的取值问题进行探讨):

**步骤 1** 获取  $m$  数量的待评价程序的运行时间数据。

**步骤 2** 计算样本数据的均值  $\bar{X}$  和方差  $S^2$ 。

**步骤 3** 计算  $Ex = \bar{X}$ ,  $En = \sqrt{\frac{\pi}{2}} \times \frac{1}{n} \sum_{i=1}^n |x_i - \bar{X}|$ ,

$He = \sqrt{S^2 - En^2}$ 。

**步骤 4** 以  $En$  为期望,以  $He^2$  为方差,生成正态随机

数  $En'$ 。

步骤 5 以  $Ex$  为期望, 以  $En'^2$  为方差, 生成正态随机数  $x$ 。

步骤 6 计算确定度  $\mu = \exp\left(-\frac{(x - Ex)^2}{2En'^2}\right)$ 。

步骤 7 若云滴数量达到预定数量  $n$ , 结束; 否则, 返回步骤 4 继续执行。

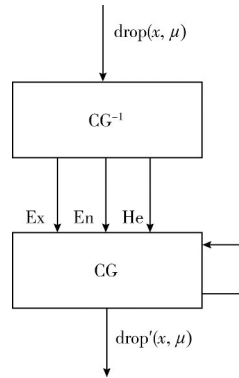


图 2 评价方法的整体框架

3 应用实例

为了说明本文提出的基于云模型的程序时间性能评价方法的实际应用效果, 本文接下来将使用单程序时间性能评价和多程序时间性能比较两个应用实例来进行验证。待评价程序运行环境如下: ① LeNet 程序: Red Hat 4. 4. 7 操作系统, Intel Xeon CPU, 16G RAM; ② Sharpness 程序: CentOS 6. 6 操作系统, Intel Xeon CPU, 32 G RAM。

3. 1 单程序时间性能评价

单程序时间性能评价是对一个程序运行时间的衡量, 本文选取 LeNet 程序进行应用实例验证。参数设置为  $m=70$ ,  $n=1000$ 。

首先执行  $m$  次程序, 记录  $m$  次程序运行的时间。之后利用样本数据计算云模型数字特征, 见表 1。

表 1 数字特征

Ex	En	He
1. 8522	0. 3495	0. 0552

利用数字特征生成数量为  $n$  的云滴, 结果如图 3 所示。

对生成的云模型图谱进行分析。首先需要观察云模型的期望; 其次是确定度, 在图谱中表现为云滴的集中程度, 体现了一种倾向的稳定程度; 最后是随机性, 体现了离群的程度。从图 3 中可以看出, 该程序的期望运行时间是在 2 s 附近, 生成的云滴较为集中, 表示程序整体运行时间比较稳定。

为了与实际程序运行情况进行比较, 执行程序 1000

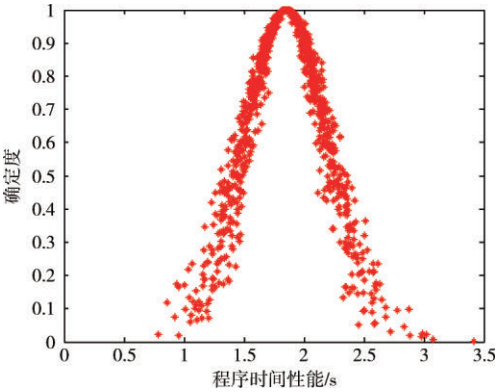


图 3 LeNet 云滴生成

次, 生成频数直方图, 如图 4 所示。从图 4 中可以看出, 该程序的实际运行情况与评价的结果基本吻合。

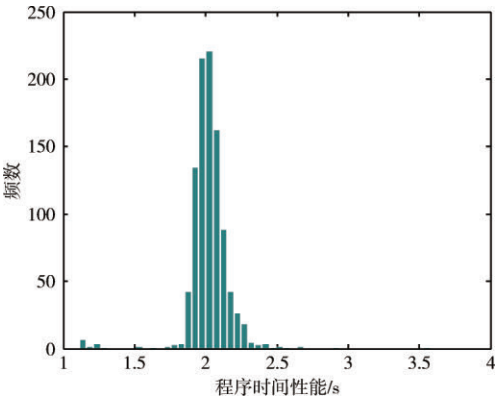


图 4 LeNet 频数直方图

3. 2 多程序时间性能比较

多程序时间性能比较是对多个程序的运行时间进行比较, 主要用于比较多个优化版本程序的优化效果。本文选取 sharpness 程序进行多程序时间性能比较的实例验证。sharpness 程序共有两个 SIMD 优化的版本 sharpness \_SIMD1 和 sharpness \_SIMD2, 分别进行了不同的 SIMD 优化, 评价的目标是找到运行时间最优的 SIMD 优化版本。参数设置为  $m=100$ ,  $n=1000$ 。

首先, 每个程序执行  $m$  次, 并记录  $m$  此程序运行的时间, 之后利用样本数据计算云模型数字特征, 见表 2。

表 2 数字特征

	Ex	En	He
sharpness	11. 9818	0. 0088	0. 0114
SIMD1	6. 7002	0. 0285	0. 0596
SIMD2	11. 9532	0. 0139	0. 0101

利用数字特征生成数量为  $n$  的云滴, 最终结果如图所示, 图 5 为 sharpness 原程序, 图 6 为 SIMD1 程序, 图 7 为 SIMD2 程序。

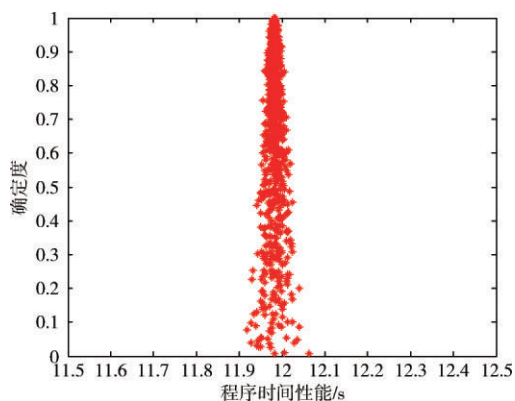


图5 Sharpness 程序云滴生成

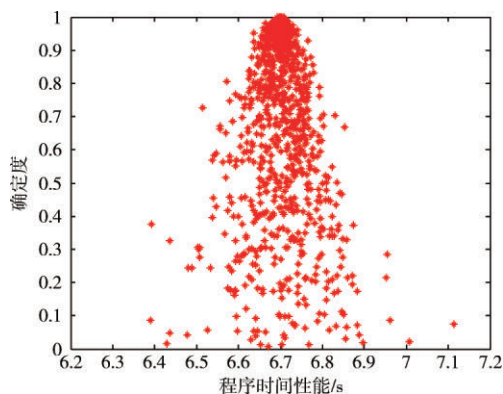


图6 SIMD1 程序云滴生成

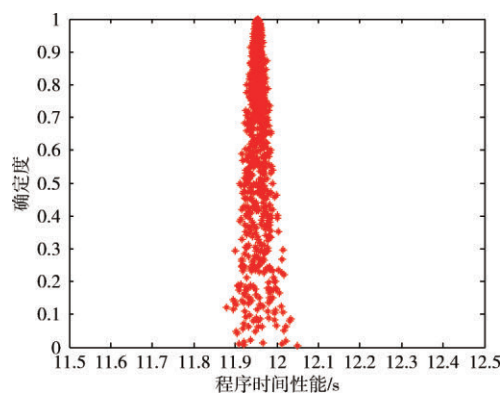


图7 SIMD2 程序云滴生成

从图6、图7中可以看出, SIMD1 程序的优化效果明显高于 SIMD2 程序的优化效果, 但 SIMD1 程序的时间性能稳定性不如原程序和 SIMD2 程序, 生成的云模型图谱的云滴不是很集中。

### 3.3 本文方法与传统方法的比较

传统的程序时间性能评价方法是通过多次执行程序进行平均值计算从而获得时间性能指标, 之后利用指标进行分析, 从而完成程序时间性能的评价。图8对本文方法和

传统方法的评价结果进行了比较, 该实验使用了 sharpness 的 SIMD1 版本程序的采样数据。

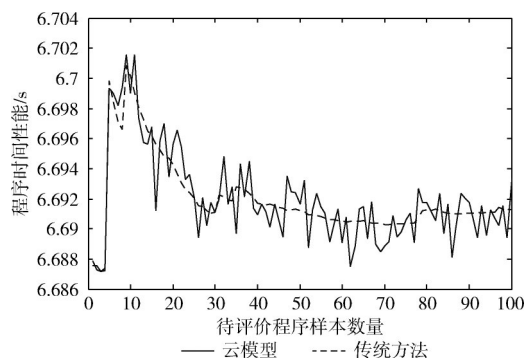


图8 本文方法与传统方法的结果比较

从图8中可以看出, 随着待评价程序的样本数量不断增加, 传统方法和本文方法的评价结果都不断趋于稳定, 同时本文方法增加了对样本数据的随机性的考虑。在评价结果上来看, 虽然本文方法和传统方法的评价方式和评价结果略有不同, 但两者的整体趋势和效果是相似的。

本文使用云模型设计了一种程序时间性能评价的方法, 结合上文的应用实例和对比实验图表, 总结该方法与传统方法的对比情况见表3。

表3 本文方法与传统方法的对比

	传统方法	本文方法
核心方法	统计数字特征	云模型
体现形式	平均数, 方差	云滴生成图
结果直观程度	不直观	直观
分析方式	定量	定性、定量
待评价程序的样本数量	大量	少量
不确定性	需进一步处理	直观表达
方法执行时间	快	较快
总体评价效率	与待评价程序的运行时间有关	高

在评价的核心方法上, 本文的方法将云模型引入到程序时间性能评价中, 充分挖掘了云模型在不确定性环境下的表示能力。在体现形式上, 传统方法的结果为平均数、中位数、方差等统计特征, 而本文方法的结果为云滴生成图, 云滴生成图能够非常直观地表达程序时间性能的分布, 包括它的稳定性和集中区域, 能够让使用者快速、直观地了解到程序运行时间的总体情况。云模型能够利用正向云发生器和逆向云发生器完成定性定量之间的转换, 从而实现程序时间性能在定性和定量两个方面的评价。传统方法往往需要固定的待评价程序的样本, 在面对运行时间很长的大型程序时会消耗大量的时间, 本文的方法只需要少量的待评价程序样本即可很好地完成时间性能的评价。在执行时间方面, 本文的方法相比较于传统方法, 主要增加的时间是在生成云滴的过程中。在总体评价效率上, 因为



传统方法与待评价程序的运行时间有关, 在面对运行时间较大的大型程序时效率较低, 本文的方法只需少量样本即可进行评价, 有着很高的总体评价效率。

## 4 实验评价

在实验部分, 本文对评价方法中的参数  $m$  和  $n$  对评价结果的影响进行测试。在模型中,  $m$  代表待评价程序的运行次数,  $n$  代表云滴的生成数量。

### 4.1 待评价程序运行次数对评价结果的影响

为了测试待评价程序运行次数对评价结果的影响, 在本实验中固定  $n$  值, 对  $m$  取值为 10, 20, 30, ..., 1000 进行评价方法的测试, 实验结果如图 9 所示。

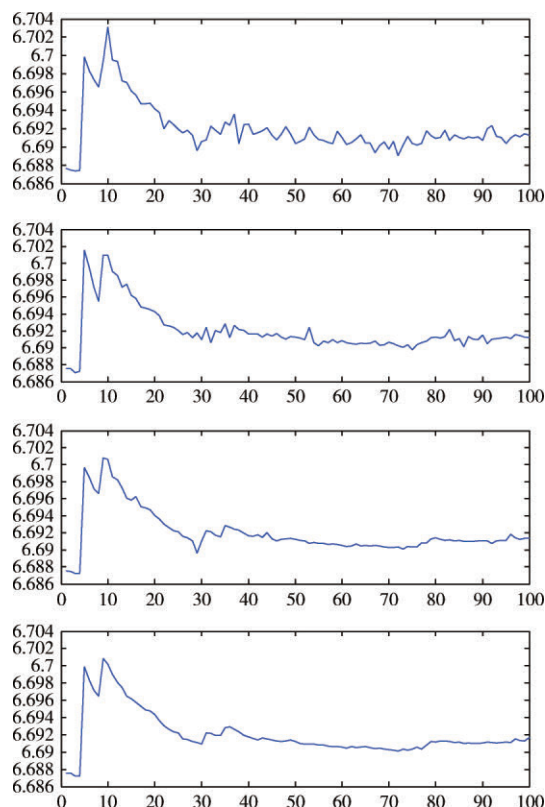


图 9 待评价程序运行次数对评价结果的影响

图 9 从上至下分别是  $n$  为 100, 200, 400, 800 这 4 种情况下的实验结果。从图中可以看出具有较大  $n$  值的图像曲线有着较少的波动, 整体比较稳定, 在  $n$  为 600 和 800 的情况下图像已经差别很小, 故最大  $n$  选为 800。从待评价程序运行次数对评价结果影响的角度来看, 最初在  $m$  较小时, 因为待评价程序运行时间的样本较小, 所以评价的结果与整体情况有着一定的偏差, 这导致了图中曲线在  $m$  较小时出现了很大的波动情况。同时也可以看出, 当  $m$  不断增大时, 评价结果会逐渐趋近于稳定。由此可见, 在条件允许的情况下,  $m$  越大, 评价方法的效果越好。

### 4.2 云滴生成数量对评价结果的影响

为了测试云滴生成数量对评价结果的影响, 在本实验中固定  $m$  的值, 对  $n$  取值为 100, 200, 300, ..., 10000 进行评价方法的测试, 实验结果如图 10 所示。

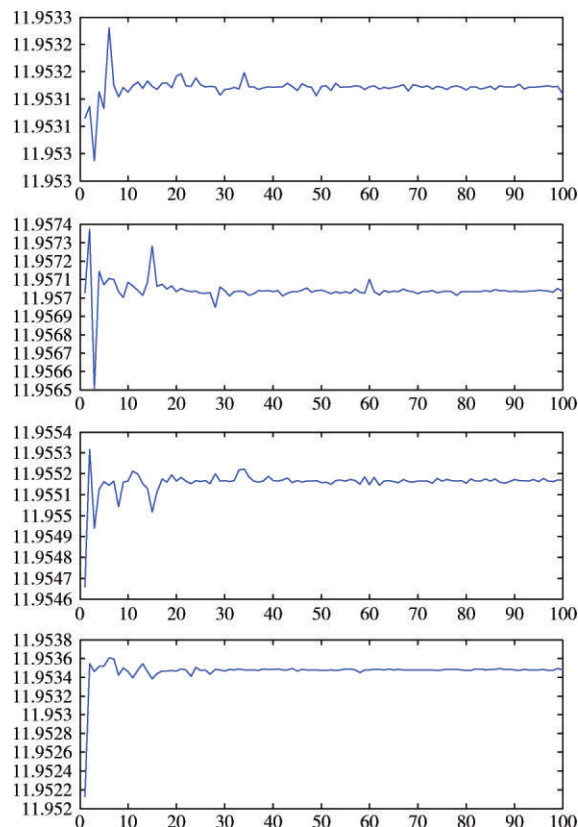


图 10 云滴生成数量对评价结果的影响

图 10 从上至下分别是  $m$  为 100, 200, 400, 800 这 4 种情况下的实验结果。从图中可以看出具有较大  $m$  值的图像曲线也有着较少的波动, 在  $n$  为 600 和 800 的情况下图像差别很小, 故最大  $m$  选取为 800。从云滴生成数量对评价结果影响的角度来看, 在  $n$  较小时, 因为云模型图谱中的云滴很少, 并不能很好地表现出整个云模型的情况, 导致评价的效果不是很好, 有着较大波动, 当  $n$  不断变大以后, 可以看到整体的趋势变得稳定, 此时评价效果已经基本稳定, 此时再增加  $n$  的值也不会提高评价的效果。

## 5 结束语

本文提出了一种基于云模型的程序时间性能评价方法, 该方法使用了云模型处理不确定性的优势, 改善了传统的程序时间性能评价方法的不足, 同时能够很好地适用于大型程序的时间性能评估。实验结果表明, 本文提出的评价方法可以有效地应用在单程序性能评价和多程序时间性能比较。在下一步工作中, 将在进一步提高程序时间性能评

价效果的基础上, 考虑如何利用云模型来对程序的综合性能进行评价。

### 参考文献:

- [1] YE Qiong, LI Shaowen, ZHANG Youhua, et al. Cloud model and application overview [J]. Computer Engineering and Design, 2011, 32 (12): 4198-4201 (in Chinese). [叶琼, 李绍稳, 张友华, 等. 云模型及应用综述 [J]. 计算机工程与设计, 2011, 32 (12): 4198-4201.]
- [2] FU Bin, LI Daoguo, WANG Mukuai. Review and prospect on research of cloud model [J]. Application Research of Computers, 2011, 28 (2): 420-426 (in Chinese). [付斌, 李道国, 王慕快. 云模型研究的回顾与展望 [J]. 计算机应用研究, 2011, 28 (2): 420-426.]
- [3] ZHANG Shibin, XU Chunxiang. Study on the trust evaluation approach based on cloud model [J]. Chinese Journal of Computers, 2013, 36 (2): 422-431 (in Chinese). [张仕斌, 许春香. 基于云模型的信任评估方法研究 [J]. 计算机学报, 2013, 36 (2): 422-431.]
- [4] WANG Shangguang, SUN Qibo, ZHANG Guangwei, et al. Uncertain QoS-aware Skyline service selection based on cloud model [J]. Journal of Software, 2012, 23 (6): 1397-1412 (in Chinese). [王尚广, 孙其博, 张广卫, 等. 基于云模型的不确定性 QoS 感知的 Skyline 服务选择 [J]. 软件学报, 2012, 23 (6): 1397-1412.]
- [5] ZHANG Qiuwen, ZHANG Yongzhi, ZHONG Ming. A cloud model based approach for multi-hierarchy fuzzy comprehensive evaluation of reservoir-indeed seismic risk [J]. Journal of Hydraulic Engineering, 2014, 45 (1): 87-95 (in Chinese). [张秋文, 章永志, 钟鸣. 基于云模型的水库诱发地震风险多级模糊综合评价 [J]. 水利学报, 2014, 45 (1): 87-95.]
- [6] WANG Jianqiang, PENG Juanjuan, ZHANG Hongyu, et al. An uncertain linguistic multi-criteria group decision-making method based on a cloud model [J]. Group Decision and Negotiation, 2015, 24 (1): 171-192.
- [7] DING Hao, WANG Dong. The evaluation method of water eutrophication based on cloud model [J]. Acta Scientiae Circumstantiae, 2013, 33 (1): 251-257 (in Chinese). [丁昊, 王栋. 基于云模型的水体富营养化程度评价方法 [J]. 环境科学学报, 2013, 33 (1): 251-257.]
- [8] Zhang Renlong, Shan Miyuan, Liu Xiaohong, et al. A novel fuzzy hybrid quantum artificial immune clustering algorithm based on cloud model [J]. Engineering Applications of Artificial Intelligence, 2014, 35 (1): 1-13.
- [9] Wang Guoyin, Xu Changlin, Li Deyi. Generic normal cloud model [J]. Information Sciences, 2014, 280 (1): 1-15.
- [10] SHEN Jinchang, DU Shuxin, LUO Yi, et al. Method and application research on fuzzy comprehensive evaluation based on cloud model [J]. Fuzzy Systems and Mathematics, 2012, 26 (6): 117-123 (in Chinese). [沈进昌, 杜树新, 罗伟, 等. 基于云模型的模糊综合评价方法及应用 [J]. 模糊系统与数学, 2012, 26 (6): 117-123.]
- [11] Kumrai T, Ota K, Dong M, et al. An incentive-based evolutionary algorithm for participatory sensing [C] //Global Communications Conference. IEEE, 2014: 5021-5025.
- [12] Mendez D, Labrador M, Ramachandran K. Data interpolation for participatory sensing systems [J]. Pervasive & Mobile Computing, 2013, 9 (1): 132-148.
- [13] WANG Ting. Research on 3D interpolation approaches based on Kriging with anisotropic spatial structures [D]. Nanjing: Nanjing Normal University, 2013: 1-65 (in Chinese). [王亭. 顾及各向异性的三维 Kriging 空间插值方法研究 [D]. 南京: 南京师范大学, 2013: 1-65.]
- [14] Ji HY, Ohm SY, Min GC. Brightness preservation and image enhancement based on maximum entropy distribution [M] //Convergence and Hybrid Information Technology, 2012: 365-372.
- [15] Solmaz G, Akbas MI, Turgut D. Modeling visitor movement in theme parks [C] //IEEE, Conference on Local Computer Networks. IEEE Computer Society, 2012: 36-43.

(上接第 434 页)