

# 基于lkp-tests的linux kernel性能分析

计算机科学与技术系 韩慧阳  
指导老师 陈康 陈渝  
2016年6月7日

# 提纲

- 研究背景、意义
- 研究目标和框架
- 研究内容
- 结论与后期工作
- 致谢

# 研究背景、意义 (详见论文第1章)

- 软件性能缺陷的广泛存在与难以检测
  - 不会造成直接的系统错误或崩溃
  - 检测方法的不成熟
    - 符号标记[5]、缺陷追踪[6]、负载测试[7]
  - 检测框架的缺乏
- lkp-tests应用潜力大但测试难度大
  - 测试冗余普遍存在
    - 测试指标冗余和测试配置冗余
  - 测试指标本身存在的变化误导缺陷检测

# 研究背景、意义

- lkp-tests发展前景严重受限
  - 创建之初力求覆盖面广泛
  - 测试数据过多限制了扩展性
  - 测试分析较少
- 去除冗余测试和指标变化模式分析的意义
  - 减少lkp-tests所需的测试量
  - 便于扩展和推广
  - 缺陷追踪方法的一个重要的数据来源，提高准确度
  - 检测性能缺陷的一个新的方向

# 研究目标和框架 (详见论文第2章)

- 研究目标
  - lkp-tests测试框架的去冗余
    - 指标冗余
    - 配置冗余
  - lkp-tests指标的变化模式分析与匹配
- lkp-tests结果形式

/result/ebizzy/100%-10x-10s/lkp-ws02/eywa-rootfs/x86\_64-rhel/gcc-4.9/6a13feb9c82803e2b815eca72fa7a9f5561d7861/

结果根目录/benchmark 名/硬件配置/负载/linux 发行版、文件系统/内核版本/编译器版本/commit ID

# 研究目标和框架

- 实验框架
  - 指标冗余
    - 选用主成分分析方法，实现指标降维
  - 指标相关性
  - 配置相关性
    - 配置的分离与单独配置的相关性分析
    - Pearson相关性衡量标准
  - 变化模式分析

# 研究内容

- 数据预处理（详见论文2.5.1及3.2.1）
  - 输入：lkp-tests测试结果源数据（磁盘中）
  - 输出：方便计算、整理的文本格式的文件
  - 目的：保证后续分析工作可行性和正确性

# 研究内容

- 数据预处理
  - 数据提取
    - 保存文件类型及其格式的确定
    - 从源数据提取到csv文件中
    - 建立索引、提取规模预估和多核加速
  - 数据清洗
    - 去除不完整数据
    - 去除bool值
    - 数据归一化



# 研究内容

- 数据预处理结果
  - 对应78个benchmark, 有78个csv格式的文件
  - 格式如下：

表 2-2 csv 结果文件的格式

bench mark	c1	c2	filesy stem	kernel	com piler	commit ID	Update .boot
aim7	100	lkp	eywa	x86_6	gcc-	6a13feb9c8	179.89
	%-1	-ws	-rootf	4-rhel	4.9	2803e2b815	
	0x-1	02	s			eca72fa7a9f	
	0s					5561d7861	

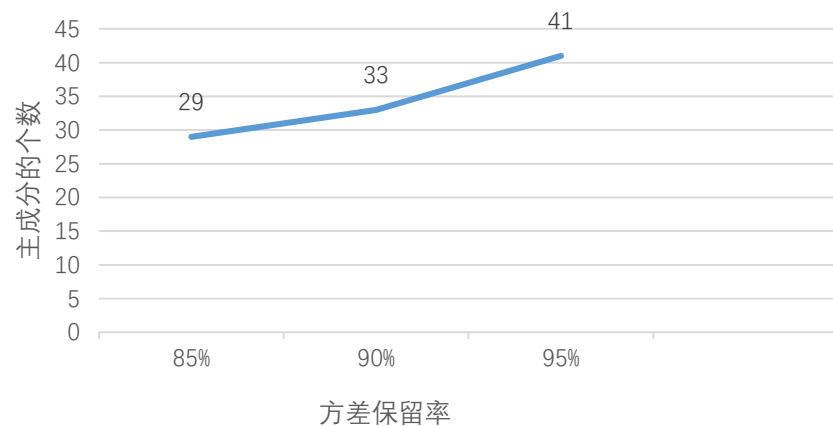
# 研究内容

- 指标降维（论文2.5.2及3.2.2）
  - lkp-tests力求覆盖全面，因此中存在大量的冗余测试
  - 测试指标总数大且包含很多赘余和无效指标
  - 测试规模限制了lkp-tests进一步的扩展
  - 降维有助于减少冗余、精简测试、帮助分析测试意义
- 方法论
  - 根据数据特征和研究目标选择降维算法（PCA）
  - 单个benchmark实施降维
  - 降维结果的分析与评估

# 研究内容

- 指标降维的结果（以aim7为例）
  - 选定主成分个数（33）

方差保留率和主成分个数关系图



# 研究内容

- 指标降维的结果
  - 前五个主成分解释了67%的方差

表 3-3 PCA 结果示例

---

Principal Component Analysis					
Call principal(r = Harman23.cor\$cov, nfactors = 5, rotate = "carimax")					
Standardizes loadings(pattern matrix) base upon correlation matrix					
	RC1	RC2	RC3	RC4	RC5
SS loadings	3.52	2.92	4.12	3.05	3.72
Proportion Var	0.24	0.17	0.11	0.08	0.07
Cumulative Explained	0.24	0.41	0.52	0.60	0.67

---

# 研究内容

- 指标降维结果分析
  - 见表3-4和表3-5
  - 对应每一个主成分选择一个指标
  - 从15681个指标降为33个
  - aim7的测试重点——系统的吞吐率

# 研究内容

- 指标相关性的分析（论文2.6.1及3.3.1）
  - 帮助减少指标性冗余
  - 分析benchmark内部指标的关系
  - 分析benchmark测试的侧重点
- 方法论
  - 确定相关性衡量标准（KPI）
  - 考虑相关性分析的置信度（数据的重叠程度）
  - 选择单个benchmark内部指标相关性分析
  - 选择相关性量度——Pearson相关系数

# 研究内容

- 指标相关性分析结果

- 结果示例：

表 3-6 ebizzy 的指标相关性分析结果示例

ebizzy.time.involuntary_context_switches	0.656361592
ebizzy.time.maximum_resident_set_size	0.618942495
ebizzy.time.minor_page_faults	0.992006463
ebizzy.throughput	1.000754148
ebizzy.time.voluntary_context_switches	0.579119672

- 准KPI和KPI的相关性较大
- KPI及其关系特别近的几个指标变化平缓，变化不大，可考虑删除
- ebizzy测试重点是系统IO

# 研究内容

- 配置相关性分析（论文2.6.2及3.3.2）
  - 减少配置冗余
  - 分析系统配置的特点和不足
- 方法论
  - 寻找衡量标准（主要配置）
  - 判断数据重叠程度
  - 选择需要测试的配置（综合配置和commit）
  - 衡量相关性



# 研究内容

- 配置相关性分析结果
  - 综合配置（硬件、linux版本）

表 3-7 综合配置的相关性分析结果

结果（仅展示 5 个）	配置	相关性
	1000-sync_disk_rw 和 vm-lkp-a05	0.62
	10000-jmp_test 和 lkp-a05	0.47
	1BRD_48G-btrfs-10-sync_disk_rw 和 ivb44	0.20
	30-performance-1-16G-ext4-500-disk_wrt 和 lkp-hsx03	0.13
	400-add_double 和 lkp-a04	0.08

# 研究内容

- 配置相关性分析结果
  - Commit相关性（了解版本迭代之间的特点）

表 3-8 commit ID 相关性分析结果

commit ID	相关性
1f93e4a96c9109378204c147b3eec0d0e8100fde	0.92
2a1ed24ce94036d00a7c5d5e99a77a80f0aa556a	0.87
2c6625cd545bdd66acff14f3394865d43920a5c7	0.67
31ade3b83e1821da5fbb2f11b5b3d4ab2ec39db8	0.14
3959df1dfb9538498ec3372a2d390bc7fbdbfac2	0.07

# 研究内容

- 指标变化模式检测（论文2.7及3.4）
  - 分析指标本身变化模式
  - 消除指标本身存在的变化的影响
- 方法论
  - 拓扑排序
  - 分析主分支上的变化
  - 模式识别与匹配

# 研究内容

- 配置变化模式检测结果
  - 波动型
  - 多与底层设计相关
  - 跳变型
  - 多为迭代型指标
  - 线性型
  - 多为整体性指标

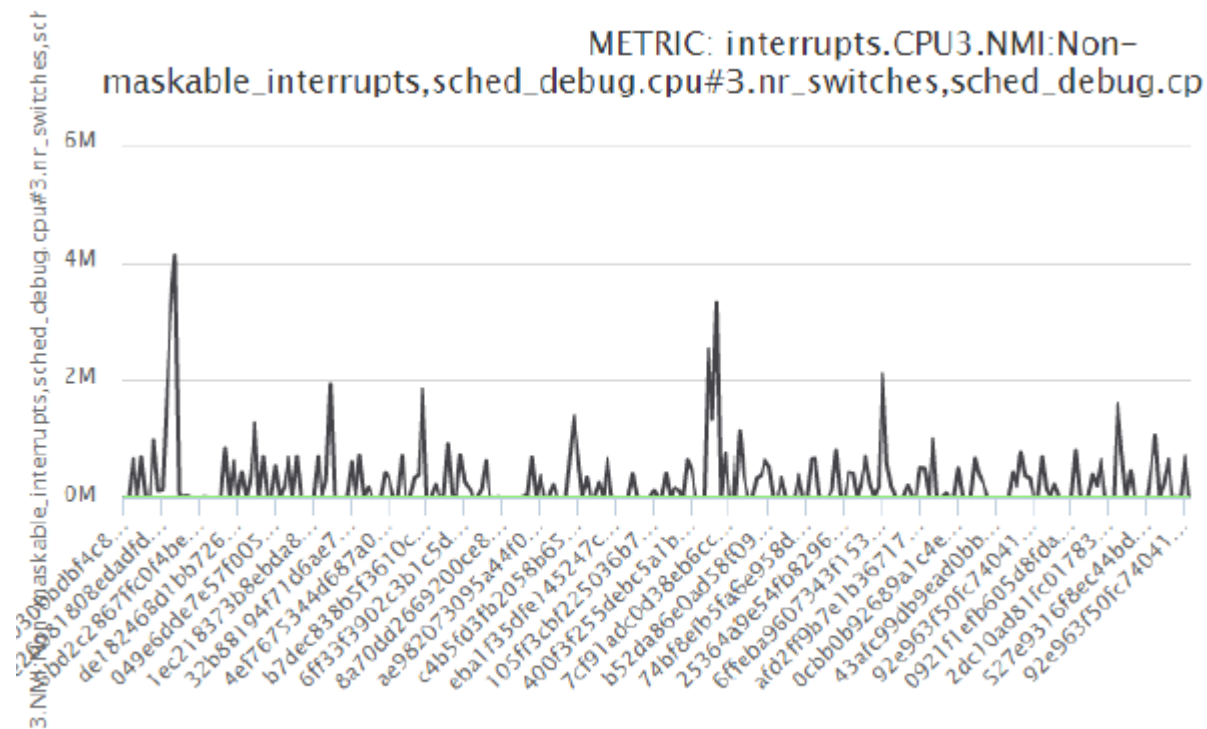


图 3-2 指标波动变化模式示意图

# 结论

- 主要进行了一下四个方面的工作：
  - 指标降维
  - 指标相关性分析
  - 配置相关性分析
  - 指标变化模式的变化
- 对于帮助lkp-tests的精简、改进和推广有重要意义

# 致谢

- 感谢陈渝老师在整个实验过程中提供的指导与帮助，老师的鼓励与指引给我带来了强大的动力！
- 感谢黄瀛工程师等在整个实验中给予我指导和帮助！

谢谢！