



单位代码

学 号ZY1321222

分 类 号

密 级不涉密

北京航空航天大学

B E I H A N G U N I V E R S I T Y

课程学习(笔记)

数理统计笔记

院 (系) 名 称	软件学院
专 业 名 称	软件工程与管理
学 生 姓 名	杨文静
指 导 教 师	王海泉

2013年10月



数理统计笔记

学 生： 杨文静

指导教师： 王海泉

摘 要

随着网络速度的提高，集群系统的应用越来越普遍，如何有效利用集群中空闲结点的内存以提高集群的整体性能这一问题越来越受到重视。

本文介绍了一种...

本文实现了一个 XXX 的原形系统，并对它的可用性和性能进行了测试。测试表明...

本文将首先对近年来相关的研究进行叙述，指出前人已经进行的工作和问题所在；之后，本文将介绍远程内存池的设计和远程内存池原形系统的实现；再后将给出对原形系统的测试数据，说明它的可用性和性能；最后将指出不足和改进的方向。

关键词：数理统计, 笔记



Notes for Mathematical Statistics

Author: Zhang San

Tutor: Wang Haiquan

Abstract

In recent years there has been a great interest in the use of clusters. Exploiting idle memory in clusters to avoid swapping and improve the performance is taken more and more attention.

In this paper, we introduce ...

This paper describes the design and implementation of XXX, and also implements a prototype. We tested and profiled the prototype, and found that ...

This paper first introduce previous work in related fields, then present the design detail of our implementation. In section 5, it deals with the detailed performance evaluation and analysis of our designs. Then, in the last section, it draw the conclusion and state future works.

Key words: First, Second



目录

1	绪论	1
1.1	背景及目的	1
1.2	国内外相关研究	1
1.3	问题的提出	1
1.4	论文构成	1
2	估计理论	2
2.1	参数的点估计	2
2.1.1	基本概念	2
2.1.2	频率替换法	2
2.1.3	矩估计法	3
2.1.4	极大似然估计法	3
2.2	信息不等式	3
2.3	相合估计	3
2.4	区间估计	3
3	结构与流程	4
4	设计与实现	5
5	性能测试	6
	结论	7
	致谢	8
	参考文献	11
	附录A XXX 源码简介	12
	附录B XXX 关键数据结构源码	13
	B.1 connector	13



附录C XXX 的使用	14
C.1 编译	14



1 绪论

1.1 背景及目的

介绍背景及目的.

1.2 国内外相关研究

介绍国内外相关研究.

根据要求, 如果引用是句子的一部分, 如: [1]提出, ..., 就要用\cite; 否则, 如: 有些人认为不是这样的^[2-4]. 但是..., 就要用\upcite

1.3 问题的提出

提出问题.

1.4 论文构成

论文结构.



2 估计理论

统计推断主要分为参数估计和假设检验，本章分析参数估计。参数估计又分为点估计和区间估计。

2.1 参数的点估计

2.1.1 基本概念

- 参数：在参数分布簇 $\{P_\theta, \theta \in \Theta\}$ 中，通常将任何与总体有关的带估计的量都看成参数，它可以是决定总体参数和参数的实函数。可以表示为 $q(\theta)$ 。
- 估计量：参数的任何实质统计量 $T(X_1, X_2, \dots, X_n)$ 都称为估计量。 $T(x_1, x_2, \dots, x_n)$ 称为估计值。
- 样本和样本观察值： X_1, X_2, \dots, X_n 称为样本， x_1, x_2, \dots, x_n 为样本观察值。在这里可以不区分。
- 统计量 $T(X_1, X_2, \dots, X_n)$ 为参数 $q(\theta)$ 的估计，记为 $\hat{q}(x_1, x_2, \dots, x_n) = T(X_1, X_2, \dots, X_n)$ 。

2.1.2 频率替换法

其中心思想是，大数定理，用频率替换相应的概率。考虑的对象是独立重复试验。

例子1 对于多项分布， $p(n_1, n_2, \dots, n_m) = n! / (n_1! n_2! \dots n_m!) p_1^{n_1} p_2^{n_2} \dots p_m^{n_m}$ 其最简单的估计是 $\hat{p}_i = n_i / n$ 。

$$\text{if } \theta = (\theta_1, \theta_2, \dots, \theta_l) \text{ and } p_i = h_i(\theta_1, \theta_2, \dots, \theta_l), i = 1, 2, \dots, m, l \leq m.$$

1. 将 θ_j 表示为 p_i 的函数。
2. 带入 $q(\theta)$, 可得 $q(\theta) = g(p_1, p_2, \dots, p_m)$ 。
3. 然后用频率替换概率 $q(\theta) = g(n_1/n, n_2/n, \dots, n_m/n)$



2.1.3 矩估计法

其中心思想是，大数定理,用样本矩替换总体矩。样本矩依概率收敛于总体矩。依然使用替换原理。

由课本43页例题2.1.4可知，令 $\mu_1 = E(x) = \mu$, $\mu_2 = E(X^2) = Var(X) + (E(x))^2 = \sigma^2 + \mu^2$,将 μ 和 σ^2 表示为k阶中心距的函数即可。

经验1 矩估计不唯一，处理这种问题的一种简便方法是构造方程组的总体矩的阶数尽可能小。

经验2 $E(x) = \mu$, $Var(X) = E(X^2) - (EX)^2$.

2.1.4 极大似然估计法

中心思想是参数估计使得某事件发生的概率大。

对每个x的值，取 $p(x, \hat{\theta}(x)) = \max_{\theta \in \Theta} p(x, \theta)$. 所以一个参数的极大似然估计也可以是一个实函数也可以是组合函数。

给定 x_1, x_2, \dots, x_n 为简单样本，则样本的似然函数为：

2.2 信息不等式

2.3 相合估计

2.4 区间估计



3 结构与流程

这一部分保密.

图3.1中,

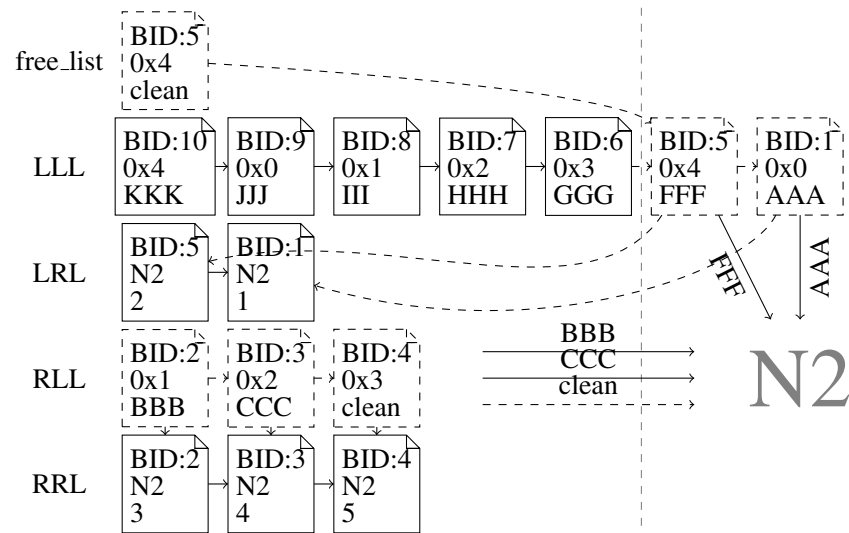


图 3.1: 示例三



4 设计与实现

这一部分保密.



5 性能测试

性能很好啊.



结论

根据要求, 结论一节是没有节号的.



致谢

首先, 我要衷心感谢北京航空航天大学计算机学院四年来对我的辛勤培养。我在计算机学院四年的经历将成为我一生的宝贵财富。

然后, 我要感谢....

感谢 Linus Torvalds, 他无私的开放了 Linux 的源代码, 为我们提供了课题和饭碗。感谢 VIM 的作者 Bram Moolenaar(虽然我没有给乌干达的可怜儿童和 Bram 捐款), 他设计的编辑器是世界上最好的编辑器。感谢 $\text{T}_{\text{E}}\text{X}$ 的作者 D. E. Knuth 教授、 $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X} 2_{\epsilon}$ 的作者 Leslie Lamport 以及 $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X} 2_{\epsilon}$ 中无数个宏包的设计者们, 他们的设计组成的这套排版系统使我可以排出精美的论文, 另外在紧张的论文撰写同时给我带来了许多快乐。

最后, 感谢 $\text{C}_{\text{T}}\text{E}_{\text{X}}$ 社区的所有朋友们。



参考文献

- [1] E. A. Anderson and J. M. Neeffe. An exploration of network ram[R]. Technical Report CSD-98-1000, UC Berkley, December 1994.
- [2] A. Acharya and S. Setia. Availability and utility of idle memory in workstation clusters[J]. *ACM SIGMETRICS Performance Evaluation Review*, 1999.
- [3] N.J. Boden, D. Cohen, R.E. Felderman, A.E. Kulawik, C. L. Seitz, J.N. Seizovic, and W. Su. Myrinet: A gigabit-per-second local area network[J]. *IEEE Micro*, 1995, 15(1):29–36.
- [4] Daniel P. Bovet and Marco Cesati. *Understanding the Linux Kernel*[M]. O'Reilly, 2005.
- [5] Jonathan Corbet, Alessandro Rubini, and Greg Kroah-Hartman. *Linux Device Drivers*[M]. O'Reilly, 2005.
- [6] M. J. Feeley, W. E. Morgan, F. H. Pighin, A. R. Karlin, H. M. Levy, and C. A. Thekkath. Implementing global memory management in a workstation cluster[J]. *ACM SIGOPS Operating Systems Review*, 1995, pages 201–212.
- [7] M. D. Flouris and E. P. Markatos. The network ramdisk: Using remote memory on heterogeneous nodes[J]. *Cluster Computing*, 1999, 2(4):281–293.
- [8] M. J. Frankling, M. J. Carey, and M. Livny. Global memory management in client-server dbms architectures[A]. In *Proceeding of the 18th VLDB Conference*[C]. 1992.
- [9] Jizhong Han, Dan Zhou, Xubin He, and Jinzhu Gao. I/O profiling for distributed ip storage systems[A]. In *Proceeding of The Second International Conference on Embedded Software and Systems*[C]. 2005.
- [10] X. He, Q. Yang, and M. Zhang. A caching strategy to improve iscsi performance[A]. In *Proceeding of Local Computer Networks*[C]. 2002.
- [11] M. R. Hines, M. Lewandowski, and K. Gopalan. Anemone: Adaptive network memory engine[D]. Master's thesis, Florida State University, 2003.



-
- [12] L. Iftode, K. Li, and K. Petersen. Memory servers for multicomputers[A]. In *Proceeding of the IEEE Spring COMPCON 93*[C]. 1993:538-547.
- [13] S. Koussih and S. Setia A. Acharyam. Dodo:a user-level system for exploiting idle memory in workstation clusters[A]. In *Proceeding of the Eighth IEEE International Symposium on High Performance Distributed Computing*[C]. 1999.
- [14] S. Liang, R. Notonha, and D. K. Panda. Swapping to remote memory over infiniband: An approach using a high performance network block device[J]. *IEEE Cluster Computing*, September 2005.
- [15] Robert Love. *Linux Kernel Development*[M]. Sams Publishing, 2005.
- [16] E. P. Markatos and G. Dramitions. Implementation of a reliable remote memory pager[A]. In *Proceeding of the 1996 Usenix Technical Conference*[C]. 1996.
- [17] John McCalpin. Streambenchmark. <http://www.streambench.org>.
- [18] T. Newhall, S. Finney, K. Ganchevm, and M. Spiegel. Nswap:a network swapping module for linux clusters[A]. In *Proceeding of Euro-Par'03 International Conference on Parallel and Distributed Computing*[C]. Klagenfurt, Austria, 2003.
- [19] J. Oleszkiewicz, L. Xiao, and Y. Liu. Parallel network ram: Effectively utilizing global cluster memory for large data-intensive parallel programs[A]. In *Proceeding of International Conference on Parallel Proceeding*[C]. 2004:577-592.
- [20] Fabrizio Petrini, Eitan Frachtenberg, Adolfo Hoisie, and Salvador Coll. Performance evaluation of the quadrics interconnection network[J]. *Journal of Cluster Computing*, April 2003, 6(2):125–142.
- [21] H. Tang Sun, M. Chen, and J. Fan. A scalable dynamic network memory service system[A]. In *Proceeding of High-Performance Computing in Asia-Pacific Region*[C]. 2005.
- [22] T. S. Trevisan, V. S. Costal, L. Whately, and C. L. Amorim. Distributed shared memory in kernel mode[A]. In *Proceeding of Computer Architecture and High Performance Computing*[C]. 2002.



-
- [23] L. Xiao, S. Chen, and X. Zhang. Adaptive memory allocations in clusters to handle unexpectedly large data-intensive jobs[J]. *IEEE Transactions on Parallel and Distributed Systems*, June 2004, 15(6):577–592.
- [24] The infiniband architecture. <http://www.infinibandta.org/spec>.



附录A XXX 源码简介

本附录对 XXX 的源代码做一简要介绍。

以下是 XXX 源码树的结构：

```
|-- Makefile
|-- common
|   |-- Makefile
|   |-- config.h
|   |-- debug.c
|   |-- debug.h
|   |-- fast_printk.c
|   |-- slabs.c
|   |-- slabs.h
|   |-- tty_printk.c
|   '-- ulist.h
```



附录B XXX 关键数据结构源码

本附录列出 RMP 关键数据结构的代码，以便读者理解前文提及的数据结构。

B.1 connector

connector.h 文件定义了 connector 结构。

```
1  /*
2   *
3   * connector.h
4   *
5   * ZhangSan, Mar. 20, 200X
6   *
7   * 连接抽象层是一个面向内核提供抽象连接的机制，每种类型的连接通过提供
8   * 一组 ops 和一个连接标识向核心应用提供透明的连接。
9   */
10
11 #ifndef CONNECTOR_H
12 #define CONNECTOR_H
13
14 #include "common/config.h"
15
16 #ifdef __KERNEL__
17 #include <linux/module.h>
18 #include <linux/byteorder/generic.h>
19 #include <asm/msr.h>
20 #endif
21
22 #define CONNSTRING_MAX 32
23 #define CONN_TYPE_SOCKET 1
24 #define CONN_TYPE_NOEXIST 10
25
26 /* 消息标识*/
```



附录C XXX 的使用

本附录介绍 XXX 的编译和使用。

C.1 编译

XXX 的源码经测试可以在 Kernel 2.6.11.12 和 Kernel 2.6.13 上编译通过...

获得源代码后，在根目录下执行“make”即可开始编译。正常的编译过程将有以下输出：

```
$ make
make -C manager
make[1]: Entering directory `/tmp/rmp/manager'
rm -f ../common/debug.o
```

之后会生成以下文件：

```
./common/fast_printk.ko
./common/tty_printk.ko
```