

具备高存储密度的新型 NAND 设备管理方案

卫兵¹, 郭玉堂¹, 宋杰², 张磊^{3*}

(1. 合肥师范学院 计算机科学与技术系, 合肥 230601; 2. 安徽大学 计算机科学与技术学院, 合肥 230039;

3. 计算智能与信号处理教育部重点实验室(安徽大学), 合肥 230039)

(* 通信作者电子邮箱 zhanglei@ahu.edu.cn)

摘要:针对嵌入式系统中 NAND 设备存储密度较低的问题,提出一种高存储密度的新型设备管理方案。通过研究大量 NAND 存储结构和 BCH 校验编码设计,在页面中找到一种通用的信息存储结构模式。使得冗余区(OOB)编码满足错误纠正码(ECC)纠错能力的同时可容纳设备分区管理信息,从而将主页面全部用于数据存储,并以此为基础进行了设备读写、损益均衡机制的设计。实验结果表明,所提方案中 NAND 设备数据存储密度可达 98%,优于当前多数主流文件系统。该方案具备很高的数据存储密度,设备读写效率和擦写寿命相对稳定,在嵌入式系统平台中具备很好的应用优势。

关键词: NAND; 错误纠正码; 存储密度; 设备管理; 损益均衡

中图分类号: TP302.1; TP302.7 **文献标志码:** A

New NAND device management solution with high storage density

WEI Bing¹, GUO Yutang¹, SONG Jie², ZHANG Lei³

(1. Department of Computer Science and Technology, Hefei Normal College, Hefei Anhui 230601, China;

2. School of Computer Science and Technology, Anhui University, Hefei Anhui 230039, China;

3. Key Laboratory of Intelligent Computing and Signal Processing (Anhui University), Ministry of Education, Hefei Anhui 230039, China)

Abstract: Focused on the problem of low storage density in embedded systems, in this paper, a new NAND device management solution with high storage density was proposed. In the proposed solution, a generalized mode of information structure in NAND page was designed by researching a great number of NAND storage structures and BCH(Bose-Chaudhuri-Hocquenghem) parity coding programs. In the mode, data layout in Out of Band (OOB) could achieve Error Correcting Code (ECC) capability while accommodating device management information of partition, thus the main page could be completely used for data storage, which can be treated as a basis for development of device read-write solution and Wear Leveling mechanism. The experimental results show that the proposed solution improves storage density up to 98%, and it is superior to most current common file systems. Having an excellent data storage density, as well as relatively stable device read-write efficiency and Program/Erase (P/E) endurance, the solution has good application advantages in embedded systems.

Key words: NAND; Error Correcting Code (ECC); storage density; device management; wear leveling

0 引言

目前嵌入式平台中 NAND Flash 已成为主流的存储介质。NAND 设备具备自身特殊的物理特性,既不属于字符设备,也不属于块设备,同时,还存在坏块和擦写(Program/Erase, P/E)寿命问题^[1]。这些因素影响嵌入式文件系统中 NAND 设备管理层的设计,如何高效管理和使用 NAND 设备成为文件系统研发的热点。基于 NAND 的主流文件系统主要有闪存日志型文件系统(Journaling Flash File System2, JFFS2)、另类闪存文件系统(Yet Another Flash File System2, YAFFS2)、无排序区块图像文件系统(Unsorted Block Image File System, UBIFS)类型,这些文件系统在数据存储、输入输出(Input/Output, I/O)速率、损益均衡(Wear Leveling)、垃圾回收等方面都进行了很多针对性的设计^[2]。JFFS2 是一个日志结构

(log-structured)的文件系统,架设于内存技术设备(Memory Technology Device, MTD)之上,系统节点信息占用 NAND 存储页面,存储空间利用率不高^[3]。UBIFS 是 JFFS2 的下一代,具备优异的数据 I/O 速率和可扩展性,架设于无排序区块镜像(Unsorted Block Image, UBI)设备之上,依赖 UBI 对 NAND 进行管理。UBI 设备负责坏块管理、逻辑卷管理、Wear Leveling 等功能,UBI 设备信息需要占用 NAND 存储块的前两个页面,UBIFS 节点信息也需要占用页面存储空间,因此 UBIFS 数据存储密度也较低^[4]。YAFFS2 充分利用了页面空间,将坏块标志码、错误纠正码(Error Correcting Code, ECC)、文件系统管理信息存储于冗余区(Out of Band, OOB),具备很高的数据存储密度^[5-6]。但 YAFFS2 对于页面 OOB 区域的使用定义存在与 NAND 底层驱动程序冲突的情况,导致系统移植操作出错。实际使用中通常需要针对驱动层进行额外

收稿日期:2014-03-04;修回日期:2014-04-12。

基金项目:安徽省高校省级自然科学研究重点项目(KJ2013A217);安徽省优秀青年人才基金资助项目(2011SQRL020ZD)。

作者简介:卫兵(1984-),男,安徽六安人,助教,硕士,主要研究方向:嵌入式系统、模式识别、智能数据处理;郭玉堂(1962-),男,安徽安庆人,教授,博士,主要研究方向:图像处理、模式识别、计算机网络;宋杰(1966-),男,安徽合肥人,副教授,博士,主要研究方向:嵌入式系统;张磊(1982-),男,安徽蚌埠人,讲师,博士研究生,主要研究方向:信号处理、嵌入式系统。

修改,存在诸多不便。另外,YAFFS2 中使用自带 ECC 校验编码,也会与驱动层 ECC 校验发生冲突,使用中限制了 ECC 校验的选择性和灵活性^[5-6]。

针对主流 NAND 文件系统中设备管理层存在数据存储密度低或者与底层驱动冲突等情况,本文在对 NAND 设备结构、ECC 校验信息以及存储块管理需求进行分析的基础上,提出一种新型 NAND 设备管理方案。设计中数据存储结构及设备管理结构进行优化,在提高设备数据存储密度的同时,为系统上下层提供灵活可靠的操作接口。

1 OOB 区域结构分析与设计

传统文件系统中只有 YAFFS2 有效利用了 NAND 设备页面 OOB 区域。本文通过分析对比,在 OOB 中设计了一套通用存储结构,将 OOB 区域按照 ECC 校验码布局划分为不同扇面(Sector),再对不同 Sector 进行统一设计和管理,以适应所有制程型号 NAND 的使用需求。

1.1 OOB 结构与 ECC 校验码分析

NAND 设备按照页面结构可分为小页面型(Small Page)和大页面型(Large Page)。Small Page 结构为 512 B 主页面(Main)加 16 B 冗余区(OOB),ECC 纠正需求通常为 1 bit/528 B^[7]。Large Page 结构包括 2048 B Main + 64 B OOB、4096 B Main + 128 B OOB、4096 B Main + 224 B OOB、8 192 B Main + 448 B OOB、8 192 B Main + 512 B OOB 等多种结构类型,ECC 纠正需求通常为 1 bit/528 B、24 bit/1 024 B、40 bit/1 024 B 不等^[8]。

目前系统级芯片(System On Chip, SOC)中 NAND 控制器广泛使用 BCH 校验码作为 ECC 编码方式,BCH 码具备编码效率高、纠错能力强等特点。BCH 编码针对当前 NAND 设备型号可以支持从 4 bit 到 60 bit 不同纠错能力的校验编码^[9-10]。根据 BCH 编码特点可以对页面结构进行设计调整。

1.2 新型 OOB 存储结构

通过以上分析,设想将 NAND 页面布局进行重新设计,使得 OOB 区在满足 ECC 纠错能力的同时尽可能节约出空余区段作为文件管理区(File Management Area, FMA)用于记录 NAND 管理信息。图 1 为 NAND 页面结构设计示意图。

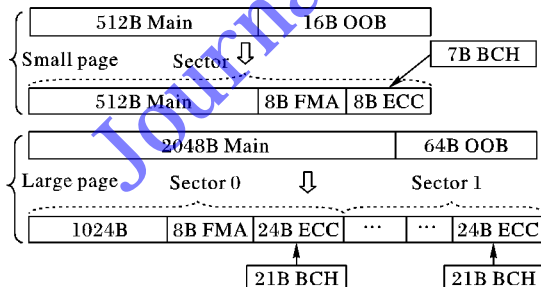


图1 新型页面存储结构设计

图 1 列举了页面 512 B 和 2048 B 型 NAND 使用新存储结构设计示意图。Small Page 结构中,512 B Main + 16 B OOB 页面,采用 520 B-4 bit BCH 编码模块,校验码长为 14,可计算出校验码总长度 $14 \times 4 = 56 \text{ bit} = 7 \text{ B}$ ^[9-10]。即页面结构划分为:512 B Main + 8 B FMA + 8 B ECC,后 8 B 作为校验码区可以满足 BCH 校验码的存放,实现对前 520 B 的 4 bit 纠错,前 8 B 区段被节约出用作 FMA。在 Large Page 结构中,2048 B Main +

64 B OOB 页面划分为两个 1024 B Main + 32 B OOB Sector,针对每个 Sector 采用 1032 B-12 bit BCH 编码模块,校验码长为 14,可计算出校验码总长度 $14 \times 12 = 168 \text{ bit} = 21 \text{ B}$ 。即 Sector 结构划分为:1024 B Main + 8 B FMA + 24 B ECC,后 24 B 作为校验码区可以满足 BCH 校验码的存放,实现对前 1032 B 的 12 bit 纠错,满足 NAND 芯片的 1 bit/528 B ECC 需求。

图 1 中 2048 B Main + 64 B OOB 页面采用 2 个 Sector 串联式存储结构,主要考虑 NAND 控制器采用存储器直接访问(Direct Memory Access, DMA)方式编程时可以顺序向 NAND 引脚写入数据。当 NAND 控制器写完缓存中的 1024 B Main + 8 B FMA 数据后,BCH 编码器会运算出 ECC 校验码,紧接着顺序写入(此时 8 B FMA + 24 B ECC 占用原本 Main data 区域),然后再进行下一个 Sector 处理。这样就可以实现快速页面编程(Page Program),而将 8 B FMA + 24 B ECC 数据存入 OOB 区域则需要控制器执行低效率的随机编程(Random Program)。需要提及的是,采用新型存储结构会导致原本 OOB 中坏块标志位(Bad Block Mark, BBM)跟随第一个 FMA 一起被移走,而出厂时少数坏块的 BBM 仍旧处于传统位置,造成管理不一致问题,在后续驱动部分设计中会加以解决。

同理,其余型号 Large Page 结构的 NAND 也依据该思想进行划分,将页面结构划分为 2,4,8 以及更多的 Sector,结构为 1024 B Main + 8 B FMA + XB ECC,满足 ECC 校验的同时划分出统一的 8 B FMA 用于设备管理。本文将目前主流型号结构的 NAND 进行逐一分析,统一设计出页面结构类型列表。如表 1 所示。

表1 NAND 新型页面结构类型

NAND 出厂页面结构(Main + OOB)/B	NAND 出厂 ECC 纠错需求	页面拆分成 Sector 数目	Sector 结构(Main + FMA + ECC)/B	ECC 校验码长度/B	ECC 纠错能力
512 + 32	1 bit/528 B	1	512 + 8 + 8	7	4 bit/528 B
2048 + 64	1 bit/528 B	2	1024 + 8 + 24	21	12 bit/1032 B
4096 + 128	1 bit/528 B	4	1024 + 8 + 24	21	12 bit/1032 B
4096 + 224	24 bit/1032 B	4	1024 + 8 + 48	42	24 bit/1032 B
8192 + 436	24 bit/1032 B	8	1024 + 8 + 48	42	24 bit/1032 B
8192 + 448	24 bit/1032 B	8	1024 + 8 + 48	42	24 bit/1032 B
8192 + 512	24 bit/1032 B	8	1024 + 8 + 48	42	24 bit/1032 B
8192 + 640	40 bit/1032 B	8	1024 + 8 + 72	70	40 bit/1032 B

从表 1 分析中可以看出,目前主流 NAND 页面结构均可采用 Sector 划分思想对页面进行重新布局分配。Small Page 型号的 Sector 结构为 512 B + X,其余 Large Page 为 1024 B + Y。另外,针对两种类型 NAND 的 BBM 位置不同(Small Page 结构中出厂 BBM 位于 OOB 区域第 6 个字节,Large Page 位于第 0 个字节),设计中分别对两种 Sector 结构中 8 B FMA 布局进行了相应规划。如图 2 所示。

图 2 为 Sector 中 8 B FMA 数据布局设计。逻辑块编号(Logic Block Number, LBN)记录物理块所映射的逻辑块编号。擦写计数器(Erase Counter, EC)记录物理块擦写次数。BBM 即坏块标志位,设计中将 BBM 扩展为两个字节长度以提高坏块判别准确性。由于采用新型页面 Sector 结构,BBM 标志位会随 FMA 被移走,而传统出厂坏块标志位会被其他数据填充,因此,在底部驱动层添加一次数据交换操作,将

Sector 页面布局中 BBM 标志位和传统坏块位置上的数据进行交换,再执行写数据操作。同理,驱动层读取数据后对缓存中进行数据交换再返回至上层。

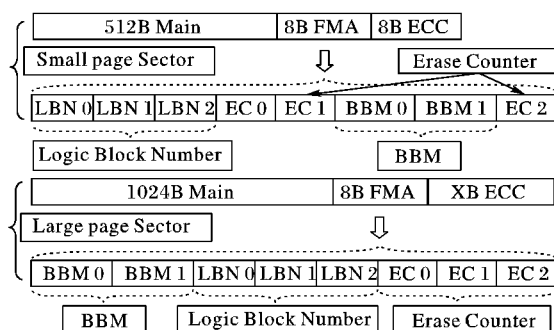


图2 8B FMA 数据布局设计

2 NAND 设备管理层架构

依据 NAND 页面结构划分与 OOB 数据布局设计,可对 NAND 设备进行分区管理。设计中定义两种类型分区:静态设备分区 (Static Device Partition, SDP) 与动态设备分区 (Dynamic Device Partition, DDP)。其中,SDP 用于静态数据存储,DDP 用于动态数据存储。NAND 设备管理架构如图3所示。

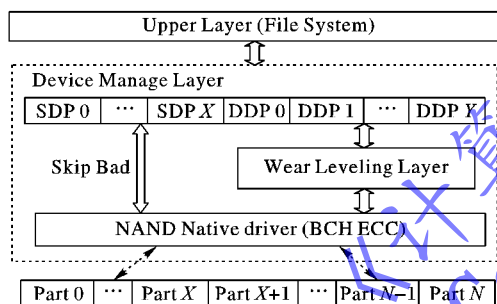


图3 NAND 设备管理架构

2.1 SDP 设计

SDP 存储静态数据,如系统 Boot-loader 代码区、内核代码区、根目录文件区等。这类分区在系统中通常运行状态是只读的,例如启动过程中读取系统级代码。SDP 利用 Sector 中 LBN 区段将逻辑存储块映射到物理存储块,并跳过坏块。考虑 SDP 主要用于存储系统级代码,升级次数通常处于可预测范围内,当出现意外坏块时寻找 EC 次数最小的存储块重新建立映射,不需要进行复杂损益均衡设计,结构简洁可靠。SDP 存储块管理如图4所示。

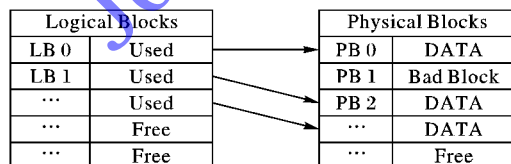


图4 SDP 存储块管理

2.2 DDP 设计

DDP 通常用于存储应用程序文件目录,系统运行中会不断进行数据更新,尤其是网络下载应用,因此需要涉及设备读写、坏块管理、Wear Leveling 等一系列设计。DDP 存储块结构如图5所示。

DDP 将所有正常存储块统一分为已用块队列 (Used

Pool) 和空闲队列 (Free Pool)。Used Pool 为正在使用的存储块,通过 FMA 结构中 LBN 实现逻辑块到物理块的映射;Free Pool 为空闲存储块队列,FMA 结构中 LBN 为 0xFF。DDP 数据操作中:若写数据时上层系统通过 LBN 映射表找到对应物理块,擦除该物理块,然后将数据写入目标块中;若写数据时 LBN 映射表中逻辑编号无物理块映射,则从 Free Pool 中取出 EC 最小的物理块,写入数据并建立映射关系;若删除目标数据,则将 Used Pool 中目标物理块 FMA 结构中 LBN 改写为 0xFF,加入 Free Pool 中。

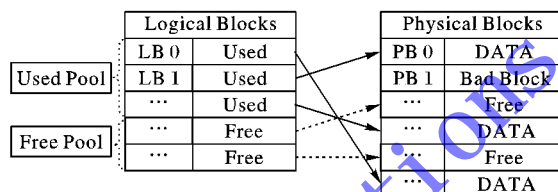


图5 DDP 存储块管理

DDP 运行过程中需要充分考虑 NAND 存储块损益均衡问题。Wear Leveling 层采用后台 (Background) 线程运行方式^[11-12]。设计中使用增序链表 (Increasing EC list) 维护分区中所有物理块的 EC 值,调用 Erase 动作后链表中对应物理块的 EC 计数器发生变动,启动 Wear Leveling 线程:分析链表中 EC 最大值与最小值的差距,若差值超过预设门限值 (注:EC 最大值所在物理块为最后一次 Erase 块),则意味分区存储块的擦写寿命失去平衡,将两个物理块中数据作交换,并修改 LBN 映射表 (Change LBN mapping)。若 Erase 动作为删除数据,则将 EC 最小值对应的物理块交换到 Free Pool 中。系统中将门限值预设为 5000。图6为 DDP 写数据操作以及 Wear Leveling 层工作流程。

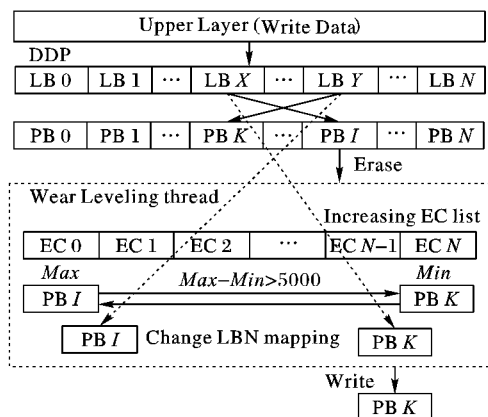


图6 DDP 工作流程

3 方案分析评估

3.1 存储密度分析

使用新型页面 Sector 布局方案,以 BCH 校验编码为基本单位对页面进行 Sector 标准划分,将 NAND 设备管理层信息均压缩至 OOB 区域,使得主页面区域全部使用作原始数据存储,设备空间利用率上优于目前大多数闪存文件系统的设备管理层。本文对所提方案与主流文件系统进行了对比测试,在 ARM Cortex-A9 处理器平台上选用 SAMSUNG K9K8G08U0M 型 NAND 作为测试设备,K9K8G08U0M 的容量为 1 GB,页面为 2 KB,块为 128 KB。系统分区中制作了

JFFS2、YAFFS2、UBIFS 和本文方案 DDP 等 4 个分区,大小均为 32 MB,分区中预留坏块替换数统一设置成 5。使用 2 KB 的 bin 文件对各分区进行连续写操作以测试分区存储容量,测试数据统计如表 2 所示。从实验结果可以看出,采用 Sector 布局的本文方案和 YAFFS2 在设备存储密度上明显优于其他方案,同时,本文方案在架构上克服了 YAFFS2 与底层兼容性不足的缺陷。

表 2 各种方案 NAND 分区存储密度测试对比

文件系统	分区大小/MB	存储容量/MB	存储密度/%	与底层兼容性
JFFS2	32	25.20	78.75	无冲突
YAFFS2	32	30.15	94.20	有冲突
UBIFS	32	24.90	77.80	无冲突
本文方案 DDP	32	31.37	98.00	无冲突

3.2 设备寿命分析

设备分区采用管理信息存储于 OOB 区域的设计思想,在数据擦写操作时会面临擦写次数增多的困扰。当删除文件节点时,相应的存储块会被擦除,文件系统上层需要擦除 Sector 主页面数据和 FMA 字段,保留 BBM 和 EC 信息,再将物理块放入 Free Pool 中。而 NAND 物理块擦除时会对所有存储位进行充电,因此,设备管理层需要将 BBM 和 EC 信息写回 Sector 分区,系统再次使用 Free Pool 队列中块时需要再次进行擦除。UBI 采用两个页面存放管理信息,充分利用了闪存物理块可按页面顺序擦写的特性,启用 Free Pool 块时可节省一次擦除操作,是该管理设备的优点。该问题在 YAFFS2 中同样存在,以 P/E 寿命为代价换取高存储密度,从系统设计角度可通过 Wear Leveling 层尽量加以克服。方案中 DDP 动态分区在应用程序执行大量数据删除和下载时 P/E 效率较低,执行数据更新则具备较好效率。

4 结语

采用 NAND 页面 Sector 标准结构划分思想,结合高效的 BCH 编码方式,通过统一设计的数据布局将设备管理信息压缩至 OOB 区域,从而获得很高的数据存储密度。通过 Wear Leveling 层设计对固有的 P/E 寿命问题加以克服和改善,使得整体性能和稳定性处于较好的状态。方案中 DDP 设备 P/E 寿命问题需要进一步研究改进,设备与上层文件系统接口层也有进一步设计空间。针对新出现的 NAND 页面结构类型需要不断维护更新列表。

参考文献:

- [1] LI Q, SUN M. Design of NAND Flash memory-based embedded file system [J]. Application Research of Computers, 2006, 23(4): 231-239. (李庆诚, 孙明达. 基于 NAND 型闪存的嵌入式文件系统设计[J]. 计算机应用研究, 2006, 23(4): 231-239.)
- [2] SUN X, SHI X. Performance evaluation about Flash file system on embedded Linux [J]. Control and Automation, 2012, 27(10): 175-243. (孙晓荣, 时兴. 基于嵌入式 Linux 的 Flash 文件系统的实时性能研究[J]. 微计算机信息, 2012, 27(10): 175-243.)
- [3] LI J, YAN P, PENG K. Improvement of installation mechanism of JFFS2 file system [J]. Computer Applications and Software, 2008, 25(8): 237-239. (李杰, 鄢萍, 彭凯. JFFS2 文件系统挂载机制的改进[J]. 计算机应用与软件, 2008, 25(8): 237-239.)
- [4] HAN C, CHEN X, LI X, et al. Impact of UBIFS wear-leveling on system I/O performance [J]. Computer Engineering, 2009, 35(6): 260-262. (韩春晓, 陈香兰, 李曦, 等. UBIFS 损耗均衡对系统 I/O 性能的影响[J]. 计算机工程, 2009, 35(6): 260-262.)
- [5] YANG C, LEI H. Improvement and optimization of embedded file system based on NAND Flash [J]. Journal of Computer Applications, 2007, 27(12): 3102-3104. (杨春林, 雷航. 基于 NAND Flash 的嵌入式文件系统的改进与优化[J]. 计算机应用, 2007, 27(12): 3102-3104.)
- [6] WILDANI A, MILLER E L, WARD L. Efficiently identifying working sets in block I/O streams [C]// SYSTOR 2011: Proceedings of the 4th Annual International Conference on Systems and Storage. New York: ACM Press, 2011: Article No. 5.
- [7] ZHANG H, YAN Y, LUO Y. Optimization design of NAND Flash transition layer based on the large-scale NAND Flash [J]. Computer Engineering and Science, 2011, 33(4): 81-85. (张辉, 晏益慧, 罗宇. 大容量 NAND Flash 文件系统转换层优化设计[J]. 计算机工程与科学, 2011, 33(4): 81-85.)
- [8] LI J, JIN L, LI G, et al. NAND Flash error correction arithmetic based on ECC embedded BCH code [J]. Journal of Harbin Engineering University, 2012, 33(11): 1399-1404. (李进, 金龙旭, 李国宁, 等. ECC 嵌入 BCH 码的闪存纠错算法[J]. 哈尔滨工程大学学报, 2012, 33(11): 1399-1404.)
- [9] WANG J, SHEN H. Design of BCH encoder/decoder for NAND Flash controller [J]. Computer Engineering, 2012, 33(11): 1399-1404. (王杰, 沈海斌. NAND Flash 控制器的 BCH 编/译码器设计[J]. 计算机工程, 2010, 36(16): 222-225.)
- [10] WANG F, HE X, ZHU W, et al. Error detection and correction algorithm for RS code based on storage with Flash memory [J]. Computer Engineering, 2011, 37(12): 245-247. (王方雨, 何昕, 朱玮, 等. 基于闪存储存的 RS 码检纠错算法[J]. 计算机工程, 2011, 37(12): 245-247.)
- [11] CHANG L. On efficient wear leveling for large-scale Flash-memory storage systems [C]// Proceedings of the 2007 ACM Symposium on Applied Computing. New York: ACM Press, 2007: 1126-1130.
- [12] CHEN F, LUO T, ZHANG X. CAFTL: a content-aware Flash translation layer enhancing the lifespan of flash memory based solid state drives [C]// Proceedings of the 9th USENIX Conference on File and Storage Technologies. Berkeley: USENIX Association, 2011: 6-20.
- [13] (上接第 2433 页)
- [14] KRESTA J V, MACGREGOR J F, MARLIN T E. Multivariate statistical monitoring of process operating performance [J]. The Canadian Journal of Chemical Engineering, 1991, 69(1): 35-47.
- [15] WANG H, LI S. The bounds of restricted isometry constants for low rank matrices recovery [J]. Science China: Mathematics, 2013, 56(6): 1117-1127.
- [16] LI H, XIAO D. Survey on data driven fault diagnosis methods [J]. Control and Decision, 2011, 26(1): 1-9. (李哈, 萧德云. 基于数据驱动的故障诊断方法综述[J]. 控制与决策, 2011, 26(1): 1-9.)
- [17] NIAZI A, LEARDI R. Genetic algorithms in chemometrics [J]. Journal of Chemometrics, 2012, 26(6): 345-351.