文章编号:1001 - 9081(2006)08 - 1890 - 04

体全息数据存储文件系统空间分配策略研究

易法令[1,2]谢长生[2],吴 非[2]
(1. 长江大学 计算机科学学院,湖北 荆州 434023;
2. 华中科技大学 外存储系统国家专业实验室,湖北 武汉 430074)
(flyi@jznu.net)

摘 要:通过分析体全息数据存储的物理寻址方式,提出了对逻辑块地址实行二维编址的策略。以二维逻辑块地址为基础,对体全息数据存储文件系统空间分配的连续性进行了研究,提出了"顺序连续"和"随机连续"的概念。对二维区域采用了四叉树结构进行组织,并以四叉树为基础设计了成组分配空闲块算法。在体全息数据存储文件系统原型系统的基础上,对模拟的全息存储体进行"文件级"的测试,结果证明二维分配策略能有效地提高文件数据块分配的连续性。

关键词:体全息数据存储;文件系统;分配四叉树;逻辑块地址
中图分类号: TP311.12 **文献标识码**:A

Research on space allocation strategy of the volume holographic data storage

YI Fa-ling[1,2], XIE Chang-sheng[2], WU Fei[2]
(1. *College of Computer Science, Yangtse University, Jingzhou Hubei* 434023, *China*;
2. *National Storage System Laboratory, Huazhong University of Science and Technology, Wuhan Hubei* 430074, *China*)

Abstract: Through analyzing physical addressing mode of volume holographic data storage (VHDS), two-dimensional addressing strategy of logical block address was given, by which the data-block of the file was allocated. By researching the continuity of file data-block allocation on two-dimensional logical block address, the concepts of "sequence continuity" and "stochastic continuity" were presented. Based on allocation quadruple-tree which was adopted as framework of the two-dimensional logic area, the allocation algorithm for multiple block was designed. On the prototype system of VHDS, testing of the allocation strategies of data-block in the "file" level was made. The testing results prove that two-dimensional allocation strategy can make file data-blocks to be allocated more consecutive.

Key words: volume holographic data storage; file System; allocation quadruple-tree; logical block address

0 引言

新型三维存储技术——体全息数据存储技术正在迅速发展,具有存储容量大、数据传输率高及按内容寻址功能等特点,从物理介质的层面满足了人们对存储设备的要求,是极具潜力的新一代存储方案[1,2]。由于体全息数据存储系统的物理寻址方式、存取模式等方面与磁盘有很大的差别,现有的文件系统显然不能适应体全息存储的要求,因此,对多维存储文件系统的研究将是存储技术的一个新的研究热点[3,4]。

体全息数据存储的文件系统在逻辑层应与现有的文件系统相容,这样,多维存储设备才能作为一个存储单元方便地加入到现有的存储体系中。因此,对文件系统的研究主要集中在存储体物理块的分配及与文件系统逻辑层的过渡上。本文主要研究文件系统存储块的分配策略。文件系统存储块的分配与存储介质的特性紧密相关,由于物理存储地址在表示方式上的复杂性,所以实际分配策略中通过将物理存储地址映射成逻辑块地址(Logical Block Address,LBA),按 LBA 进行存储块的分配。文件系统的存储分配策略所追求的目标是在访问文件时,存取装置移动的物理距离尽可能的短,这样就能有效地提高磁盘的访问效率。如果 LBA 能够与物理寻址相对应,那么实际分配存储空间时就需要对单个文件尽量分配连续的 LBA。

1 二维编址与相关定义

在磁盘文件系统中,一般用一维 LBA 来映射存储块的物理地址,一维的 LBA 与存储的物理地址是一一对应的。在磁盘系统中,虽然是二维存储方式,存取数据时需要两种运动,即盘片的高速旋转和磁头的来回移动,但由于盘片的旋转是磁盘启动后就开始的,并且一直保持不变,所以能够变化的只有磁头径向的来回移动。因此,一维的 LBA 能与实际的存储相对应,连续的 LBA 在物理上也是连续的[5]。对体全息数据存储而言,由于物理寻址时,需要通过多个机械的运动或光电参数的改变,一维的 LBA 显然不能很好地映射多维存储的物理地址,所以必须适当提高 LBA 的维数。在实际的研究中,采用的是二维 LBA。

定义 1 二维 LBA:将物理上的多维存储方式映射到逻辑的二维平面上,以二维逻辑地址简化物理寻址的多维性。

由于二维 LBA 的引入,文件系统的存储块分配的连续性也发生了很大的变化,这种变化与文件实际操作紧密相关。文件系统中大部分文件操作都需要对文件进行定位,单个文件的定位操作主要是两个方面:1)顺序存取文件时的顺序定位;2)移动到文件的任意位置的随机定位(比如,文件操作中

收稿日期:2006 - 02 - 13;**修订日期**:2006 - 04 - 30 **基金项目**:国家 973 规化资助项目(G1999030106)

作者简介:易法令(1969 -),男,湖北荆州人,副教授,博士,主要研究方向:图形图像处理、大规模数据存储、并行处理; 谢长生(1957 -),男,湖北襄樊人,教授,博士生导师,主要研究方向:大规模数据存储、嵌入式系统; 吴非(1975 -),女,湖北武汉人,讲师,博士,主要研究方向:大规模数据存储.

的 seek 操作)。对文件逻辑上的定位在物理上就是存取设备的移动,如果文件定位能够保证存取设备的连续性,显然能提高文件存取的效率。与这两种定位方式相对应的是两个方面的连续性,即顺序连续和随机连续。设一个文件是由 N 个逻辑块组成的有序序列,则可以给出如下定义:

定义 2　顺序连续:文件的第 i 个逻辑块与 $i+1(i+1<N)$ 个逻辑块存放在物理上连续的区域,则认为其是顺序连续。因为在文件数据块分配过程中,往往会出现顺序不连续的情况,所以需要用一个间隔度来度量顺序连续的程度。间隔度用逻辑上相邻的块在物理位置上间隔的存储块数来描述,间隔度越大,顺序连续性越差,间隔度为 0 则表示完全顺序连续。

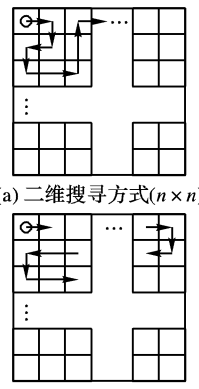
定义 3　随机连续:文件的第 i 个逻辑块与第 $j(i\neq j)$ 个逻辑块存放在物理上连续的区域,则认为块 i 与 j 之间是随机连续。文件的整体随机连续程度能反映文件随机定位存取的效率。同样文件的随机连续性也可以用间隔度来度量,不过间隔度则是指文件所分配的任意两个数据块在物理位置上间隔的存储块数。

在一维 LBA 方式下,随机连续与顺序连续是一致的,所以在进行存储块分配时,只是考虑其顺序的连续性。但在二维 LBA 方式下,对同一文件,完全顺序连续的两种分配方式由于其随机连续性不同,可能会导致完全不同的文件访问效率,下面对两种连续性进行具体分析。

2　连续性分析

2.1　顺序连续

顺序连续性主要是分析文件的相邻逻辑块的存储情况,对存储体而言,就是文件相邻逻辑块对应不同物理存储块之间的间隔度。由于文件在存放时,是按文件内容的顺序在存储体中搜寻空闲块并进行分配,所以采用不同的空闲块搜寻方式,直接影响到文件的顺序连续性。图 1 是在二维 LBA 方式下,存储块两种不同空闲块搜寻方式示意图,其中图(a)是一种根据二维存储的特点采取的二维扩散式搜寻方式;图(b)则是把二维存储一维化,用一维顺序方式搜寻,实际上就是磁盘文件系统的搜寻方式。设图 1 是一个 $n\times n$ 的二维矩阵,其中有 $m(1<m\leqslant n\times n)$ 个空闲块,为方便起见,设(0,0)为起始空闲块,现分析分别采用图 1 中一维和二维搜寻方式时,下一个空闲块与起始空闲块的间隔度,这两块实际上是存放相邻文件内容的两块。



(a) 二维搜寻方式($n\times n$)

(b) 一维搜寻方式($n\times n$)

图 1　空闲块搜寻方式

开始搜寻右边第一个,设 $N=n^2$,其搜寻概率为 $P(S_1)=1$,空闲块的概率 $P(E_1)=\dfrac{m-1}{N-1}$;则右边第二块的搜寻概率 $P(S_2)=P(S_1)-P(S_1)\times P(E_1)$,第二块为空闲块的概率 $P(E_2)=\dfrac{m-1}{N-2}$,第三块的搜寻概率 $P(S_3)=P(S_2)-P(S_2)\times P(E_2)$,依次类推:第 i 块的搜寻概率 $P(S_i)=P(S_{i-1})-P(S_{i-1})*P(E_{i-1})(N-i\geqslant m-1)$。

设第 i 块与起始块之间的间隔度为 D_i,则综合整个搜寻过程可以求得下一个空闲块与上一块的间隔度为:

$$D_{next}=\sum_{i=1}^{k}D_i*P(S_i)*P(E_i)\quad k\leqslant N-m+1 \tag{1}$$

由于 D_i 与搜寻方式有关,所以先分析式(1)后面部分。设:

$$DP_i=P(S_i)*P(E_i) \tag{2}$$

根据上述递推公式:

$$P(S_1)=1,P(E_1)=\frac{m-1}{N-1};\text{则:}DP_1=\frac{m-1}{N-1}$$

$$P(S_2)=P(S_1)-P(S_1)\times P(E_1)=\frac{N-m}{N-1},P(E_2)=\frac{m-1}{N-2};\text{则:}DP_2=\frac{(N-m)(m-1)}{(N-1)(N-2)}$$

……

依次类推可得出:

$$DP_i=\frac{(N-m)(N-m-1)\cdots(N-i-m+1)(m-1)}{(N-1)(N-2)\cdots(N-i)}\qquad i>2 \tag{3}$$

由式(3)可得:

$$\frac{DP_i}{DP_{i-1}}=\frac{N-i-m+2}{N-i}=1-\frac{m-2}{N-i} \tag{4}$$

从式(4)可以看出:随着不断向后搜寻,也就是 i 不断增加,DP_i 不断下降,并且下降的速度越来越快。

根据图 1 的搜寻方式,可以得出如下的间隔度表达式 D_i。

1)一维方式,坐标为 (x,y) 块与起始块之间间隔度为:

$$D_{(x,y)}=\begin{cases}y-1 & x\leqslant y\\ x-1 & x>y\end{cases} \tag{5}$$

根据搜寻顺序可以得出搜寻顺序 i(i 从 1 开始计数)与坐标之间的关系为:

$$\begin{cases}x=(\text{int})i/n\\ y=\begin{cases}i\%n & x\text{ 为偶数}\\ n-1-i\%n & x\text{ 为奇数}\end{cases}\end{cases} \tag{6}$$

2)二维方式,图 1(a)的搜寻方式可以看成是围绕着初始块的层次搜寻,那么,第 1 层是坐标为(0,0)的起始块,第 2 层是连续的第 2、3、4(共 3 块)与第 1 块的间隔度为 0,第 3 层是连续的第 5、6、7、8、9(共 5 块)与第 1 块的间隔度为 1,依次类推第 j 层共 $2*j-1$ 块,与起始块的间隔度为 $j-2$。搜寻顺序 i 与层数 j 之间的关系为:

$$j=\left\lceil\sqrt{i+1}\right\rceil \tag{7}$$

设 $N=16\times16$,m 分别为 64、128、192,根据表达式(4)~(7)可以得出图 2。图中 1_D_i 表示一维搜寻方式下间隔度与 i 的关系图,2_D_i 表示二维搜寻方式下间隔度与 i 的关系曲线。DP_i 曲线表示在 m 取一定值时,DP 值与 i 的关系。从图中可以看出,m 越大,则 DP 的初始值越大,但其值下降得越快。同时,m 越大,其对应的 D_{next} 求和区间越小。从图中还可看出:求和区间越小,一维的 $D_{next}(\sum_i DP_i*1\_D_i)$ 与二维的 $D_{next}(\sum_i DP_i*2\_D_i)$ 差别越大。所以当 m 值较大,也就是空闲块较多时,采用图 1(a)所示的二维搜寻方式更有利于保证空闲块分配的顺序连续性。对于 m 值较小的情况,需要采用另外的分配策略。

2.2　随机连续

对存储体而言,某一文件的随机连续性就是指文件所分配的存储块的集中程度,如果在文件操作中,对文件中任意位置的定位是等概率的,那么随机连续性可以用同一文件所分配的数据块两两之间的间隔度来度量,但由于不同文件所分配的数据块数目不同,因此在随机连续性的计算中还需考虑文件块的数量。设一个文件分配了 M 个数据块,则其随机连续性为:

$$C_s=\frac{1}{(M-1)M/2}\sum_{i=1}^{M-1}\sum_{j=i+1}^{M}D(i,j) \tag{8}$$

其中:$D(i,j)$ 指第 i 块与第 j 块之间的间隔度,$(M-1)M/2$ 是求和的间隔度的数目。现以图 1 所示的两种搜寻方

式分析随机连续性。设图 1 所示的二维区域数据块都是空闲的,则对同一文件按两种搜寻方式分配数据块,很显然,两种方式的分配结果都是顺序连续的,但其随机连续性则有很大的区别。图3 表示在 $n = 16$ 时,两种搜寻方式随着分配的存储块增加的随机连续性的对应值。从图中可以看出,采用一维搜寻方式,其随机连续性的值在达到一个极大值后以一种波浪式的方式前进;而二维方式则是随着分配块数的增加,其随机连续性的值缓慢地增加,最后整个二维区域充满后,两者的值相等。由于在体全息数据存储系统中,单个物理存储块的容量比较大,所以只有大的文件才分配很多的物理存储块。但是,在实际的文件系统中,大的文件只是占很少的一部分。如果二维区域很大的话,单个文件只可能占很少的行(或列)数。另外,如果按图 1(a)的二维方式搜寻,层数过大以后,其效果与一维搜寻类似,因此,在实际采用二维方式搜寻时,要根据存储块的组织结构采取适当的搜寻方式。

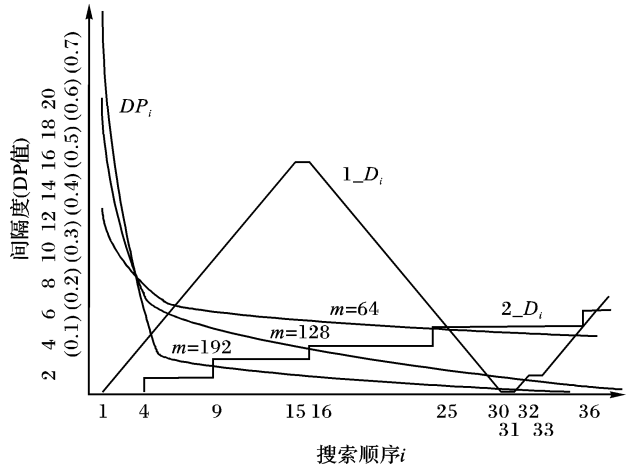


图2 两种搜寻方式的间隔度及 DP_i 与搜寻顺序 i 的关系

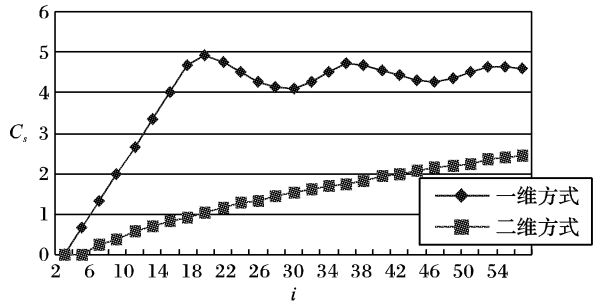
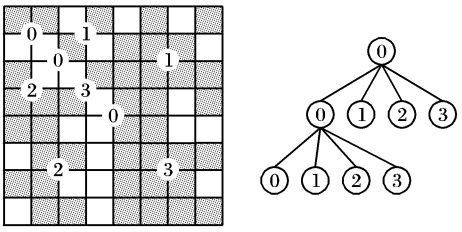


图3 两种搜寻方式的随机连续性趋势

3 分配算法

文件存放的连续性是以文件的基本操作为基础的,所以对连续性高的文件进行操作,无论是顺序还是随机存取都具有较高的效率。通过对文件系统空闲块分配的顺序连续性和随机连续性进行分析可以看出,单个文件数据块应存放在尽量集中的二维区域(随机连续),并且在分配时需考虑文件块的逻辑顺序(顺序连续)。因此,对二维区域进行组织管理时,既要考虑其区域性,又要考虑其顺序和层次性。四叉树是一种比较好的二维区域组织结构,它具有递归的特征,同时也可以对四个子树的顺序进行定义,符合文件系统二维空闲块分配的要求,下面给出分配四叉树的相关定义。

定义 4 分配四叉树:把 $2n \times 2n$ 的二维子区域在纵横两个方向进行剖分,构成分配四叉树,四叉树的节点中保存区域范围即左上和右下角的坐标,以及节点所包含的空闲块数。具体方法如下:1) 按 x 方向中分,记录相关坐标;2) 按 y 方向中分,记录相关坐标;3) 重复 1)、2) 直到某一区域阈值为止。四叉树子树的顺序根据具体的分配策略确定。



(a) 8×8二维区域划分 (b) 分配四叉树

图4 二维区域划分及其对应的四叉树

图 4 给出了一个 8×8 的二维区域划分及其对应的四叉树图,图 4(a)中空白块表示空闲块。把二维区域按分配四叉树的方式剖分,便可得到图 4(b)的四叉树图,图中的 0、1、2、3 编号是按其在剖分后的位置划分的。在实际分配过程中,四叉树的每个节点的四个子树是有顺序的,这种顺序就是搜寻的顺序,不一定是位置的顺序。

建立了存储块的分配四叉树后,就可以通过四叉树进行空闲块的分配。在分配的过程中需要考虑分配的单位。在体全息数据存储文件系统中,对单个文件的数据块分配可以分为三种情况,即区段、块和多块[4]。其中对大文件存储进行多块分配能充分发挥体全息数据存取速度快的特点。由于实际分配方式的多样性,决定了空闲块的分配不仅能以块和区段为单位,还可以以节点为单位进行多块成组分配。这里的节点既可以是叶节点,也可以从叶节点向上搜寻非叶节点,但向上搜寻的层次需要限制,因为向上的层次越多,对应的二维区域越大,二维区域增大必然导致所分配空闲块的连续性变差。搜寻时既可以采取从下至上的按固定的顺序进行搜寻,也可以从上向下,通过计算当前节点与索引节点所在叶节点之间的距离来进行搜寻,因为每一层四叉树节点上都保存有该节点对应二维区域的空闲块数。

以块或区段为分配单位进行分配时,一般从文件的索引节点所在的四叉树节点开始进行搜寻,如果不存在空闲块(区段),则向上搜寻其父节点直到搜寻到空闲块(区段)为止。成组分配多块时,分配时要考虑空闲区域的整体分布特性,成组分配空闲块算法如下:

1) 从文件索引块所在的节点搜寻;

2) 如果当前节点的空闲块数 $\geq N$,则转 4);

3) 否则,按序搜寻另外三个兄弟节点,如果其中存在空闲块数 $\geq N$ 的节点,则转 4);否则:

(a) 在规定的层次内搜寻父节点,如果父节点的空闲块数 $\geq N$,则转 4);

(b) 否则,按二维顺序搜寻邻近的叶节点,转 2);

4) 如果节点对应的二维区域空闲块比例超过 1/2,则按二维搜寻顺序分配数据块;

5) 否则,按二维矩阵区域邻域搜寻算法确定的顺序分配数据块;

其中 N 表示一次分配的存储块数,采用的是首次适应策略。在成组分配算法中,确定分配数据块的节点,也就是确定了分配数据块的二维区域。在二维区域中,如果空闲块数目较大,根据图 2 的分析,采用二维搜寻顺序分配数据块连续性较好;否则,可以采用一种邻域搜寻法来确定分配存储块的顺序。在图 5 的算法中,则是用空闲块的比例为 1/2 来作为采用两种算法的阈值。限于篇幅,对二维矩阵区域邻域搜寻算法不作具体分析。

4 实验评估

华中科技大学对体全息数据存储通道进行了比较深入的研究,在体全息数据存储通道基础上,构建了一个体全息数据

存储文件系统的原型系统[6]。在此系统中,通过模拟的文件操作序列对存储空间的分配算法进行"文件级"的测试分析[7]。

4.1　测试平台

测试平台在硬盘中划出一部分空间来模拟存储体,其中存储体的规模是5G字节,存储页(块)的大小是57 088字节,这也是我们的存储通道通过编、解码操作后每页实际的有效数据规模。每页共分8个区段,每个区段的大小为7 136字节,在索引节点中共7个区段指针,当分配数据长度超过7个区段时即需要分配1块。整个存储体共94 040页,元数据区共占了140页,按312×301进行二维分布。

比较分配策略主要分析存储文件数据块的连续性,从顺序连续和随机连续两个方面分析比较。连续程度可以通过间隔度进行度量,但是由于文件长度不同,所分配的存储块数目也不一致,为此需要定义一个平均间隔度。设文件分配了n个存储块(区段),也就是索引节点对应的N个地址,这N个地址按文件内容的先后构成一个序列$\{R_1,R_2,\cdots,R_i,\cdots,R_n\}$,其中相邻的两个地址$R_i$与$R_{i+1}$之间的间隔度用$D_i$表示,任意的两个地址$R_i$与$R_j$之间的间隔度用$D_{ij}$表示那么:

$$\text{顺序连续的平均间隔度为:}ADseq = \frac{\sum_{i=1}^{n-1} D_i}{n-1} \tag{9}$$

$$\text{随机连续的平均间隔度为:}ADstoc = \frac{\sum_{i=1}^{n-1}\sum_{j=i+1}^{n} D_{ij}}{n(n-1)/2} \tag{10}$$

其中,随机连续的平均间隔度与式(8)的随机连续性的表达式是一致的,因此能够反映文件分配的随机连续性。

为对比分析几种分配策略总体的顺序连续平均间隔度,还定义了一个每文件的平均间隔度:用DPF表示,其值为总的顺序连续间隔度与总文件数之比。

4.2　测试方式与结果分析

根据前面对空闲块搜寻方式的分析,一共测试了四种文件数据块的分配策略:第一种是按一维搜寻方式对数据块进行分配,其逻辑地址是一维的,通过一种映射方式将一维逻辑块地址与二维物理结构对应起来。后面三种都是采用与物理结构对应的二维逻辑块地址,但在存储块的组织上不是采用完全二维结构,而是把整个二维区域分成若干个标准的$2n*2n$区域(如图5所示),每个区域采用四叉树结构进行管理。其中,第二、三种方式都是采用32×32区域进行管理,第二种与第一种相比只是改变了搜寻空闲块的次序,即从一维的顺序搜寻变成了二维区域的扩散搜寻方式;通过这两种数据块的分配方式的结果进行比较可以确定搜寻方式对文件系统分配数据块连续性的影响。第三种分配策略二维区域组织与第二种相同,但在搜寻方式上进行了优化。第四种分配策略的二维组织方式与前两种有很大的不同:一方面单个的标准二维区域更大(按64×64进行二维区域组织);另一方面四叉树的叶节点不是单个存储块,也是一个标准的二维区域(4×4区域)。

测试系统通过体全息存储文件系统的原型系统对以上四种分配策略进行了测试,得到了如图6所示的结果,图中的横坐标表示四种分配策略,纵坐标表示间隔度值,其中:由于随机连续的平均间隔度$ADstoc$的值较小,所以将其值乘10以方便对比分析。测试数据验证了文件分配的连续性与空闲空间搜寻方式之间的关系。总体来说,可以得出以下结论:

1)与一维搜寻方式相比,二维搜寻方式有效地提高了文件分配的顺序连续性和随机连续性。

2)二维分配策略的空闲块搜寻顺序与分配四叉树的高度及叶节点的规模密切相关,可以针对不同的叶节点规模设计相应的分配算法。

3)在二维分配策略中,当叶节点也是二维区域时,与按块分配相比,按叶节点分配更有利于提高文件分配的随机连续性。

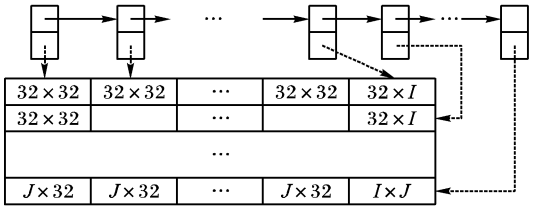


图5　二维分配策略的组织结构

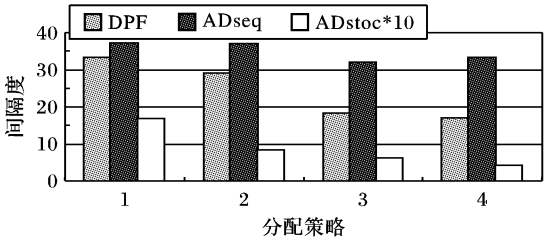


图6　四种分配策略结果对比

5　结语

在体全息数据存储的文件系统中,存储体空闲块的分配与体全息存储的寻址方式、存储内容的分布等重要存储特性紧密相关。根据存储体寻址的多维特性,设计了二维的逻辑块地址,并以此为基础,在文件数据块的分配中提出了"顺序连续"和"随机连续"的概念,并把它们作为衡量空闲块分配策略的重要指标。"顺序连续性"反映了文件内容的逻辑顺序与分配存储块地址的一一对应性;"随机连续性"则表明文件数据块的集中性或区域性。通过对一维和二维空闲块搜寻方式进行分析对比,从理论的角度证明在二维区域空闲块较多的情况下,按照二维搜寻的顺序分配存储块,文件存储的连续性更强。

通过相应的映射方式将物理地址映射到二维的逻辑区域,对二维区域存储块的组织采用四叉树的方式,以四叉树为基础设计了成组分配空闲块算法。在实际的体全息数据存储的通道上,构建了模拟的全息存储体,通过"文件级"的测试证明二维分配策略能有效地提高文件分配的连续性。

参考文献:

[1] CLAIRE G, YISI L, YUAN X, *et al*. Photorefractive materials and devices are becoming viable alternatives for information systems[J]. IEEE CIRCUITS & DEVICES MAGAZINE, 2003, 19(11): 17-23.

[2] 曹良才,何庆声, 尉昊斌,等. 10Gb/cm3小型化体全息数据存储及相关识别系统[J]. 科学通报 2004, 49(23): 2495-2500.

[3] Holographic memory becomes a reality[EB/OL]. http://www.cyberarmy.net/forum/hardware/messages/247834.html.

[4] YI FL, XIE CS, WU F, *et al*. The file system research based on the volume holographic data storage[A]. Proceedings of SPIE, Seventh International Symposium on Optical Storage[C]. 2005, 5966Q: 1-6.

[5] 徐小玲. IDE接口硬盘读写技术[J]. 电子科技大学学报, 2002, 31(6): 636-641.

[6] WU F, XIE C, HU D, *et al*. The design and research of high-speed channel for volume holographic data storage[A]. Proceedings of SPIE 2004[C]. 2004, Vol 5282: 1072-1076.

[7] ZHOU M, SMITH AJ. Analysis of Personal Computer Workloads[A]. Modeling, Analysis and Simulation of Computer and Telecommunication Systems, Proceedings of 7th International Symposium[C]. 1999. 208-217.