网络出版时间：2014-11-24 09:25
网络出版地址：http://www.cnki.net/kcms/detail/51.1196.TP.20141124.0925.025.html

一种空间高效的分布式数据存储方案

张柄虹[1]，张串绒[1]，焦和平[2]，张欣威[1]，李智伟[3]
(1. 空军工程大学信息与导航学院，西安 710077；2. 西北工业大学，西安 710072；3. 75150 部队，湖南 衡阳 421131)

摘　要：针对分布式数据存储中空间效率低，计算复杂度高等问题，基于 Jordan 矩阵和拉格朗日差值公式，提出了一种一般访问结构上高效的分布式数据存储方案。方案是计算安全的，空间利用率与理论安全的方案相比提高了 m^2 倍，每个存储服务器只需维护长度很短的秘密份额，就可以实现大数据的分布式存储。在数据存储过程中，存储服务器根据双线性对的性质计算并贡献影子份额，确保秘密份额的安全性。方案具有可公开验证性，有效防止了数据分发者与存储服务器的欺骗。最后对方案的正确性、安全性、拓展性，空间效率等进行分析，表明方案在分布式数据安全存储中具有很好的应用前景。

关键词：秘密共享；空间高效；Jordan 矩阵；分布式数据存储
中图分类号：TN918.1　　**文献标志码：**A

Scheme of high space efficiency in distributed storage

ZHANG Bing-hong[1], ZHANG Chuan-rong[1], JIAO He-ping[2], ZHANG Xin-wei[1], LI Zhi-wei[3]
(1. School of Information & Navigation, Air Force Engineering University, Xi'an 710077, China ; 2. Northwestern Polytechnical University , Xi'an 710072, China; 3. 75150 Troops, Hunan 421131, China)

Abstract: Focusing on the problem of low space efficiency and high computational complexity in distributed network, this paper proposed a scheme of high space efficiency in distributed data storage on general access structure based on the theory of Jordan matrix and the formulary of Lagrange differential. This scheme was computational secure, which improved the efficiency of m^2 times compared to those which are theoretical secure. Each storage server could share a long secret with each of them keeping a short share. In the process of data storing, storage server just computed and contributed the shadow according to the theory of bilinear pairing, assuring the safety of the share. The scheme was publicly verifiable, so cheating between secret distributer and storage server was avoided. In the end of the paper, it analyzed the validity, security, expansibility and space efficiency of the scheme. The result indicates that the scheme can be of good use in secure distributed storage.

Key Words: secret sharing; space efficiency; Jordan matrix; distributed storage

0 引言

分布式数据存储在日常生活工作中有着极为广泛的应用。小到个人电脑中的文件备份，大到公司重要资料的存储等都有涉及。近来，腾讯网盘、百度网盘、金山网盘等以云存储为技术背景的网络存储服务就是分布式数据存储的典型代表。

秘密共享是分布式数据存储中重要的安全工具。秘密共享的概念是由 Shamir[1]和 Blakley[2]分别根据 Lagrange 差值多项式和矢量空间的性质提出的。其中 Shamir 提出的方案是信息论安全的，攻击者的计算能力再强也无法获得信息。

但同时信息论安全的方案要求秘密份额的长度不得小于秘密的长度，即如果所需要的秘密大小是 $1M$，秘密在 n 个节点中存储，则需要存储的秘密将变成 nM。同时，我们知道 (t,n) 门限共享方案中，需要至少 t 个节点合作才可以恢复秘密，也就是说信息的有效传输效率为 $1/t$，当 $t=n$ 时，传输效率降为 $1/n$。文献[3]提出一种基于双线性对的可公开验证的多秘密共享方案，可以实现多秘密的分享，效率有所提高。但方案也是信息论安全的。在大数据的今天，实现信息论安全的秘密共享方案，存储量以及传输效率的劣势凸显。特别是在云存储中，为了实现数据的安全存储及恢复，避免单点失效等问题，往往由很多个云存储服务器共享秘密[4]。为了提高空间效率，计算安全的方案被提了出来。

Parakh 和 Kak [5]提出了一种计算安全的空间高效的秘密共享方案，他们的方案将 k 个子秘密作为多项式 $f(x)$ 的函数值，求出 $f(x)$ 的表达式。根据求出的 $f(x)$，继续计算新的点，最后将 n 个子秘密分发给 n 参与者保存。刘艳红等[6]在分析[5]所提出方案缺陷的基础上，提出了一种不需要安全信道的空间有效秘密分享方案，利用分组加密实现对任意长度秘密的共享，同时秘密份额只需要保持一个分组长度。方案在空间利用率上克服了文献[5]方案子秘密个数不得小于门限值，子秘密不得相关等限制。杨晓元[7]等提出了一种可公开验证的短份额 (t,n) 门限秘密共享算法，该方案有效抵抗统计攻击和任意少于 t 个恶意的参与者的合谋攻击，同时各参与者保存的份额很短，使得空间效率提高。文献[8]在杨晓元[7]等的基础上提出了攻击结构上的短份额秘密共享方案，每个参与者都可以验证秘密份额的

基金项目：国家自然科学基金项目(61272486；61103231)；陕西省自然科学基础研究计划面上计划(2011JM8012)
作者简介：张柄虹(1989-)，男，四川江油人，硕士研究生，主要研究方向为密码学与信息安全研究(zbh_1989ing@163.com)；张串绒(1964-)，女，教授，硕导，主要研究方向为密码学与信息安全；焦和平(1964-)，男，陕西西安人，高级教师，主要研究方向为应用数学；张欣威(1992-)，男，湖北襄阳人，硕士研究生，主要研究方向为密码学与网络安全；李智伟(1978-)，男，湖南衡阳人，在职研究生，主要研究方向为安全管理.

真实性，并且参与者只需要保存一个无需更新的秘密份额就可以共享一个大秘密。但他们的方案只适合于门限访问结构上的应用，在一般访问结构上的应用存在局限性。

本文基于 Jordan 矩阵和椭圆曲线上的双线性对，提出一种一般访问结构上高效的分布式数据存储方案。方案是计算安全的，空间利用率在[5][6][7]的基础上进一步提高。利用双线性对的性质，在数据恢复阶段，存储服务器只需贡献自己的影子份额而非秘密份额，因而保存一个秘密份额就可以实现多个秘密数据共享过程的实现。方案中哈希函数的应用，保证了存储服务器可以公开验证秘密份额的正确性，以防止数据分发者与存储服务器的欺骗行为。

1 基础知识

1.1 Jordan 矩阵[11]

设 A 为 $n\times n$ 的实矩阵，记为 $A\in R^{n\times n}$。则存在一个 Jordan 矩阵 J 与 A 相似

$$J=\begin{pmatrix} J_1(\lambda_1) & & & \\ & J_2(\lambda_2) & & \\ & & \ddots & \\ & & & J_r(\lambda_r) \end{pmatrix}$$

每个子块 $J_i(\lambda_i)$ 表示如下

$$J_i(\lambda_i)=\begin{pmatrix} \lambda_i & 1 & & \\ & \lambda_i & & \\ & & \ddots & \\ & & & 1 \\ & & & \lambda_i \end{pmatrix}$$

即存在一个可逆矩阵 P，使得 $P^{-1}AP=J$。

定理 1 设 $A\in R^{n\times n}$，J 和 P 分别为矩阵 A 的 Jordan 矩阵和相似变换矩阵。若 J 已知，求得 A 的可能性等价于猜测 P。

证明：设 J 对角线上的元素为 $\lambda_1,\lambda_2,...,\lambda_n$，由

$$\begin{aligned} &P^{-1}AP=J\rightarrow AP=PJ \\ &\rightarrow A(p_1,p_2,...,p_n)=(p_1,p_2,...,p_n)J \quad (1) \\ &\rightarrow \begin{cases} Ap_1=\lambda_1 p_1 \\ Ap_j=t_j p_{j-1}+\lambda_j p_j \end{cases},2\le j\le n \end{aligned}$$

其中，t_j 为 λ_j 正上方的元素，其值为 0 或 1。

由（1）可知：当 λ_j 已知，A 未知时，即便 $t_j=0$，由 λ_j 也无法推出 p_j 的任何信息。因而得 A 的可能性等价与猜测 P。

关于 Jordan 矩阵的更多知识，可参见文献[9]。

1.2 椭圆曲线离散对数问题（Elliptic Curve Discrete Logarithm Problem，ECDLP）[10]

令 $E(GF(p^m))$ 为有限子域 $GF(p)$ 上的椭圆曲线在 m 次扩域上的有理子群，P,Q 为群上的任意两点，已知整数 k，可以求得 $Q=kP$。则由 P,Q 和 $E(GF(p^m))$ 求出 k 是困难的。

1.3 双线性对（Bilinear Pairing）映射[9]

G 是阶为 q 的循环加法群，q 是一个大素数，P 是 G 的生成元，G_1 是具有相同阶 q 的循环乘法群，a,b 为 Z_q^* 中的元素。映射 $e:G\times G\rightarrow G_1$ 称为双线性映射，并且拥有以下性质：

（1）双线性：$\forall P,Q\in G,\forall a,b\in Z_q^*$，$e(aP,bQ)=e(P,Q)^{ab}$；

（2）非退化性：存在 $P,Q\in G$，使得 $e(P,Q)\neq 1$。

（3）可计算性：对于所有 $P,Q\in G$，总存在有效的计算方法计算 $e(P,Q)$。

2 一般访问结构上的分布式数据存储方案

方案需要一个公告板，用于存放一些公开量，只有数据分发者 $Dealer$（下文简记为 D）可以修改公告板上的信息，其他存储服务器只能阅读和下载。数据分发者负责系统参数的计算和发布，并将存储服务器公钥和与数据相关的信息公布在公告板上。方案分为三个阶段：初始化阶段、数据分发阶段和数据恢复阶段。

2.1 系统初始化

记 $U=\{U_1,U_2,...,U_n\}$ 为 n 个存储服务器的集合。令 G 是阶为 q 的循环加法群，q 是一个大素数，Q 是 G 的生成元。数据分发者 D 随机选择一个 $s\in Z_q^*$ 作为系统私钥，计算 $P_{pub}=sQ$ 作为系统公钥。每个存储服务器 U_i 随机选择自己的私钥 $a_i\in Z_q^*$，将 $P_i=aP$ 作为自己的公钥发送给数据分发者 D，并由数据分发者 D 确定 $P_i\neq P_j,i\neq j$。若 $P_i=P_j,i\neq j$，则要求存储服务器重新选择 $a_i\in Z_q^*(i=1,2,...n)$，直到 $P_i\neq P_j,i\neq j$。D 将 Q,P_{pub} 公布在公告板上。

访问结构表示为 $\Gamma=(\Gamma_1,\Gamma_2,...,\Gamma_t)$，$D$ 为访问结构 Γ 中的每个授权子集 Γ_i 选择 $r_i\in GF(q)(i=1,2,...,t)$ 作为 Γ_i 的标识，并将 r_i 在公告板上公布。

2.2 数据分发阶段

设共享的数据为 S，选择安全系数 m，将数据分成长度相等的 m^2 份，即 $S=s_1\|s_2\|...\|s_i\|...\|s_{m^2}$，可以通过分割填充技术确保 $|s_1|=|s_2|=...=|s_{m^2}|$。安全系数 m 的取值根据每个授权子集的存储服务器数量以及数据长度可自行设置。该方案同样适用于分享多数据 $S_1,S_2,...,S_k(k>1)$，只需通过分割填充技术将多数据分解使其满足前述的要求即可。数据分发者将数据共享于 n 个存储服务器中，从而使得只有 Γ 中任意授权子集 Γ_i 的存储服务器才能恢复数据；而非授权子集的存储服务器不能恢复数据。

（1）数据分发者 D 将数据 $S=s_1\|s_2\|...\|s_i\|...\|s_{m^2}$ 组成矩阵 A

$$A=\begin{pmatrix} s_1 & s_2 & \cdots & s_m \\ s_{m+1} & s_{m+2} & \cdots & s_{2m} \\ & & \cdots & \\ & & \cdots & \\ s_{m(m-1)+1} & s_{m(m-1)+2} & \cdots & s_{m^2} \end{pmatrix}$$

（2）计算 A 的 Jordan 矩阵 J 以及相似矩阵 $P=(p_1,p_2,...,p_m)$，其中 p_i 为 m 维列向量，p_i 的长度为 $|p_i|=\dfrac{|S|+|b|}{m^2}$，$|b|$ 为填充的数据长度。Jordan 矩阵 J 可以

用 β 表示如下，并公布在公告板上。

$$\beta=\begin{pmatrix}\lambda_1 & l_1\\ \lambda_2 & l_2\\ \cdot & \cdot\\ \lambda_s & l_s\end{pmatrix}$$

其中 $l_i(i=1,...,s)$ 表示每个子 Jordan 矩阵的秩。

（3）随机选择一个整数 a，并构造一个 m 次多项式

$f(x)=p_1+p_2x+p_3x^2...p_mx^{m-1}+ax^m$ （2）

其中 p_i 为相似矩阵 P 的第 i 列。对于 $i=1,2,...,m$，计算 $f(i)$。

（4）对 Γ 中授权子集 $\Gamma_j=(U_{j,1},U_{j,2},...,U_{j,|\Gamma_j|})$，计算 $C_{j,|\Gamma_k|}=e(a_{j,|\Gamma_k|}P_{j,|\Gamma_k|},P_{pub})$，其中 $a_{j,|\Gamma_k|}$ 为存储服务器的私钥，$P_{j,|\Gamma_k|}$ 表示 Γ_j 中的第 $k(1\le k\le|\Gamma_j|)$ 个存储服务器的秘密份额。$C_{j,|\Gamma_k|}$ 即为存储服务器 $U_{j,|\Gamma_k|}$ 的影子秘密。利用 Γ_j 的标识 r_j 及存储服务器影子秘密 $C_{j,|\Gamma_k|}$，计算公开信息 $F_j=f(r_j)\oplus C_{j,|\Gamma_1|}\oplus\cdots\oplus C_{j,|\Gamma_j|}$。

（5）取一个强 Hash 函数 H，计算每个存储服务器影子秘密 C_i 的哈希值 $H(C_i)$。

（6）将 $msg=(H,H(C_1),...,H(C_n),F_1,F_2,...,F_m,f(1),...,f(m))$ 公布在公告板上。

2.3 数据恢复阶段

任意授权子集 Γ_j 中的存储服务器利用他们的秘密份额以及公开信息可以恢复数据，同时在数据恢复阶段并不要求存储服务器贡献自己秘密份额，而只需要提供一个影子份额。不失一般性，取授权子集 $\Gamma_j=(U_{j,1},U_{j,2},...,U_{j,|\Gamma_j|})$ 中的存储服务器合作进行数据恢复：

（1）存储服务器 $U_{j,k}$ 根据公告板上的信息，利用自己的秘密份额 $P_{j,|\Gamma_k|}$，计算 $C_{j,|\Gamma_k|}=e(a_{j,|\Gamma_k|}P_{j,|\Gamma_k|},P_{pub})$，并发送给数据恢复者（数据恢复者可以是 Γ_j 中的成员，也可以是可信的第三方）。

（2）数据恢复者对收到的影子份额进行哈希计算，得到 $H(C_{j,|\Gamma_k|})$，并将结果与公告板上对应存储服务器的影子秘密的哈希值进行比较。若两者不相等，则表明存储服务器贡献的影子份额错误，数据恢复者要求存储服务器重新发送影子份额，直到提供正确的影子。

（3）数据恢复者根据存储服务器贡献的影子份额，以及公告板上所共享数据的信息，计算

$$f(r_j)=F_j\oplus C_{j,|\Gamma_1|}\oplus...\oplus C_{j,|\Gamma_j|}$$

（4）利用 msg 以及第（3）步的结果，数据恢复者可以得到 $m+1$ 个点 $(1,f(1)),(2,f(2)),...,(m,f(m)),(r_j,f(r_j))$，根据 Lagrange 差值公式可以得到

$$\begin{aligned}f(x)&=\sum_{i=1}^{m}f(i)\frac{x-r_j}{i-r_j}\prod_{k=1,k\ne i}^{m}\frac{x-k}{i-k}\\&+f(r_j)\prod_{k=1}^{m}\frac{x-k}{r_j-k}\\&=p_1+p_2x+p_3x^2...p_mx^{m-1}+ax^m\end{aligned}$$

（5）根据所求 $f(x)$ 的系数，可以得到 $P=(p_1,p_2,...,p_m)$。同时由公告板上的 β，得到 Jordan 矩阵 J。最后由 $A=PJP^{-1}$，可以得到秘密矩阵 A，从而恢复出所共享的数据 $S=s_1\|s_2\|...\|s_i\|...\|s_{m^2}$。

若分享的是多个数据，则只需根据对应的分割填充技术对数据矩阵 A 进行处理，从而恢复所共享的多个数据。

3 分析与讨论

3.1 正确性分析

本文提出的方案，首先将大数据 S 经过分割填充分解成 m^2 个小数据块。利用 Jordan 矩阵的特性，将得到的数据矩阵 A 进行转化。结合计算得到的相似矩阵 P，根据拉格朗日差值公式，得到一个 $m+1$ 的多项式 $f(x)=p_1+p_2x+p_3x^2...p_mx^{m-1}+ax^m$，数据分发者计算并将数据信息隐藏在公开信息中。在数据恢复阶段，存储服务器贡献自己的影子份额 $C_k=e(a_kP_k,P_{pub})$，计算可以得到各授权子集的标识信息 $f(r_i)$，再结合公告板上的信息，利用拉格朗日差值公式，数据恢复者就可以恢复数据 S。因而，该方案是正确的。

3.2 安全性分析

（1）对于秘密共享方案而言，可验证性是发现共享存储服务器欺骗的重要性质。本文提出的方案能够在数据恢复阶段，验证各存储服务器提供的影子份额的正确性。根据哈希函数的性质，由 $H(C_i)$ 求出 C_i 在计算上是不可行的。而要找到一个 $C_i^{'}\ne C_i$，使得 $H(C_i^{'})=H(C_i)$ 在计算上也是不可行的。任何存储服务器都可以根据公告板上的信息，验证存储服务器提供影子的正确性，防止了数据恢复者与部分存储服务器的勾结。

（2）一般访问结构的秘密共享方案，要求满足两个基本条件：

a、任一授权子集的存储服务器合作可以恢复秘密；

b、任一非授权子集的存储服务器合作不能恢复秘密。

本文提出的方案，基于 Jordan 矩阵理论和拉格朗日差值定理，要恢复 m 次多项式需要已知 $m+1$ 个满足多项式的点。从公告板上可以得到 m 个点 $(1,f(1)),(2,f(2))...(m,f(m))$，由式（2）可以知道，任一授权子集 Γ_i，存储服务器合作可以得到第 $m+1$ 个点 $(r_i,f(r_i))$，而非授权子集无法得到该点。

（3）基于哈希函数的特性，攻击者从公开信息 $H(C_i)$ 计算存储服务器拥有的影子份额 C_i 在计算上是不可能的。而即便攻击者在影子份额传输过程得到了影子秘密 C_i。由双线性对的性质以及椭圆曲线上的离散对数问题，可以知道，已知系统公钥 P_{pub}，由 $C_i=e(a_iP_i,P_{pub})$ 计算出 P_i 是不可能的。所以，存储服务器只需要保存一个秘密份额就可以在多个数据共享过程中分享多个数据。

3.3 性能分析

（1）共享大秘密

由文献[11]可以知道，现有的一般访问结构上的秘密共享方案[12]不能直接用来共享大秘密，否则系统效率将大大降低。在取模运算中，模值越大，计算复杂度越高。而在本方案中，将数据分割成 m^2 个子数据，利用 Jordan 矩阵的特性以及拉格朗日差值对数据进行共享。在一次共享过程中，存储服务器只需维护一个秘密份额就可以实现 m^2 个子数据的共享，提高了方案的效率。

对于共享短份额的多个数据，本文提出的方案同样适用，其原理与共享大数据分割后形成的短份额秘密一致。

（2）空间效率分析

方案在空间利用率上较文献 [5][6] [7]进一步提高。为了方便比较，我们取 Shamir 方案的空间利用率为 1。由表 1 可以看出，共享同样大小的数据，本文与刘艳红等[6]方案空间利用率是最高的，其他方案都受到门限的限制，同时也是由自己根据需要划分的子秘密块数决定的。但刘艳红等[6]方案是单一多项式的，而本文方案在一个数据 Jordan 矩阵中隐含了 m 个多项式，简化了数据的分享过程。

表一 空间利用率对比

方案	秘密划分块数	空间利用率
Shamir	无	1
杨晓元等	r^2 （r 为门限值）	r^2
Parakh 和 Kak	$k \geq r$ （r 为门限值）	k
刘艳红等	d （由自己设定）	d
本文	m^2 （m 由自己设定）	m^2

（3）可拓展性

本文提出的方案具有很好的拓展性，可以动态地增加和删除存储服务器而不需要太大的变化。

a、若有新成员 U_k 加入，则首先需要自己选择私钥 $a_k \in Z_q^*$，并计算公钥 $P_k = a_k P_{pub}$，并由数据分发者计算确保 $P_k \neq P_i (i = 1,2,...,n)$。不妨设新成员的加入授权子集为 Γ_k，若 Γ_k 为 U_k 加入原有授权子集 Γ_p 构成的新授权子集。这种情况下，授权子集的数量并没有改变，只是授权子集 Γ_k 中的存储服务器发生变化。数据分发者计算 U_k 的影子秘密 $C_k = (q, P_k, P_{pub})$，重新计算 $F_k = F_p \oplus C_k$ 并修改公告板上的相应信息即可。

若 Γ_k 为 U_k 加入数据共享方案后产生的新的授权子集，则数据分发者为新授权子集 Γ_k 选择 $r_k \neq r_i (i = 1,2,...,t)$ 作为其标识。并重复数据分发阶段的（4）（5）（6）步即可。

b、若有成员 U_k 退出，其退出并没有影响授权子集的数量，即只是 U_k 所在的授权子集 Γ_k 存储服务器数量发生变化。则只需计算 $F_k^{'} = F_k \oplus C_k$，在公告板上更新并删除 U_k 相应的公开信息即可。

若成员 U_k 的退出造成了授权子集数量的变化，则只需要在公告板上删除对应变化授权子集以及 U_k 的公开信息即可。

4 结束语

本文基于文献[7]中的秘密共享思想，提出了空间高效的分布式数据共享方案，该方案不仅克服了文献[7]只适用于门限方案的局限性，适用于一般访问结构，而且可以灵活地确定数据矩阵的分割值，并应用于多数据的共享。在防止存储服务器内部欺骗和外部攻击者窃取数据方面，利用双线性对的性质和哈希函数问题的难解性，保证了方案的安全性。通过对方案的分析，方案具有很好的拓展性，只需更改少量的公开信息，就可以实现存储服务器动态灵活地加入或退出。在分布式数据存储的实际应用中有很好的利用价值。

参考文献

[1] SHAMIR Adi. How to share a secret [J].**Communications of the ACM**, 1979, **22**(11): 612-613.

[2] BLAKEY G R. Safeguarding cryptographic keys [J].**AFIPS National Computer Conference**, 1979: 313-317

[3]张柄虹,张串绒,焦和平,张欣威,高胜国. 一种基于双线性对的公开可验证多秘密共享方案[J].空军工程大学学报,2014,15(4),83-87.

[4] MINOWA Tadashi, TAKAHASHI Takeshi. Secure distributed storage for bulk data [J]. **Lecture Notes in Computer Science**,2012,**7667**(1):576-583.

[5] PARAKH A, KAK S. Space efficient secret sharing [C]. Proceedings of the 4th Annual Compuer Science Research Conference. University of Oklahoma, 2009.

[6] 刘艳红,张福泰.不需要安全信道的空间有效秘密分享方案[J].计算机学报,2012,**9**(35):1816-1821.

[7] YANG Xiao-yuan, LIU Zhen, ZHANG Wei, GUO Dun-tao. A high efficiency Data Distribution Algorithm in Distributed storage[C].2009 Fifth International Conference on Information Assurance and Security,2009,**1**:627-630

[8] ZHAO Da-wei, Peng Hai-ping, WANG Cong, YANG Yi-xian. A secret sharing scheme with a short share realizing the threshold and the adversary structure [J].**Computer &Mathematics with Applications**,2012,**64**(4):611-615.

[9] 徐仲,张凯院,陆全等.矩阵论简明教程[M].2 版.北京:科学出版社, 2005.

[10] KOBLITZ N. Elliptic curve cryptosystems [J]. **Mathematics of computation** ,1987,**48**:203-209.

[11] PANG Liao-jun, WANG Yu-min. A new (t, n) multi-secret sharing scheme based on Shamir's secret sharing [J]. **Applied Mathematics and Computation**, 2005, **167**(2): 840-848.

[12] WANG Shiuh-Jeng . Direct construction of a secret in generalized Group oriented cryptography [J].**Computer Standards and Interfaces**, 2004, **26**(5): 455-460.