# 基于最优内插预测的科学数据压缩方法*)

吴国清[1,2] 陈 虹[2] 徐小文[1,2]
(中国工程物理研究院研究生部 北京 100088)[1]
(北京应用物理与计算数学研究所高性能计算中心 北京 100088)[2]

**摘 要** 在海量科学数据存储和传输压力愈来愈大的背景下,我们针对结构网格离散的科学计算数据,研究了基于二维9点、三维27点最优内插预测的科学数据压缩方法。数值实验表明该方法大大优于现有的压缩算法,可以较好地解决科学计算数据的压缩存储问题。
**关键词** 数据压缩,预测编码,科学数据,线性方程组

## Scientific Data Compression Method Based on Optimized Interpolate Predition

WU Guo-Qing[1,2] CHEN Hong[2] XU Xiao-Wen[1,2]
(Graduate School of Chinese Academic Engineering Physics, Beijing 100088)[1]
(High Performance Computing Center, Institute of Applied Physics and Computational Mathematics, Beijing 100088)[2]

**Abstract** Application in scientific computing often operate with large volume of output and input data, the space required for data storage and data transfer over communication networks are bottleneck in scientific computing. In the paper, we present a compression method based on optimized prediction operator which applied to structured scientific data. the proposed approach may be used both for lossy and lossless compression and is well suited for scalar fields. Numerical results indicate that our method successfully outperforms universal compression algorithm greatly.
**Keywords** Prediction operator, Scientific data compression, Linear equations

## 1 引言

随着对复杂现象的全空间、高精度、高分辨率的模拟仿真变为现实,由其产生的庞大的结果数据大量消耗着用户的存储资源和网络传输资源[6],对绝大多数高性能计算平台造成了巨大的压力。而现有的文本或图像压缩算法或工具(如gzip、lzo、winzip等)没有考虑到大规模科学数据的特点,压缩效果往往不尽人意,因此研究具有针对性的科学数据压缩方法具有很重要的理论意义和实用价值。

在一些科学研究项目中,基于预测方法和变换方法的科学数据压缩技术近年来陆续得到了发展。例如Vadim Engelson等人提出了基于整型数差分的科学数据压缩方法[1],其主要思想是首先将浮点数通过内存拷贝的方法转换成整型数,然后用高阶差分值代替原数据,然后再进行熵编码。美国重力波观测项目开发了一种基于随机数编码格式和小波变换树的科学数据压缩方法[2],并使用该项目的数据进行测试,相比于差分与gzip结合的方案,其压缩比大约提升20%。美国劳伦斯实验室针对$n$维纯量场数据提出的基于Lorenzo外插预测算子[3]的压缩方法,在L∞意义下甚至超过了基于小波变换的压缩方法,并且对内存的使用也非常小。在图像压缩领域取得巨大成功的JPEG2000标准也在组织补充三维体数据的国际压缩标准JP3D[5],其在科学数据压缩中的应用值得期待。

本文在文[3]的基础上,针对结构网格离散的科学数据,提出了一种基于最优内插预测的科学数据压缩方法,详细给出了二维9点和三维27点的最优内插预测算子的生成方法及反预测方法。我们将预测与反预测过程分别转化为矩阵计算问题,初步探索了预测编码的理论问题。在本文的数值实验中,我们使用数学模型数据和Lared S数据进行了测试。实验表明,我们提出的基于最优内插预测的科学数据压缩方法大大优于现有的通用压缩算法。

本文第2节首先给出了基于内插预测的科学数据压缩的整体框架,然后介绍了算法设计的详细内容;第3节中,基于实验结果对算法的优缺点进行分析;最后总结全文。

## 2 基于内插预测的压缩方法

### 2.1 压缩/解压缩流程

预测编码是数据压缩理论的一个重要分支。它根据数据之间的相关性,利用前面的一个或多个数据对下一个数据进行预测,然后对预测误差进行编码。如果预测比较准确,那么误差信号就会很小,可以用较少的码位进行编码,以达到数据压缩的目的。在科学计算中,函数及物理量通常是有连续性的,相邻网格点的数据之间有很强的相关性,可以通过某种插值算子利用相邻网格点的数据对当前网格点数据进行预测。基于内插预测的科学数据压缩流程框图如图1所示。

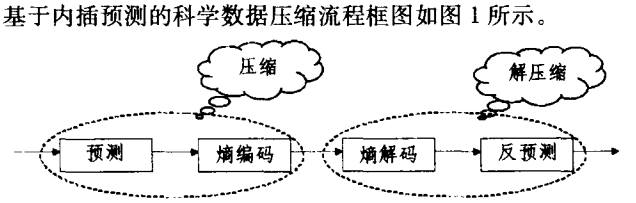



图1 基于预测的科学数据压缩/解压缩

### 2.2 预测算子的构造

预测算子的好坏直接关系压缩比的高低,下面以二维结构网格离散的科学数据为例,研究内插预测算子的设计方法。

*)中国工程物理研究院科学技术资金资助课题(No.20040658)。吴国清 博士研究生,研究方向为科学数据管理;陈虹 研究员,研究方向为科学数据管理与信息安全;徐小文 博士研究生,研究方向为并行计算。

假设二维结构网格规模为 $N=n\times n$，$M(i,j)$ 为当前待预测的网格点，$M(i-1,j-1)$、$M(i,j-1)$、$M(i+1,j-1)$、$M(i-1,j)$、$M(i+1,j)$、$M(i-1,j+1)$、$M(i,j+1)$、$M(i+1,j+1)$ 为其相邻网格点，距离点 $M(i,j)$ 路径长度为 1 的点的预测权重为 $\alpha$，路径长度为 2 的点的预测权重为 $\beta$，我们可以考虑如下形式的一般内插预测格式：

$$u'_{i,j}=\alpha(u_{i-1,j}+u_{i,j-1}+u_{i,j+1}+u_{i+1,j})+\beta(u_{i-1,j+1}+u_{i-1,j+1}+u_{i+1,j-1}+u_{i+1,j+1}) \quad (1)$$

其中，$i=0,1,\cdots,n-1$；$j=0,1,\cdots n-1$。由此得到预测误差：

$$e_{i,j}=\alpha(u_{i-1,j}+u_{i,j-1}+u_{i,j+1}+u_{i+1,j})+\beta(u_{i-1,j-1}+u_{i-1,j+1}+u_{i+1,j-1}+u_{i+1,j+1})-u_{i,j} \quad (2)$$

容易看出，可以将二维结构网格的预测误差写为如下形式：

$$e=Au \quad (3)$$

其中，$e=(e_{0,0},e_{0,1},\cdots,e_{i,j},e_{i,j+1},\cdots,e_{n,n})\in R^N$，$u=(u_{0,0},u_{0,1},\cdots,u_{i,j},u_{i,j+1},\cdots,u_{n,n})\in R^N$，矩阵 $A\in R^{N\times N}$，具体有如下结构：

$$A=\begin{bmatrix} \ddots & & & & & & & & & & \\ & \ddots & & & & & & & & & \\ & & \ddots & & & & & & & & \\ & & & \ddots & & & & & & & \\ & & & & \ddots & & & & & & \\ \cdots & \beta_{i,i-n-1} & \alpha_{i,i-n} & \beta_{i,i-n+1} & \cdots\alpha_{i,j-1} & -1_{i,j} & \alpha_{i,j+1} & \cdots & \beta_{i,i+n-1} & \alpha_{i,i+n} & \beta_{i,i+n+1} & \cdots \\ & & & & & & & & \ddots & & & \\ & & & & & & & & & \ddots & & \\ & & & & & & & & & & \ddots & \end{bmatrix}_{N\times N}$$

称 $A$ 为预测算子，$u$ 为原始数据，$e$ 为预测误差。这样，预测过程就可通过式(3)式实现。

下面讨论 $\alpha$ 与 $\beta$ 的选取，使预测算子有较高的预测精度。假设 $x$ 方向的步长为 $h_x$，$y$ 方向的步长为 $h_y$，分别将 $u_{i-1,j-1}$、$u_{i-1,j}$、$u_{i-1,j+1}$、$u_{i,j-1}$、$u_{i,j+1}$、$u_{i+1,j-1}$、$u_{i+1,j}$、$u_{i+1,j+1}$ 在中心点 $M(i,j)$ 使用 Taylor 公式展开，由式(1)即得：

$$\alpha(u_{i-1,j}+u_{i,j-1}+u_{i,j+1}+u_{i+1,j})+\beta(u_{i-1,j-1}+u_{i-1,j+1}+u_{i+1,j-1}+u_{i+1,j+1})=(4\alpha+4\beta)u_{i,j}+(\alpha+2\beta)h_x^2\frac{\partial^2 u}{\partial x^2}|_{i,j}+(\alpha+2\beta)h_y^2\frac{\partial^2 u}{\partial y^2}|_{i,j}+O(h_x^4+h_y^4) \quad (4)$$

当 $\begin{cases}4\alpha+4\beta=1\\ \alpha+2\beta=0\end{cases}$，即 $\begin{cases}\alpha=0.5\\ \beta=-0.25\end{cases}$ 时，有最高的 4 阶预测精度。

类似地，可以得到三维 27 点最优内插格式的预测算子，距离待预测点路径长度分别为 1、2、3 的网格点的预测权重 $\alpha$、$\beta$、$\gamma$：

$$\begin{cases}\alpha=0.5\\ \beta=-0.25\\ \gamma=0.125\end{cases}$$

此时预测算子有最高 6 阶预测精度。

最高预测精度的 9 点与 27 点最优内插预测算子的可用图 2 表示。

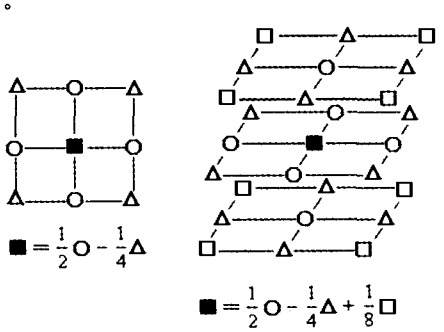



图 2　9 点与 27 点预测算子

当 $u_{i,j}$ 为边界点时，在外面延拓一层虚拟网格点，且假设虚拟网格点的值均为零，并且当 $|e_{ij}|<0.1\varepsilon$ 时，赋值 $e_{ij}=0$，这里 $\varepsilon$ 为反预测过程中，迭代停止的判断条件，即当 $\|r^{(k)}\|_2=\|e-Au^{(k)}\|_2<\varepsilon$ 时，反预测过程停止。

值得注意的是，在式(3)中由于 $A$ 是大规模的稀疏矩阵，因此我们在算法的具体实现中使用了稀疏矩阵的压缩存储技术(CSR)，目的是尽可能地减少内存使用开销。

### 2.3　反预测

解决了预测算子的设计之后，我们最关心的是如何进行反预测，与 Lorenzo 外插预测算子不同，我们提出的预测算子是隐式格式。由式(3)可知，其反预测过程即为给定 $A$、$e$，求解 $u$(由于我们在预测和反预测过程中使用相同的预测算子，因此预测算子不需要存储)，即求解如下线性方程组：

$$Au=e \quad (5)$$

对于大规模科学数据压缩问题，反预测过程中需要求解大规模的线性方程组。因此，线性方程组的高效求解对解压缩时间有重要的影响，可能成为整个压缩算法的瓶颈。求解线性方程组有直接法和迭代法两种。由于受计算复杂度和存储量的限制，并考虑到系数矩阵的稀疏性，通常采用迭代法。对于经典的迭代法，如 Jacobi 方法、Gauss-Seidel 方法等，由于收敛速度太慢，数值实验表明，不适合我们的问题。考虑到方程组(5)的系数矩阵 A 具有对称性、稀疏性的特点，并且往往是正定的，因此，我们考虑使用目前求解线性方程组最流行的方法之一，Krylov 子空间迭代法进行求解。其中，共轭梯度(CG)和广义最小残差(GMRES)方法是这类迭代法的典型代表。在我们的问题中，尝试使用了预处理的 GMRES 和 CG 方法求解线性方程组(5)，并使用 LASPACK 实现该过程。

### 2.4　算法分析

在该压缩方案中，预测过程只有数据在有限字长的计算机中进行运算所产生的计算误差，是近似无损的。在解压缩的过程中，根据用户的需要，设定反预测过程中所容忍的精度损失，即误差控制 $\varepsilon$。精度越高，反预测过程时间开销越大，当 $\varepsilon$ 足够小时，该压缩/解压缩过程对原始数据没有精度上的损失，该算法即为无损压缩，当然也可以降低精度要求以换取解压缩过程的时间开销。另外，如果仅仅要求有损压缩，不对数据精度损失做过高要求，我们还可以对预测过程所得到的预测误差进行量化以进一步提高压缩比。

本文研究了二维 9 点或三维 27 点的最优内插预测算子，前者对内存的使用开销比较小，而后者则是预测精度更高。对于规模庞大的数据集，我们可以根据需求来选择不同的预测算子。本文的测试实验中，我们使用二维 9 点预测算子来实现对三维数据集的逐页预测与反预测。

## 3 测试实验

我们使用了数学模型数据和物理数值模拟数据进行测试实验，比较其压缩比及压缩/解压缩时间开销，其中熵编码部分，我们使用 deflate 算法。

### 3.1 算例 1

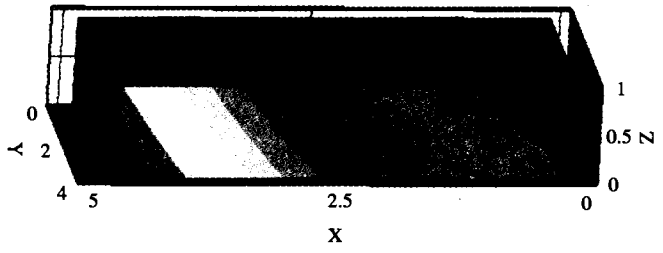



图 3 数学模型测试数据

数学模型测试数据：在三维长方体区域 $\Omega(x,y,z)=[0,5]*[0,4]*[0,1]$ 上计算函数 $f(x,y,z)=2^x+y^3+e^{-z}$ 离散值。将计算区域均匀剖分成 200×200×200 个网格，一次编号为 $(i,j,k)$：$i=1,2,\cdots,200$；$j=1,2,\cdots,200$；$k=1,2,\cdots,200$ 在网格点上的函数值为：$f_{i,j,k}=f(x_i,y_j,z_k)$，所有 $f_{i,j,k}$ 构成函数 $f(x,y,z)$ 在计算区域 $\Omega(x,y,z)$ 的离散解（如图 3）。以 64 位双精度二进制存储离散解，得到的原文件大小为 64005184(Byte)。

为了比较我们得到的二维 9 点最优预测算子的效果，对比了两种 9 点格式预测算子（其中预测算子(1)为具有最高四阶预测精度的预测算子）：

预测算子(1)：($\alpha=0.5,\beta=-0.25$)

$$e_{i,j}=0.5(u_{i-1,j}+u_{i,j-1}+u_{i,j+1}+u_{i+1,j})-0.25(u_{i-1,j-1}+u_{i-1,j+1}+u_{i+1,j-1}+u_{i+1,j+1})-u_{i,j}$$

预测算子(2)：($\alpha=0.15,\beta=0.1$)

$$e_{i,j}=0.15(u_{i-1,j}+u_{i,j-1}+u_{i,j+1}+u_{i+1,j})+0.1(u_{i-1,j-1}+u_{i-1,j+1}+u_{i+1,j-1}+u_{i+1,j+1})-u_{i,j}$$

其实验结果在表 1 和表 2 中给出。

表 1 数学模型数据压缩效果对比($\varepsilon$=1e-5)

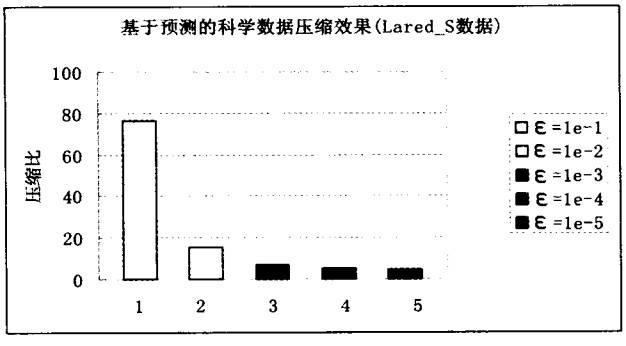
| 预测算子 | 压缩时间(s) | 压缩文件(Byte) | 压缩比 |
|---|---|---|---|
| 预测算子(1) | 3.26 | 186885 | 342.5 |
| 预测算子(2) | 4.77 | 5051972 | 12.7 |
| 无预测 | 14.78 | 52620338 | 1.2 |

表 2 数学模型数据解压缩时间(s)对比($\varepsilon$ =1e-5 )

| | ILU−GMRES | SSOR−CG |
|---|---|---|
| 预测算子(1) | 183.3 | 不收敛 |
| 预测算子(2) | 1292.4 | 932.9 |
| 无预测 | 4.18 | |

从压缩测试结果我们看到，预测算子(1)有非常高的预测精度，其所获得的压缩比远远超出使用预测算子(2)和没有使用预测算子的压缩方法，并且我们还可以看出，虽然预测过程会引入一些时间开销，但在对预测过程所生成的残差进行熵编码时，由于节省了 deflate 算法中字符串匹配的时间开销，导致总的压缩时间开销反而大大减小。从解压缩测试结果看，预测算子(1)的反预测过程，使用不完全 LU 分解(ILU)作为预条件子的 GMRES 方法(ILU-GMRES)时，迭代过程的时间开销远小于预测算子(2)，而使用以 SSOR 为预条件子的 CG 方法(SSOR-CG)时，迭代过程的时间开销都非常大，其中求解预测算子(1)的线性方程组时，迭代没有收敛。以上测试表明了我们计算得到的最优内插预测算子在压缩效果及时间开销方面的优势，以后我们均使用该最优内插预测算子及 ILU-GMRES 来实现对大规模科学数据的压缩与解压缩。

### 3.2 算例 2



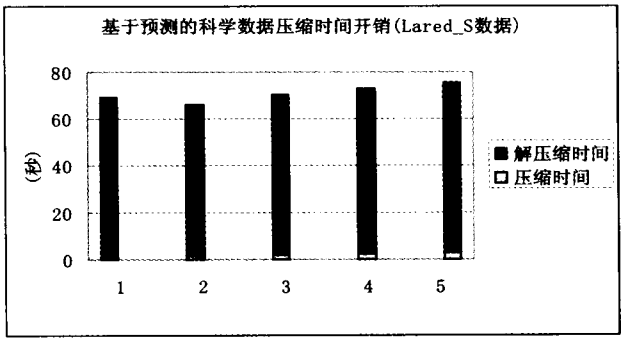
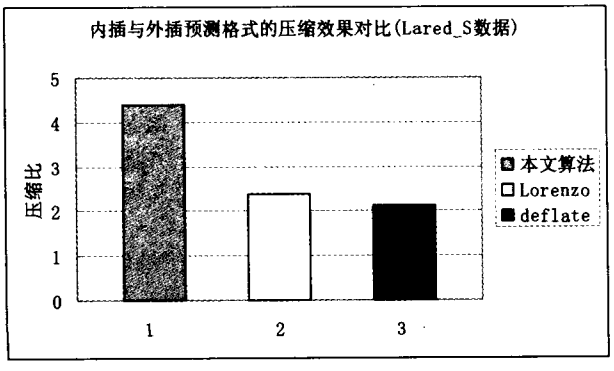




图 4 Lared_S 数据压缩测试结果



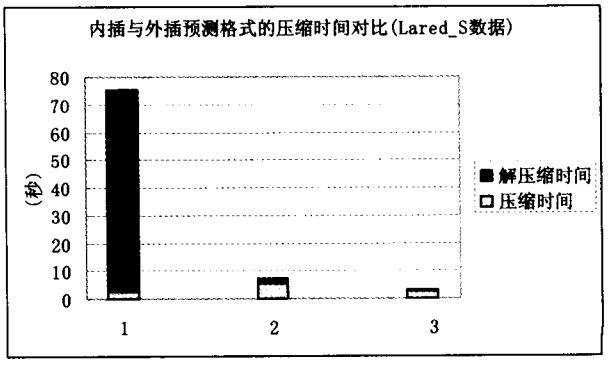




图 5 与 Lorenzo 预测算子的测试比较

(下转第 44 页)

(1)网络丢包率不同的情况

本文通过改变丢包率来对所提出的算法的性能进行验证,选择的丢包率参数分别为3%、5%、10%。随着网络丢包率的增大,必将导致视频测试序列的图像质量的下降。从表1中可以看出,采用HMP算法后视频测试序列的PSNR值相较FLP算法有所提高,并且随着丢包率的增大,这种性能上的差别就更加显著。这主要是由于HMP在打包时尽量使每个包包含一个相对独立的内容,减少包间的相关性,把由于前一包的丢失而对后续包所带来的影响减到最低,同时该算法还加入了对重要信息的保护,不会出现由于重要信息的丢失而造成收到数据也不能解码的情况。

(2)视频流比特率不同的情况

本文通过改变视频测试序列的量化参数(QP)进而改变视频测试序列的比特率,针对相同视频测试序列,本文选择的QP参数大小分别为24、20、16。从表1中可以看出,采用HMP算法可以很好地适应各种码流,这主要是由于当QP相对较小时,该算法组包采用以切片为单位,并尽量放入多个切片;当QP相对较大时,该算法组包采用以图片或帧组为单位,同时严格限制包长在MTU范围内,从而可以适应各种码流的视频序列,同时提高了视频流序列在网络上传输的效率。

(3)视频测试序列类型不同的情况

本文采用的视频测试序列都是标准的测试序列,选择了有场景转换的Foreman测试序列和无场景转换的News测试序列。针对两种不同类型的测试序列,在分别采用HMP算法后图像质量在性能上都相比采用FLP方法有所提高。由此可以看出,对不同类型的视频序列HMP算法都能起到一定的作用。

仿真结果显示本文提出的HMP算法,具有传输的高效性和丢包的鲁棒性,在丢包率较大的网络状况下仍获得良好视觉质量。

**结论** H.264实时视频传输是多媒体网络应用的重要课题。本文针对H.264编码特性提出了适用于H.264视频流实时传输的RTP传输模型,并在此基础上提出了混合模式组包算法HMP。实验结果证明,本文提出的HMP算法具有传输的高效性和丢包的鲁棒性,在丢包率较大的网络状况下仍获得良好视觉质量。

## 参 考 文 献


1 Schulzrinne H, Casner S, et al. RTP: A Transport Protocol for Real-Time Applications. RFC3550, July 2003

2 Turletti T, Huitema1 C. RTP Payload format for H. 261 Video Streams [S]. RFC2032, Oct. 1996

3 Zhu C. RTP Payload Format for H. 263 Video Streams [S]. RFC2190, Sept. 1997

4 Bormann C, Cline L, Deisher G, et al. RTP Payload Format for the 1998 Version of ITU2T Rec1H. 263 Video (H. 263+)[S]. RFC2429, Oct. 1998

5 Wenger S, Hannuksela M M, et al. RTP Payload Format for H. 264 Video. RFC3984, Feb. 2005

6 Wenger S. H. 264/AVC Over IP. IEEE Transaction on Circuits and System for Video Technology, 2003, 7(13): 645～656

7 Martini M G, Mazzotti M, Chiani M. Fixed-packet-length Transcoding for Error Resilient Video Transmission over WCDMA Radio Links. In: Proc. of Packet Video 2003, Nantes, April 2003


(上接第17页)

下面,我们以数值模拟程序Lared_S(辐射多群扩散流体力学界面不稳定性程序)计算的能量物理量第1000个时间步的数据为例,使用二维9点最优内插预测算子及ILU-GMRES来测试压缩/解压缩效果。

从表中我们可以看出,随着ε的减小,数据精度损失越来越小,压缩比也越来越小,而压缩与解压缩总的时间开销均呈递增的趋势。

### 3.3 算例3

Lorenzo预测算子[5]是外插预测算子,文[5]证明了它对于形如:

$$F(x,y)=ax+by+c$$

$$F(x,y,z)=ax^2+by^2+cz^2+dxy+exz+fyz+gx+hy+jz+k$$

的多项式数据具有最高的预测精度,下面我们用本文的内插预测算子与其做一对比。

从以上内插与外插的压缩测试结果可以看出,使用内插预测的压缩比大大优于使用外插预测时的情形,但是使用内插预测的解压缩时间开销远大于使用外插预测时的情形,因此二者各有优缺点。

**总结** 本文所得到的图2所示的内插预测算子具有较高的预测精度,基于该预测算子的科学数据压缩算法可以根据用户对科学数据的不同精度要求,灵活调整ε的值,以达到有损甚至无损压缩的目的,测试结果表明该方法取得了很好的压缩效果。此外,该压缩算法非常鲜明的一个特点就是其压缩时间远远小于解压缩时间,这为大规模科学数据的实时、在线压缩提供了有效手段。

## 参 考 文 献


1 Engelson V, Fritzson D, Fritzson P. Lossless Compression of high volume data from simulation. IEEE Computer Society, March 2000. 754～765

2 Klimenko S, Mitselmakher G. Lossless Compression of LIGO data. Technical Note, California Institute of Technology, 2000

3 Ibarria L, Lindstromy P, Rossignac J, et al. Out-of-core compression and decompression of large n-dimensional scalar fields. Eurographics, 2003, 22

4 Trott A, Moorhead R, McGinley J. Wavelets applied to lossless compression and progressive transmission of float point data in 3D curvilinear data. IEEE Computer Society Press, Oct. 1996. 385～388

5 Gamitoa M N, Dias M S. Lossless Coding of Floating Point data with JPEG 2000 Part 10. 2004

6 http://hdf.ncsa.uiuc.edu

7 Ibarria L, Lindstrom P, Rossignac J. Scientific data compression. http://www.llnl.gov/iscr/guests/students/FY05_posters/IbarraLawrence.pdf

8 吴乐南,徐孟侠. 数据压缩. 北京:电子工业出版社,2000

9 陈虹,夏芳,宋磊. 三维等离子体粒子模拟程序的数据模型和IO性能改进. 计算机工程与应用,2004,20:104～107

10 徐树方. 矩阵计算的理论与方法. 北京:北京大学出版社,2001