

Lossless Compression of Float-Point Data^{*}

Yi Sun¹, Qingguo Xu^{1, 2}, Xuelong Lin³

¹) Engineering and Computer Science, Shanghai University, Shanghai, China, 200072

²) Shanghai Key Laboratory of Computer Software Evaluating and Testing, Shanghai, China, 201112

³) College of Mechanical Engineering, Donghua University, 201620

Abstract—To solve lossless compression of float-point problem effectively, the construction of float-point, defined by standard IEEE754, lossless wavelet transform and lossless integer coding were studied. A design method of lossless compression of float-point based on lifted wavelet transform is presented. The float-point data is mapped to the integer creatively in the method, by analyzing the construction of the float-point. And then the integer data is transformed by lifted wavelet transform. Finally, the processed data is encoded by lossless integer compression method. The method is used to compress high precision float-point data lossless. The feasibility of the method is validated by practical application.

Keywords—lossless compression, float-point, lifted wavelet transform, integer compression

浮点型数据的无损压缩

孙毅¹ 许庆国^{1,2} 林学龙³

¹) 计算机工程与科学学院, 上海大学, 上海, 中国, 20072

²) 上海市计算机软件评测重点实验室, 上海, 中国, 201112

³) 东华大学机械工程学院, 上海, 中国, 201620

摘要为解决浮点型数据的无损压缩存储问题, 研究了 IEEE754 标准规定的浮点型数据的结构特点, 无损小波变换以及无损整数编码等算法。在此基础上, 提出了基于浮点型数据结构特点的无损浮点型数据压缩方法。该方法通过分析浮点型数据的结构, 创造性地将浮点数据映射为一个整型数据, 然后对整型数据进行提升小波变换, 最后采用无损整型压缩算法进行编码。该方法适用于高精度无损浮点型数据的压缩。通过试验证明该方法的效能。

关键字无损压缩、浮点型、提升小波变换、整型压缩

1. 前言

随着工业技术的发展, 各个行业的数据量越来越大, 尤其是高精度浮点型数据。在地质勘探、雷达探测以及仿真等实际工程以及科学计算中都需要存储和处理海量的精确数据。同时越来越多地需要实时数据, 或者需要将大量数据传输到服务器做后续的处理分析, 这要求在同样的时间和传输能力的条件下, 提高信息的传输量。随着大量的高性能低功耗的通用以及嵌入式处理器的广泛应用, 对于

原始数据进行适当复杂的压缩计算变得容易。

数据压缩根据文件压缩与解压缩前后信息是否丢失分为有损压缩和无损压缩。对于可以容忍一定程度数据丢失的情况下, 有损压缩可以达到高压比^[1]。但是随着工业中对数据分析精度的提高, 有损压缩不能满足要求。目前通常采用的手段是借助第三方软件进行数据压缩, 例如 Winrar、Zip 等; 或者采用较为简单的压缩标准保存数据, 例如国家地震台网使用的 SEED 标准 (Standard for the

^{*}Supported by National Natural Science Foundation of China(NSFC) under grant No. 60970007 and 61073050, the National Grand Basic Research Program(973 Program) of China under grant No. 2007CB310800, the Natural Science Foundation of Shanghai Municipality of China under Grant No.09ZR1412100, Shanghai Leading Academic Discipline Project (Project Number: J50103), Key Laboratory of Science and Technology Commission of Shanghai Municipality under Grant No. 09DZ2272600.

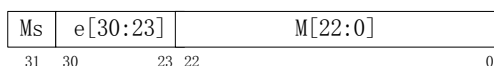


图1 IEEE754标准中float型数据存储格式

Exchange of Earthquake Data, 地震数据交换标准格式)。前一种无损压缩方式效率较高, 缺点是将文件压缩后失去了数据文件本身的结构特点; 后一种压缩方式效率较低, 但是可以清楚地了解数据文件的类型, 且对于数据的压缩也只是局限于整数的无损压缩, 不涉及浮点数无损压缩^[2]。

2. 浮点型数据无损处理

2.1 分析浮点型数据格式

目前通用计算机中使用的浮点数表示方法是 IEEE (Institute of Electrical and Electronics Engineers, 电子电气工程师协会) 1985 年制定的 IEEE754 (IEEE Standard for Binary Floating-Point Arithmetic, ANSI/IEEE Std 754-1985) 二进制浮点运算规范 (下称“标准”)。标准中规定了单精度、双精度以及双精度扩展等三种格式, 本文以 32 位单精度 float 型为例详述浮点型数据的压缩算法。

标准中规定了单精度格式由三个字段组成: 23 位尾数部分 M ; 8 位移码指数 e ; 以及一位符号位 M_s , 即图 1 所示。根据规定, 一个单精度数值的大小使用科学计数法表示为:

$$f = (-1)^{M_s} \cdot M \cdot B^e \quad (1)$$

关于标准的说明:

- 1) M_s 是数值的符号位, 表示数值 f 的正负;
- 2) M 表示尾数, 是用二进制整数表示的有效数字的小数部分, 即去掉最高位 (个位) 的 1 之后剩下的部分;
- 3) 采用规格化浮点数表示, 规格化是指浮点数的科学计数法表示有效数字的最高位不为 0。这样每一个有效数字的第一位都是 1, 所以第一个 1 可以省略。这样做同时也隐式地增加了 1 位有效数字的长度;
- 4) e 表示阶码, 采用无符号的整数移码表示。使用大于 $2^{n-1}-1$ 的数表示正阶码, 用小于 $2^{n-1}-1$ 的数表示负阶码, 在单精度浮点数中 $2^{n-1}-1$ 的值为 127;
- 5) B 是 e 的底数, 标准规定为 2。

由单精度浮点数的格式中不难发现: 一个浮点数, 其表示范围主要由阶码决定, 其精度则由尾数决定。采用 32 位长度表示数字, 不论是定点数还是浮点数, 最多能表示的数字的个数只能是 2^{32} 个, 采用浮点数表示增大了表示范围, 不会增加数字的个数, 只是数字间的间隔变大了。基于以上分析, 这里不妨将一个 32 位单精度的浮点数与一个同样二进制表示的带符号 32 位整数看作一个一一映射。通过公式 (1) 可以得出的以下结论:

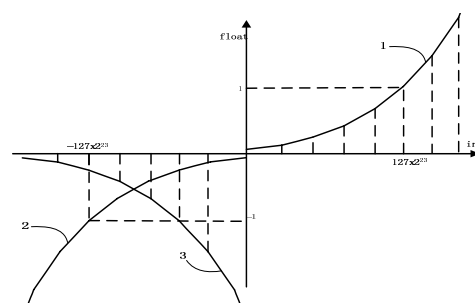


图2 Int型与Float型数据映射方程示意图

- 1) 第一位的符号位 M_s , 无论是浮点型还是有符号整型, 都表示为数值的正负;
- 2) 可以将公式看成 $float = func(int)$ 的一个函数式, 将符号位 M_s 与阶码 B^e 的乘积作为系数 k , 并且在保持阶码 e 不变的条件下, 即 int 的高 9 位不变的情况下, 可以得到 $float = k \cdot M$ 。这就是说, 在一定范围内, $float$ 数值的大小与对应的同样二进制的 int 型数值的后 23 位成正比, 即只与尾数 M 有关;
- 3) 当随着尾数 M 增大, 需要进位时, 浮点数的阶码部分递增, 即对应 int 型高位加 1, 得到在另一个数值区间内的新的函数式: $float = k' \cdot M$ 。容易得出 $k' > k$ 。所以, 在不考整型补码表示的条件下, 可以大致得到浮点型数据与整型数据一一映射的函数示意图, 如图 2 中曲线 1、2 所示。

由于整数在计算机中使用补码表示, 即正整数的表示法与原码相同, 负整数用其对应的绝对值按位取反加 1。所以, 现实情况是不加处理的使用内存拷贝, 对于负浮点数, 映射为同样二进制的相应整数的示意图应为图 2 中曲线 3 所示。为此, 先判断浮点数的正负, 如果为正数, 将原浮点数对应的二进制内存拷贝到一个整型变量中保存, 则这个整数就是映射后的值; 如果为负, 则将原浮点型数据中除去符号位以外的各位按位取反后, 再进行内存拷贝, 这样便得到图 2 曲线 2 对应的映射。

由图 2 可得, 浮点型就是将原来的整型的间隔扩大了, 并在不同的区间乘以不同的系数 k 。随着表示的数字的增大, k 随之增大。

2.2 提升小波处理

由于信号有局部相关性, 某一点的信号值可以由相邻的信号值经过适当的预测算子预测出来, 尤其是各种数字信号处理中的波形数据、图片以及视频等。其预测误差就是信号的高频信息 (即图像的细节信号), 这个过程就是预测环节。预测环节得到的高频信息通过更新算子来调整信号的下抽样, 得到低频信息 (图像的近似信号), 这个过程就是更新环节。在整个过程中, 更新环节称为原始提升,

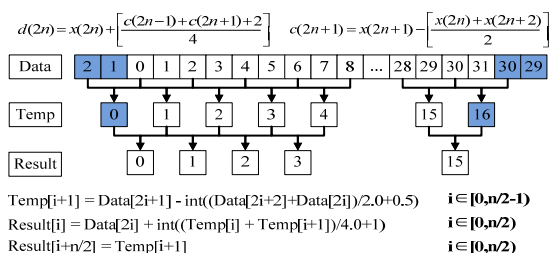


图3 提升小波变换算法

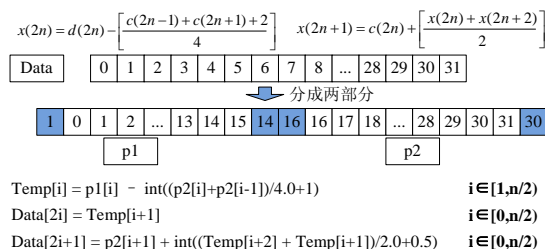


图4 提升小波变换的解压缩算法

而预测环节称为对偶提升^[3]。提升的基本步骤如下：

- 1) 分割，把原始数据 $x(n)$ 分解成两部分，奇数列和偶数列，即 $xe(n) = x(2n)$, $xo(n) = x(2n+1)$;
- 2) 预测，利用 $xe(n)$ 预测 $xo(n)$ ，即 $d(n) = xo(n) - p[xe(n)]$ ；式中， p 为预测算子， $d(n)$ 表示预测误差。在提升理论中， $d(n)$ 也称为小波系数（对应高频分量），当 $p[xe(n)]$ 越接近 $xo(n)$ ， $d(n)$ 就越小^[4]。
- 3) 更新，利用 $d(n)$ 更新 $xe(n)$ ，即 $c(n) = xe(n) + u[d(n)]$ ，式中， u 为更新算子。在提升理论中， $c(n)$ 也称尺度系数（对应低频分量），是原始数据的一个粗糙近似。

小波提升是一个完全可逆的过程，其反变换和正变换结构对称，算子符号相反，由此可以保证提升变换是一个可逆变换，可以实现精确重构。提升方案可以实现原位计算和整数提升，并且中间结果是交织排列的。选用 w5/3 小波，则变换原理及公式如图 3 所示^[5]。逆变换实现算法原理及公式如图 4 所示。

3. 无损压缩编码

对于采用提升小波变换处理后的数据，除第一个数字之外，其他数值都是细节系数，且大多是都很小且重复性很大的整数。基于此，采用改进的行程编码与 ECMA-335 中采用的整数压缩算法混合编码。

3.1 改进的行程编码

行程编码是采用“字符+记号（S）+出现次数”的形式来代替连续出现的字符实现压缩。由于不论连续出现多少个重复字符，该算法都将其转变为 3 部分，所以只对于重复出现 3 次或 3 次以上的字符采用该算法进行编码。

表 1 ECMA-335 标准中整型压缩区段

区段	字节数	掩码	二进制形式
[00000000h, 0000007Fh]	1	80h	0NNNNNNN
[00000080h, 00003FFFh]	2	C0h	10NNNNNN NNNNNNNN
[00004000h, 1FFFFFFFh]	4	E0h	110NNNNN NNNNNNNN

这里需要说明的是，其中的“记号”，这里用 63（二进制 111111）来表示。例如，(3)(2)(1)(1)(1)(1)(7)(6) 编码为 (3)(2)(1)(63)(5)(7)(6)。同时需要注意几个规定：1) 连续的字符数不能超过 62，否则应当断开，后面的再使用同样的 3 个字节存储；2) 文件中 ASCII 码值为 63 的数值，采用两个 63 来表示。

3.2 ECMA-335 整数编码^[6]

ECMA-335 标准即公共语言架构（Common Language Infrastructure, CLI）。该标准可以将不同的高级语言编写的程序在不同的系统环境下能够正确运行。其中描述了一个整数压缩算法，该算法概括的说是将整数的整个取值范围划分为几个区段，而整数数值根据其所在的区段不同，分别放置在 1、2 或 4 个字节中。

在为带符号整数确定需要用多少个字节来存放压缩值时，需要首先取得原整数的“准绝对值”，即对负数进行按位取反（而不是数学求负），然后将这个“准绝对值”左移 1 位（为符号位空出最低位），再按照表 1 列出的区段取得最终占用的字节数。

“区段”列出了每个区段的最小值（含）和最大值（含）；“字节数”列出压缩后的整型所需的字节数；“掩码”列出了用于解压缩时判断数值区段的掩码值：如果与掩码 80h 进行 &（按位与）操作后的结果为 0h，则整数值占用 1 字节；如果首字节与掩码 C0h 进行 & 操作后的结果是 80h，则压缩后的整数值占用 2 字节；如果首字节与掩码 E0h 进行 & 操作后的结果是 C0h，则压缩后的整数值占用 4 字节；

“二进制形式”列出了压缩结果的二进制形式。其中的“1”和“0”都是固定值，而“N”则表示实际整数值的有效位。

图 5 表示 ECMA-33 标准中对带符号二进制整数进行压缩的流程：

- 1) 由于计算机中整数是使用补码表示，所以须将负数转化为对应“准绝对值”，并且将符号位移动到最末一位保存。

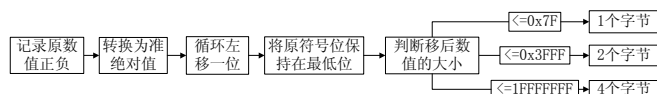


图 5 ECMA-335 标准压缩流程

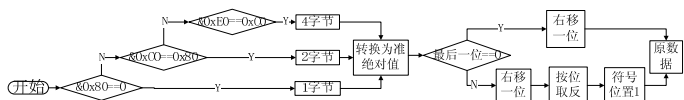


图 6ECMA-335 标准压缩流程

- 判断 1) 步中得到的数字区段，将数值按照表 1 分段，分别使用 1、2、4 三种不同长度的字符数组存放原数值；
- 按照表 1 中的掩码对上步中的最高位置位，用于读取数据时判断存放的长度。

经过以上的 3 步，一个整型数据就按照大小存放在 1、2 或者 4 个字节中。解压缩算法过程相反，如图 6 所示：

- 从文件中读取一个单字节的数值，将这个单字节数值以此与表 1 中的掩码进行按位与计算，如果首字节的二进制形式型如 0NNNNNNN，则采用 1 个字节存放整数，原整数值等于 $b[0]$ ；如果首字节的二进制形式型如 10NNNNNN，则采用 2 个字节存放整数（字节值依次为 $b[0]$ 、 $b[1]$ ），原整数值等于 $(b[0] \& 0x3F) \ll 8 \mid b[1]$ ；如果首字节的二进制形式型如 110NNNNN，则采用 4 个字节存放整数（字节值依次为 $b[0]$ 、 $b[1]$ 、 $b[2]$ 、 $b[3]$ ），原整数值等于 $(b[0] \& 0x1F) \ll 24 \mid b[1] \ll 16 \mid b[2] \ll 8 \mid b[3]$ ；
- 按照步骤 1) 可得到移位后的“准绝对值”。该数值的末位保存着原数据的符号位，保存其数值；
- 将步骤 2) 中的数值右移一位，空出最高位；并将符号位相应的置位，则得到原数值。

4. 试验结果分析

首先，验证本文所述的浮点型数值与整型数值之间的映射关系。取 $[-999.999, 999.999]$ 中间隔为 0.01 的浮点数，使用文中所述的方法映射为整型，然后将所有的点描绘在坐标轴中，即为浮点型与整型映射的方程示意图，如图 7 所示。从图中可以得到结论：当数据绝对值在 $[0, 1]$ 区间内，浮点型数据增加一点，整型数据增大很多，在 $(1, \infty)$ 区间内，随着浮点型数值的增加，整型的增加量趋于变小。其次在实际运用中，本文采用含有 24 道波形，每道 1024 个采样点的 SEG-2 标准的地震数据文件进行试验。原 SEG-2 文件的大小为 135k，分别采用不同的峰值范围以及

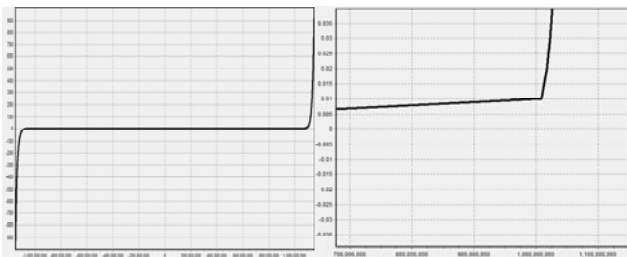


图 7Float-Int 映射示意图(左) 右下角放大以后示意图(右)

表 2 试验结果分析

波峰范围 (增量范围 0.0001)	[-1,1]	[1,10]	[10,1000]	[1000,∞)
压缩比	1.3:1	2.8:1	3.1:1	3.6:1
增量范围 (波峰范围[10,1000])	<10	<0.1	<0.01	<0.0001
压缩比	1.7:1	2.7:1	2.9:1	3.1:1

不同增量范围的数据进行多次实验，结果如表 2 所示：在相邻数据增量范围一定的条件下，浮点数的波峰范围在 $[-1, 1]$ 区间内，压缩后的压缩比较低，为 1.3:1。波峰范围大于 1000 时的浮点型压缩比较大为 3.6:1；在数据的波峰范围一定的条件下，相邻数据增量范围在 10 以内时，压缩比最小为 1.7:1，当相邻数据增量范围在 0.0001 以下的时候，压缩比最大为 3.1:1。

这一现象正好符合图 7 所示的浮点型与整型的映射关系：相邻数据增量的范围较大，或者如果浮点型数据的绝对值小于 1 的情况下，浮点数变化时，对应整型的变化值较大，此时的提升小波变换预测阶段不够精确，产生的小波系数较大，压缩比较小。如果数据变化范围较小，或者浮点数的绝对值大于 1 的情况下，浮点型数据变化时，映射成的整型数据的变化量趋于减小，则提升阶段的预测趋于准确，产生的小波系数较小，压缩比较大。

参考文献

- [1] Feng Zhanlin, Zhan Xuegong, Li Yanda. "Some practical aspects of seismic data compression based on wavelet transform". *Journal of Tsinghua University (Science and Technology)*, v.41, n.4/5, 2001:170-173.
- [2] Luo Xinheng, Zhang Zhe, Wang Chunming. "A compare of SEED data compression ratio". *Seismological and Geomagnetic Observation and Research*. 2004, v.24, n.4: 41-47, 14-19.
- [3] Swedens W. "The lifting scheme: a new philosophy in biorthogonal wavelet constructions". *Proceedings of SPIE*, 1995, 2569, pp.68-79.
- [4] Giridhar Mandyam, Neeraj Magotra, Wes McCoy. *Lossless Seismic Data Compression using Adaptive Linear Prediction*.1996.
- [5] ISO/IEC JTC1/SC29/WG1. *Coding of still pictures*, 2000-03-16/2005-04-11.
- [6] ECMA. *ECMA-335--Common Language Infrastructure (CLI) 4th Edition*. June 2006.