曲线数据压缩算法研究与应用①

尹志喜　甄国涌 (中北大学 电子测试技术国家重点实验室 山西 太原 030051)

摘　要： 在数据分析软件中，随着数据量的增加，导致数据分析的难度增加，制约了软件系统性能的进一步提高。在分析了以往数据压缩算法的基础上，提出了一种改进的数据压缩算法，通过该算法对数据进行压缩，可以保证高压缩比和低失真率参数，压缩后的数据量远远小于数原始数据量，从而可以有效的降低分析软件后续的工作量，提高分析软件的性能，尤其在数据量大和实时性要求比较高的应用中，该算法的优点更加突出。

关键词： 数据分析；数据压缩；海量数据；压缩比；失真率

Study and Application of a Curve Data Compression Algorithm

YIN Zhi-Xi, ZHEN Guo-Yong

(National Key Laboratory For Electronic Measurement Technology, North University of China, Taiyuan 030051, China)

Abstract: An increase of dada makes data analysis difficult and the systems improvement in performance is hence limited. This paper puts forward an improved data compression algorithm on the basis of researching former algorithms. This algorithm can efficiently compress data, reduce the workload of the system and improve its performance especially in dealing with mass data and real-time applications.

Keywords: data analysis; data compression; mass data; compression ratio; distortion rate

在测量系统中，为了精确测量一些信号，需要对传感器输出信号进行高速、高分辨率采样[1,2]。随着问题复杂度的提高，需要分析处理的数据量也越来越大，特别是高分辨率传感器的应用，使得需要分析的数据量正在呈几何级数增长[3]。有效的处理这些海量数据，对计算机的性能和应用软件的设计提出了更高的要求。对于海量数据的处理问题。日前有许多专家从不同侧面进行过研究。例如在数据压缩方面，已经有很多成熟的方法；在数据库以及数据仓库方面，也有一些完善的理论和方法。本文提出了一种改进的曲线数据压缩算法，该算法实现方法简单，可以有效的对海量数据进行分析和处理，满足多种情况下的实际应用需求。

1　Ramer-Douglas-Peucker (RPD)算法分析

RPD 算法的理论依据是 Attneave 的关键形状点(critical shape points)理论[4]，即曲线上的某一些关键点与另一些点相比包含更为丰富的信息，这些关键点足以表达曲线的形状特征。因此，RPD 算法是通过保留关键点删除次要点来达到曲线数据压缩的目的。该算法的基本思想是：

1) 对于数据点集合 $p_1, p_2 \cdots\cdots p_n$，设 $A = p_1$ 和 $B = p_n$，用虚线段连接 AB；

2) 在 AB 范围内的数据点集合中寻找与线段具有最大距离的点，记为 C；

3) 判断 C 点到 AB 的距离是否小于阈值 δ，若否，则设 $B = C$，执行步骤 2)；

4) 判断 B 是否到达 p_n，若否，则将 A 按原顺序放入特征点集合，在设 $A = C$、$B = p_n$，用虚线段连接 AB，执行步骤 2)；

5) 将 A、B 做为特征点，算法结束。

图 1 是一个典型的利用 RPD 算法压缩数据的例

① 基金项目:国家自然科学基金(50535030)
收稿时间:2009-06-16

子。该算法编程简单，对于弯曲起伏较小的数据，速度较快，如直线，只需找一次最大距离点就可得到结果。其缺点是对于较复杂的数据，重复循环判断的次数多，造成速度慢；必须事先知道全部数据点，然后从后向前每提取一个特征点几乎都要对全部数据进行一次扫描，比较耗费时间。

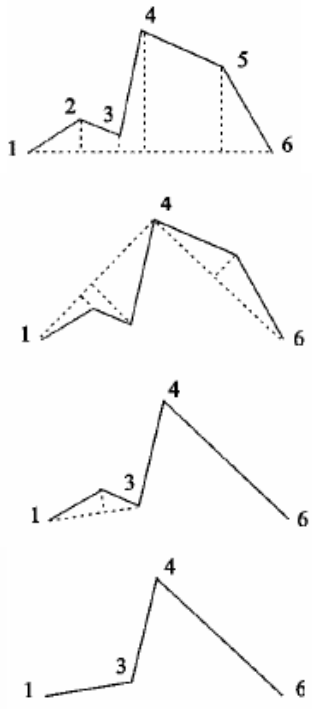


图 1　RPD 算法示意图

2　改进的RPD算法(I_RPD)

为了提高 RPD 算法的效率，只能在降低扫描次数上进行改进，尽量降低扫描的次数。通过分析 RPD 算法提出了一种可以从起始端点出发，根据设定的步长由左向右来查找特征点的算法，其具体步骤如下：

1) 对于数据点集合 $p_1, p_2 \cdots\cdots p_n$，设 $A = p_1$ 和 $B = p_1 + \Delta$ (Δ 表示步长)，用虚线段连接 AB，并将 A 放入特征点列的首位；

2) 在 AB 范围内的数据点集合中寻找与 AB 线段具有最大距离的点，记为 C；

3) 判断 C 点到 AB 的距离是否小于阈值 δ，若否，则设 $A = C$、$B = B + \Delta$ (向后移动 Δ 个位置，指向下一个数据点)，并将 C 放入特征点集合，用虚线段连接 AB，执行步骤 2)；

4) 判断 B 是否到达 p_n，若否，则设 $B = B + \Delta$，用虚线段连接 AB，执行步骤 2)；

5) 将 B 做为最后一个特征点放入特征点集合，算法结束。

图 2 是步长为 2 时利用 I_RPD 算法压缩数据的例子。该算法在绝大多数情况下只需从左向右扫描一次原始数据集即可实现特征点的提取，而且事先不需要知道全部数据点，对于需要实时处理的数据集压缩非常有效。在硬件系统实现数据压缩不方便的情况下，也可以采用该算法对读取的数据进行压缩。

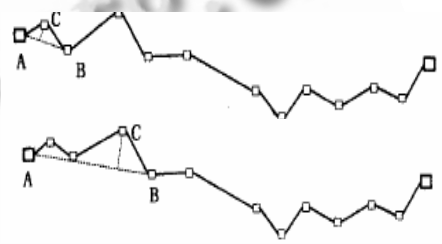


图 2　为 2 时 I_RPD 算法示意图

表 1 中对选择的三种曲线分别用两种算法进行了速度测试，每个曲线 5000 个点，时间单位为秒。从表中可以看出，除直线外，改进算法的效率远远大于未改进算法的效率。

表 1　算法效率对比情况

	直线	简单曲线	复杂曲线
RPD 算法	0.3	24.46	140
I_RPD 算法	21.75	0.4	0.52

3　I_RPD算法参数选择

3.1　Δ 参数选择

Δ 参数的选择与数据的起伏变化情况和每次取得的数据量的大小有关。由于该系统中数据的起伏变化通常集中在很短的时间内完成，所以 Δ 可以选择尽量大的一个数值，该数值由每次取得数据量大小决定，假设取得的数据中包含 n 个数据点，那么 Δ 的取值就可以定为 $n-1$。当然也可以对 n 个数据点的起伏变化情况进行事先估算，根据估算值的范围，适当减小 Δ 的取值。

3.2　δ 参数选择

δ 参数的确定与具体的硬件电路设计有关，假设

采用的是 16 位的 AD 进行采样，AD 输出电压范围是 0～5V，系统要求采样分辨率控制 0.005V 内，那么理论上的最高分辨率为 $5/2^{16}$ V。δ 参数的选择只要能够保证系统分辨率就可以了，表 2 为 δ 取不同值时的系统分辨率情况。

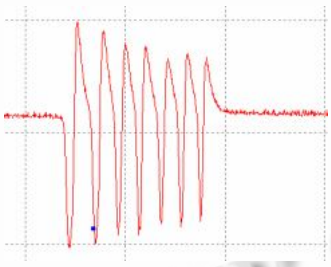
表 2　不同 δ 值时系统分辨率情况

δ	0	1	2	4	8
分辨率	$5/2^{16}$	$5/2^{15}$	$5/2^{14}$	$5/2^{13}$	$5/2^{12}$
δ	16	32	64	128	256
分辨率	$5/2^{11}$	$5/2^{10}$	$5/2^{9}$	$5/2^{8}$	$5/2^{7}$

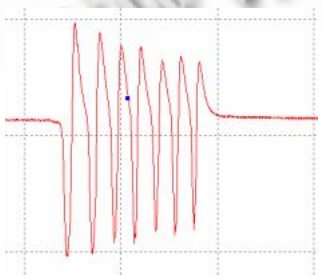
由表 2 可知，当 δ =32 时，系统分辨率为 0.0049V，值越小，压缩后失真越小；反之压缩后失真越大。因此 δ 取值范围为 0～32 即可。

4　试验验证

在某型号的弹载试验中，对弹载记录器采集的数据设计了基于 I_RPD 算法的数据分析软件，记录器数据容量最大为 10GB。实验对一个 1.8GB 的数据选取参数 $\Delta=30$，δ =32 进行压缩，压缩前后曲线分析对比效果如图 3 所示。步长 $\Delta=30$ 时，阈值 δ 不同时曲线显示时间情况如表 3 所示。



(a)　压缩前



(b)　压缩后

图 3　压缩前后效果图

表 3　步长时数据压缩前后显示时间情况

	δ =8	δ =16	δ =32	δ =64
压缩前显示时间	85s	85s	85s	85s
压缩后显示时间	4.3s	3.1s	1.7s	1.1s
压缩后数据容量	1.6GB	1.1GB	0.35GB	0.3GB

通过效果图和时间对比表可以看出，对 Δ，δ 选择合适的参数值，既可以保证数据的低失真率，还可以有效的压缩数据，为后续的数据分析节约了宝贵的时间，提高了软件的处理效率，为大数据的处理提供了技术支持。

5　结论

该算法易于实现，在实际的系统设计中既可以应用到硬件记录器系统端，也可以在软件系统端实现。在软件系统端实现时既可以在读取数据时进行压缩，也可以先不压缩，在数据分析时在对数据进行压缩分析。最后一种方法既可以保证原始数据的高保真度，又可以提高数据分析的效率，但是不能降低存储空间；前两种应用可以有效的降低存储空间，解决了长时间记录和有限存储空间的矛盾，但由于对数据进行了有损压缩[5]，导致数据失真，在实际应用中可以根据实际情况进行选择。

参考文献

1 Leming SK, Stalford HL. Bridge Weigh-in-motion System Development Using Superposition of Dynamic Truck/static Bridge Interaction. IEEE American Control Conference, 2003.

2 韩雪梅,彭虎,等. Huffman 编码用于 Sigma-delta ADC 波束形成器中 1bit 码流的压缩.生物医学工程研究, 2005,1:4－7.

3 吕京国,黄国满,杨明辉.用 Visual C++实现大数据量的快速存取.测绘学报, 2002,(3):29－32.

4 Visvalingham M, Whyatt J. Line Generalization by Repeated Elimination of Points. The Cartographic Journal, 1993,30(1):46－51.

5 华钢,闫军华,胡忠建.测控信源压缩方法研究.工矿自动化, 2004,(7):95－98.