

无损数据压缩技术研究

摘 要

本文对于无损数据压缩技术的相关问题进行研究。设计出两种无损数据压缩技术，选取了适合速变数据的分帧综合算法，并对提高实时性的手段进行了分析。

对于问题一，首先分析数据无损压缩经典算法的优缺点，进而融合经典算法设计了分帧综合算法，将数据进行分组，每一帧内采用 LZW 字典编码，每组内帧间的差值采用游程编码。经计算机仿真模拟，确定分组宽度为 25 帧/组时效果最佳。然后，以预测思想为引导设计了基于线性预测与 BP 神经网络的数据压缩技术。最后，利用压缩去除率、压缩解压时间等评价参数对两种方法进行综合分析、比较。

对于问题二，首先分析速变数据的特点，进而确定通过数据压缩效果、算法自身特性与应用潜能两个层次，选择适合速变数据的无损数据压缩方法，两个层次下分设 7 个二级指标。而后依据熵权法确定权值，并对权值进行二次修正，通过多指标综合评价模型对分帧综合算法、基于线性预测与 BP 神经网络的数据压缩技术、游程编码和字典编码四种方法进行评价，评价结果显示分帧综合算法最优。

对于问题三，首先，从提高分帧综合算法实时性的角度出发，综合分析出可通过分帧综合算法的硬件实现进一步提高算法实时性。其次，通过比较，选取 FPGA 硬件实现方案。最后，进行分帧综合算法的 FPGA 实现分析，过程包括结构划分、LZW 字典编码和游程编码的 FPGA 实现三个部分，最终达到提高算法实时性的目的。

本文在经典数据压缩方法的基础上进行创新，设计出两种无损数据压缩方法，结合 Python 进行计算机模拟仿真，对方法进行评价。在文章的最后，对模型进行推广。本文提出的方法不仅适用于航空遥测数据的处理，对大数据背景下其他领域数据的传输与处理同样具有参考、研究价值。

关键词：分帧综合算法；基于线性预测与 BP 神经网络的数据压缩技术；多指标综合评价模型；FPGA 硬件实现

一、问题重述

航天器与地面测控站之间距离遥远,采用无线电传输,地面测控站可以接收到的遥测信息其实是非常微弱的,信噪比小且信号失真大。因此需要采用数字纠错编码。航天遥测从信息传输的角度来说属于多路传输,遥测的数据量大且参数种类多,传输各类数据时都分别需要一条信道。航天遥测的传送内容除了参量信息还有图像信息(240000 比特/帧),包含的信息量大,因此可以考虑利用信息间的相关性进行压缩,航天工业发展的同时,对航天遥测信息的传输也提出了更高的要求。

近年来,航天科技发展迅猛,采集航天工程数据必不可少。而航天数据不断地呈指数型爆炸性增长使得对于信道容量的需求大幅度增长。为了在保证航天遥测信息不失真的同时,降低传输数据对于信道带宽以及传输功率的要求从而提高遥测信息的传输效率,就需要在航天遥测数据的压缩技术上进行进一步的提高。

对于近地轨道飞行任务,数据压缩可减小有效载荷重量,减少火箭上数据记录、地面采集、传输和数据处理成本费用,在深空探测任务中更是不可缺少,不仅可以提高带宽受限信道的通信能力,而且更能有效实时地传输遥测数据。对于航天遥测数据,任何信息都非常重要,所以需要对遥测数据进行无损压缩。下面请完成无损数据压缩技术的相关问题。

问题一:设计出几种无损数据压缩技术,并分析比较。

经典的无损压缩算法主要有:基于统计模式的算法,如 Shannon-Fano 编码、Huffman 编码、算术编码等;基于字典模式的算法,如 LZ 系列算法;还有一些其它的算法,如游程编码、JPEG-LS 算法、JPEG2000 的 DEM 算法、BWT 算法等。经典的算法各有优劣,而航天遥测技术的不断提高,对压缩技术提出了更高的要求,亟待设计新的压缩技术来满足处理大量数据的需求。而一般通过压缩去除率、压缩解压时间、算法复杂程度三个指标来评估比较算法的优劣性。

问题二:针对速变数据,选取合适的无损压缩技术,并说明理由。

对于遥测速变数据选取合适的无损压缩技术,需要考察数据压缩效果(压缩去除率和压缩解压时间)、算法本身特性与应用潜能。

问题三:对于选取的无损压缩技术,分析如何提高其实时性?

航天遥测数据的实时性对于数据的应用有极大的影响,所以此问需进一步探讨提高实时性的手段。

二、问题假设

1. 遥测数据在特殊波动时忽略不计
2. 压缩后的数据不受信道干扰的影响

三、符号说明

序号	符号	意义
1	CRR	压缩去除率
2	$e_{(x',y')}$	第一次预测误差
3	$e'_{(x',y')}$	第二次预测误差
4	R'	标准化后的决策矩阵
5	W'	二次修正后的权值矩阵

四、问题分析

4.1 问题一的分析

问题一要求我们设计出几种无损数据压缩技术，并分析比较。

显而易见，对于问题背景中提到的航天遥测数据，对于数据压缩的技术在压缩去除率和压缩速度上都提出了很高的要求。设计一种无损压缩技术可以从三个方面着手：

- (1) 选用新的压缩编码算法；
- (2) 在已有的经典算法基础上进行优化改进；
- (3) 结合多学科技术。

在分析比较不同压缩技术时，主要需要选取适当的评价指标，通过建模与计算机模拟等手段，依据结果比较不同压缩方法。

4.2 问题二的分析

问题二要求我们针对速变数据，选取合适的无损压缩技术，并说明理由。

为了针对航天遥测速变数据选取合适的无损压缩技术，我们需要对不同压缩方法进行评估，因此，应建立综合评价模型考察各压缩方法的技术指标。问题一中要求对提出的压缩技术进行初步分析比较，但为了使选择更具有说服力，需要将问题一中提出的压缩技术以及常用经典压缩算法代入数学模型中进行分析评估，以针对速变数据的压缩效果及算法本身特性与应用潜能等作为指标，分析结果，确定适合速变数据的无损压缩方法。

4.3 问题三的分析

问题三要求对于选取的无损压缩技术，分析如何提高其实时性。

一个压缩方法的实时性主要体现在压缩解压时间上。提高实时性除了继续优化算法，还可以结合相关学科进行优化，包括计算机软硬件技术、数字信号处理技术、数字通信技术、超大规模集成电路技术以及 DSP 通用芯片技术等。

五、问题一模型的建立与求解

5.1 经典无损压缩算法

数据一般包括四种冗余问题：①字符非均匀分布②字符重复③高使用率模式④位置冗余。为减少数据冗余，有许多经典的无损压缩算法被提出^[1]，其分类如图 1 所示。

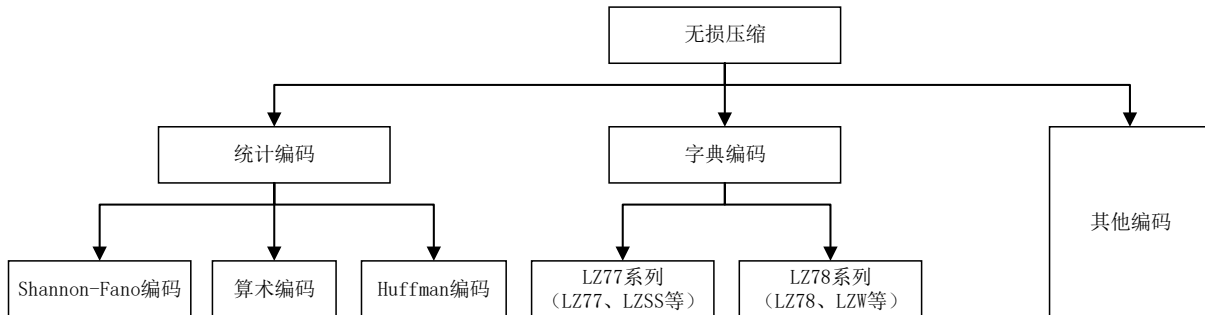


图 1 无损压缩算法分类图

通过了解不同算法的基本原理，归纳其特点如表 1。

表 1 经典无损压缩算法比较表

算法	压缩去除率	实时性	复杂度	最适用数据
游程编码	较好	较好	简单	连续重复数据
Huffman 编码	较好	需要预处理，实时性差	一般	任何数据均可
算术编码	很好	实时性差	复杂	任何数据均可
LZ77 编码	较好	较好	一般	局部相关性较好数据
LZ78 编码	较好	较好	一般	局部相关性较好数据
LZW 编码	较好	较好	简单	全局和局部相关性较好数据

从表中可以看出游程编码在压缩去除率、实时性、复杂度以及适应性上都有很好的特性，所以先从游程编码入手，考虑新算法的设计。

5.2 分帧综合算法

5.2.1 概述

航天遥测帧数据之间具有很高的相似性，目前的研究主要是对单帧数据进行无损压缩，并没有利用帧间的相似性对数据进行处理。因此，假如同时考虑单帧数据的局部相关性和帧间数据的全局相关性，在数据无损压缩过程中采取不同的方法对帧内冗余度和帧间冗余度分别进行处理，则能够达到更好的压缩效果。因此，本文提出第一种遥测数据无损压缩算法，即分帧综合算法，该算法首先把遥测帧数据按照一定长度进行划分，确定组内首帧为关键帧，对其进行 LZW 压缩，将其他帧与关键帧做差，差值进行游程编码。

5.2.2 算法原理

由于在航天遥测系统中单帧数据具有局部重复性，而帧间数据具有全局重复性的特

点，因此，可以采用无损压缩算法分别对帧内冗余度和帧间冗余度进行压缩处理。借鉴视频压缩领域的 MPEG 帧间压缩思想^{[2][3]}，得出可利用航天遥测数据的帧间相关性将航天遥测帧数据分组，对组内的单帧数据和帧间的差值分别采用不同方法进行压缩，即分帧综合算法。此外，由于航天遥测数据无损压缩的要求，对单帧数据和帧间数据的压缩也只能采用无损压缩算法。

将航天遥测数据按照固定间隔划分为若干组，将每组内第一帧数据定义为 I 帧，其他帧定义为 P 帧。分帧综合算法需要对单帧和帧间的差值分别进行压缩，但是考虑到航天遥测帧数据间具有全局相关性的特点，此算法中不必计算 P 帧与相邻帧的差值，只需对 I 帧数据进行存储，计算得到 I 帧与 P 帧的差值进行压缩。

由于航天遥测数据具有高速传输性，因此这里主要考虑采用游程编码和字典编码两种算法对遥测数据进行压缩。游程编码是指计算输入字符连续出现的次数，将连续出现的相同字符用 (X, S_c, RL) 表示，其中 X 表示数据流中的字符， S_c 起区分提示的作用，一般用输入序列中未使用的字符表示， RL 表示游程长度，即字符 X 重复出现的次数。游程编码在压缩处理过程中，只需要对所有的输入字符串序列扫描一次，即随着字符串序列的不断输入，压缩编码码字随即输出，算法实现简单，并且压缩处理的延时比较短。可以看出游程编码比较适用于对具有全局重复性的数据进行压缩。字典编码中的 LZW 算法在压缩和解压过程中动态建立字典，无需将字典又信源发送到信宿端，适用于对具有局部重复性的数据进行压缩。对于航天遥测数据具有单帧局部重复，帧间全局重复的特点，在帧内联合压缩算法中 I 帧数据采用 LZW 进行压缩，P 帧与 I 帧的差值采用游程编码进行压缩。

综上所述，本文提出的分帧综合算法的压缩过程如图 2 所示，具体实现步骤如下：

- (1) 对读入的遥测帧数据进行分组，取每组第一帧为 I 帧，其余帧为 P 帧；
- (2) 判断当前帧是否为 I 帧。如果为 I 帧，进行步骤 (3)，否则进行步骤 (4)；
- (3) 预先存储 I 帧，并且对 I 帧数据进行 LZW 压缩；
- (4) 计算 P 帧与 I 帧的差值，并且对差值进行游程编码；

每完成一次压缩，判断压缩是否完成。如果未完成，则进行步骤 (1)，否则结束整个压缩流程。

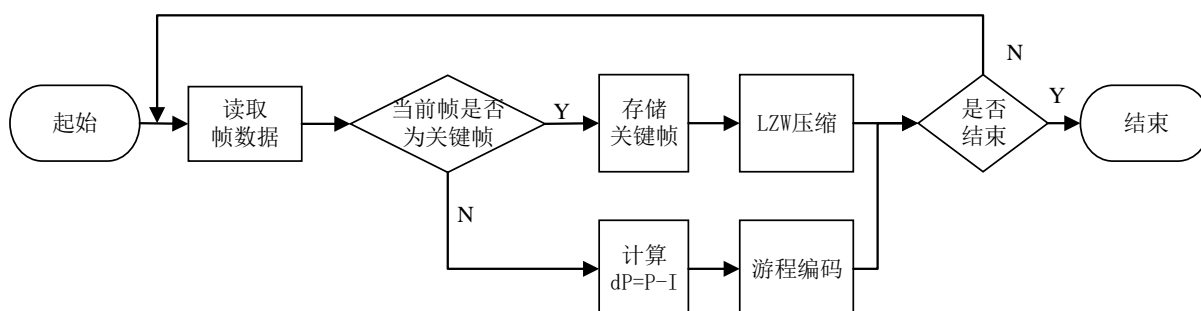


图 2 分帧综合算法的压缩流程图

解压过程则是读取压缩码字，判断是否为 I 帧，再根据结果分别采用 LZW 解压缩

和游程解压缩。解压过程如图 3 所示。

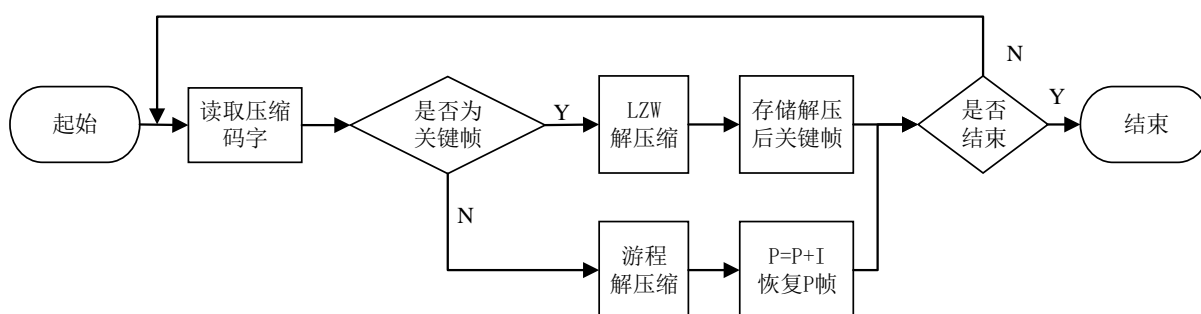


图 3 分帧综合算法的解压流程图

在采用分帧综合算法时需要将遥测帧数据进行分组，而分组数目的不同会影响 P 帧和 I 帧的全局相关性，进而影响数据压缩去除率。同时，不同的分组方式关系到 LZW 压缩的遥测帧数目，最终影响到整个压缩算法的处理延时。因此，需要对分帧综合算法中的分组方式进行讨论，确定最优压缩去除率和压缩延时下的分组宽度。

由于目前航天遥测系统采样速率一般为信号最高频率的 2.5-5 倍，因此本文仿真中分组宽度间隔为 5 分别为 5、10、15、20、25、30、35 等。通过仿真得到的数据每帧包括 512Byte，共包括 100 帧。不同分组宽度下的压缩去除率仿真结果如图 4。

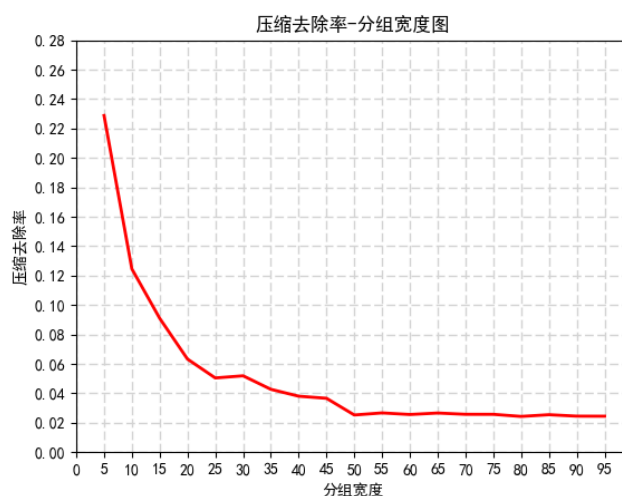


图 4 不同分组宽度下的压缩去除率仿真结果图

由图 4 可以看出不同分组宽度直接影响分帧综合算法的压缩去除率。在分组宽度从 5 逐渐增大到 95 的过程中，压缩去除率逐渐降低，当分组宽度达到 25 之后压缩去除率趋于稳定。再考虑解压时间会随着分组宽度的增加而逐渐缩短，之后趋于稳定，因此就此结果分析在分组宽度在 25 时压缩效果较好。由于航天数据相关性较高，因此运用分帧综合算法进行压缩时压缩去除率会提高。

分帧综合算法在计算 P 帧与 I 帧的差值时，差值可正可负，为了保证遥测帧数据可以正确的压缩和解压缩，需要对差值进行合理的设计。由于输入字符和计算出的差值都是 8bit 表示，差值为负时可以采用补码形式表示，并且在解压时可以通过求和恢复原始数据，所以本文提出对 P 帧与 I 帧的差值采用补码形式表示。压缩与解压结果如表 2 所

示。

表 2 帧压缩与解压结果

字符标号	0	1	2	3	4	5	6	7	8	9
I 帧	5	20	12	9	16	1	17	16	10	5
P 帧	5	20	17	5	16	1	17	16	12	10
$\partial_{=P-I}$	0	0	5	-4	0	0	0	0	2	5
∂	0	0	5	252	0	0	0	0	2	5
压缩结果	(0, 2)		(5,252)			(0,4)			(2, 5)	
解压结果	0	0	5	252	0	0	0	0	2	5
恢复 P 帧	5	20	17	5	16	1	17	16	12	10

5.3 基于线性预测与 BP 神经网络的数据压缩技术

5.3.1 数据特点及方法选取

航天遥测数据是一种时间-空间维度的数据。由于遥测对象通常为连续介质或连续对象，在空间维度内具有变化缓慢、相关性高的特点；由于遥测数据的获取频率较高，因此时间维度上数据同样具有相关性高的特点。

航天遥测数据在两个维度内均具有相关性，所以数据内含有大量冗余信息，因此无损数据压缩即可以此为思路，若能够最大限度在两个维度上减小各自的相关性，即去除两个维度内的冗余信息再进行编码，即可达到数据压缩的目的。

考虑到对于航天遥测数据压缩算法的简单性与实时性要求，在时间维度内进行数据压缩不利于实时性，因此本方法主要用于同一时刻空间维度内的数据压缩。同一时刻的数据即为同一帧数据。

针对空间维度内某一传感器采集的数据，表示方法应为 (x, y, z, p) ，其中 x 、 y 、 z 为空间位置， p 为传感器采集数据。由于此数据超过三维概念，不利于问题的简化与分析，因此，以气象格点数据^[4]为启发，将位置映射至二维空间，如经度-纬度的定位方式，将空间维度内某一传感器采集的数据转化为三维概念 (x', y', p) 。其中 (x', y') 构成一个定位点。

此三维概念可以通过图 5 表示：

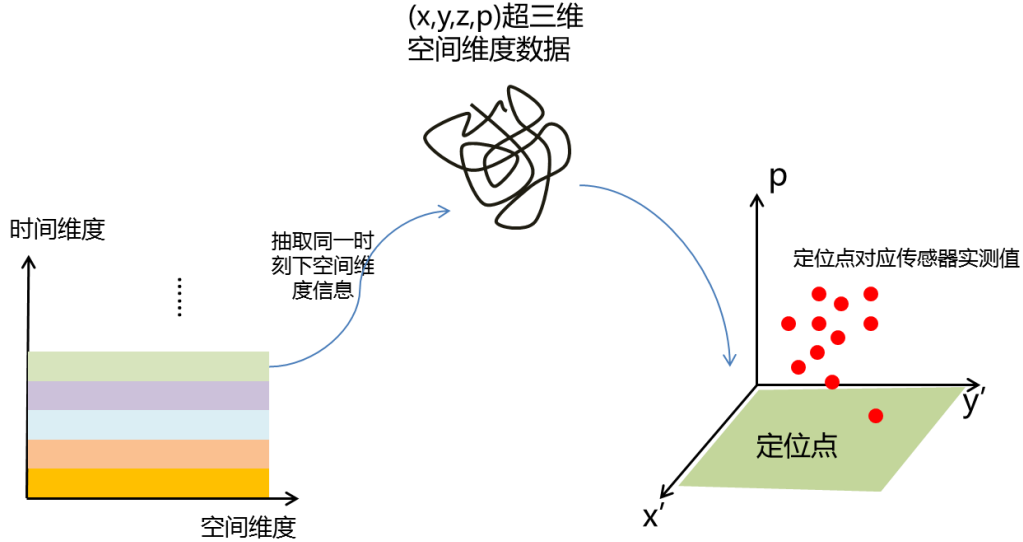


图 5 数据示意图

通过对航天遥测数据的相关性分析可知，空间维度数据间的相关性既存在线性相关性，又存在非线性相关性，因此，本模型利用线性预测方法最大可能地去除定位点间数据的线性相关性，同时利用 BP 神经网络的方法进行二次预测，进一步去除定位点间数据的非线性相关性。经过如上所述两次操作后再进行熵编码，Huffman 编码是较为通用的熵编码方法，具有算法简单、效率高的特点，因此，本模型仍采用 Huffman 编码进行编码。

5.3.2 空间维度线性一次预测

综合航天遥测数据压缩最大程度去除相关性与无损的两项要求，以此选择的预测方法具有可行性与实用性。

预测方法进行编码的基本原理为：按一序列方向前进，使用已有的数据预测序列下一数据，得到下一数据的预测值。预测值与真实值的差称为误差，预测准确度高时，误差的绝对值与方差较小，且误差值更接近 0，此时即去除了定位点之间的相关性，在预测后对误差进行编码可以最大限度的减少编码量。

而针对空间维度数据间的线性相关性去除，由数字图像处理中四连接、八连接定义启发，使用的线性方程预测方法，通过最小二乘法得到线性方程的预测参数，利用定位点周边其他定位点数据确定此定位点的预测值 $\hat{p}_{(x',y')}$ 。依据此定位点的预测值与真实值得到误差：

$$e_{(x',y')} = p_{(x',y')} - \hat{p}_{(x',y')} \quad (5-1)$$

根据定位点间的距离与相关性强弱，选取定位点周边合适数量、位置的定位点进行预测至关重要。本模型同时考虑相关性与计算复杂性，考虑选取定位点左侧、左上侧、上侧三点进行预测。关系如下式：

$$\hat{p}_{(x',y')} = a \times p_{(x'-[1],y')} + b \times p_{(x'-[1],y'-[1])} + c \times p_{(x',y'-[1])} + d$$

其中：a,b,c,d为预测系数； $\hat{p}_{(x',y')}$ 为定位点预测值； $p_{(x'-[1],y')}p_{(x'-[1],y'-[1])}p_{(x',y'-[1])}$ 为定位点左侧、左上侧、上侧的实测值

若预测准确，误差将在 0 及 0 附近变动，即将数据转换成概率分布相对集中的值，去除了其中部分线性相关性，达到了降低熵的目的。

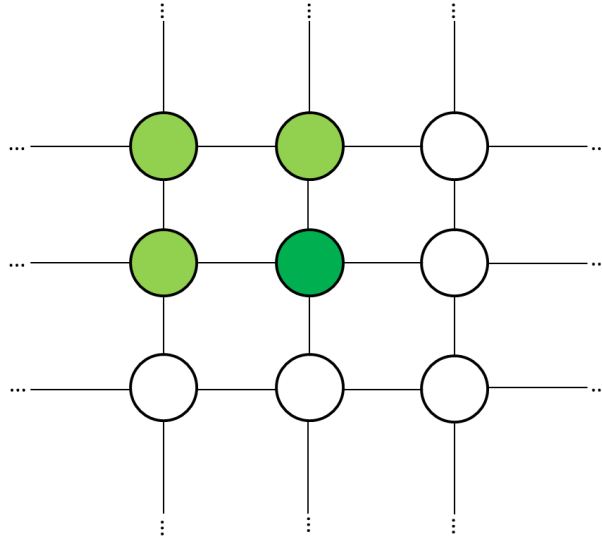


图 6 预测过程示意图

5.3.3 BP 神经网络非线性二次预测

在空间维度线性预测中，最大限度去除了数据间的线性相关性，但计算处理后不同部分的相关系数发现仍有相关性未被很好地去除，因此考虑进一步去除数据间的非线性相关性。

而神经网络是一种应用类似于大脑神经突触联接的结构进行信息处理的数学模型。神经网络层数的多少被称为神经网络的深度，而各层神经元个数的最大值为神经网络的宽度。矢量输入神经网络，经过权重 W 和偏差 b 的线性运算并嵌套入一个激活函数中进行一层的处理，其余神经层以同理进行排列。使用样本对神经网络进行训练的同时，通过损失函数对于调整结果进行量化，进而不断调整每一层权重 W 与偏差 b 的数值，使损失函数的数值尽可能小，从而使神经网络能够更好地通过输入值预测输出值。由于存在不同激活函数，神经网络可以很好地进行非线性预测。

而 BP 神经网络是一种按误差反向传播训练的多层前馈网络，BP 神经网络的基本思想是梯度下降法，利用梯度搜索技术，希望网络的实际输出值和期望输出值的误差均方差为最小。通过反向传播，也能够达到节省运算量的目的。

经以上分析可知，可以通过构建 BP 神经网络的方法，二次预测误差，从而最大限度减小数据间的非线性相关性。若本模型采用三层神经网络，分别为输入层、隐藏层、输出层，则各层间的关系为：

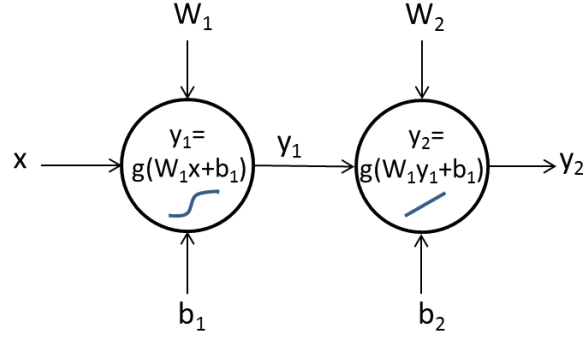


图 7 神经网络示意简图

$$\begin{aligned} y_1 &= g_1(W_1 \times x + b_1) \\ y_2 &= g_2(W_2 \times y_1 + b_2) \end{aligned} \quad (5-3)$$

其中 $g(x)$ 为激活函数， W_1 ， W_2 为权重， b_1 ， b_2 为偏差。在隐藏层可以选取 sigmoid 函数作为激活函数，而输出层可以采用线性函数：

$$\begin{aligned} g_1(x) &= \frac{e^x}{e^x + 1} \\ g_2(x) &= x \end{aligned} \quad (5-4)$$

本模型中的代价函数即为预测值与期望输出值的差的平方和：

$$\text{cost} = \sum (y_2 - y_{2\text{expect}})^2 \quad (5-5)$$

其中 y_2 为神经网络输出值， $y_{2\text{expect}}$ 为期望输出值。在本模型中通过构建此 BP 神经网络，输入第一次预测误差 $e_{(x',y')}$ ，得到第二次预测误差 $e'_{(x',y')}$ 。

5.3.4 二次误差的 Huffman 编码

Huffman 编码是一种自底至顶的变长编码方式，借助了数据结构中的树形结构，并依据 Huffman 算法构造出了最优二叉树。其主要原理是基于统计的观点，分配给出现次数多的符号以更短的编码。

Huffman 编码的主要步骤为：

- (1) 扫描需编码的数据,统计原数据中各字符出现的概率；
- (2) 利用得到的概率值创建 Huffman 树（按概率由小至大两两结合）；
- (3) 对 Huffman 树进行编码,并把编码后得到的码字存储起来(自根至叶标 0 或 1)。

针对线性预测与 BP 神经网络预测得到的二次预测误差进行 Huffman 编码，首先需将同一时刻（同一帧）内的边框值、线性预测系数、神经网络权重与偏差以及预测的误差按字节写成数据流，再通过 Huffman 编码规则进行编码。

考虑实时性要求，此过程也可以使用动态 Huffman 编码，只是需要对边框值、线性预测系数、神经网络权重与偏差、预测误差的编码顺序进行进一步确定。

5.3.5 模型的整体建立

模型的整体实现步骤为：

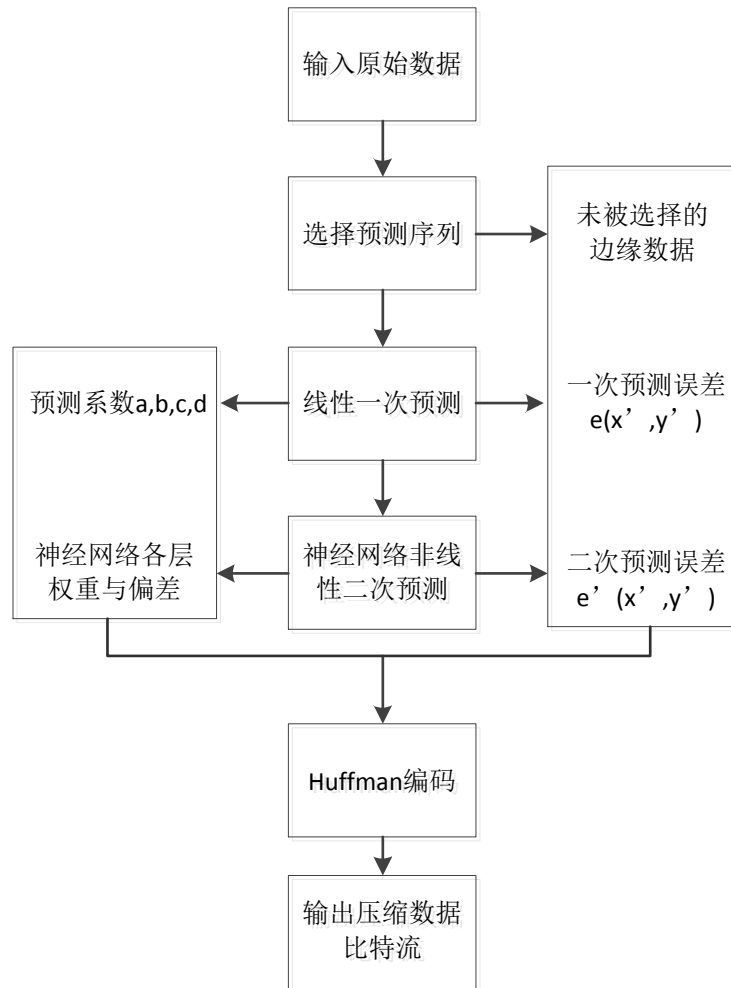


图 8 基于线性预测与 BP 神经网络的数据压缩技术流程图

5.4 评价比较

5.4.1 评价指标

1. 压缩去除率（Compression Removal Ratio）

压缩去除率，描述压缩文件的效果名，是指文件压缩掉内容的大小与压缩之前的大小之比，即原始数据压缩处理后被压缩程度的体现，计算公式如下：

$$CRR = \frac{L_{in} - L_{out}}{L_{in}} \quad (5-6)$$

式中：

CRR —— 压缩去除率

L_{in} —— 原始数据长度

L_{out} —— 压缩处理后的数据长度

通过公式我们可以看出，一般情况下，压缩去除率越低表示压缩效果越好，但同时也会导致解压时间延长。

2. 压缩解压时间

压缩解压时间是一项衡量压缩和解压的时间决定了信息传输过程中实时性的指标，

这一指标对硬件压缩的度量很重要。

3. 算法复杂度

压缩和解压的过程需要一定的硬件设施来实现，目前的各种数字处理器芯片都可以实现各种压缩解压算法，算法的实现是在同一 FPGA 上实现的，算法的复杂程度决定了占用资源的大小。因为除了压缩算法还包括捕获、跟踪以及信道编译算法，因此为了节约占用资源，对算法复杂度提出一定的限制和要求。

5.4.2 分帧综合算法的分析与评价

分帧综合算法利用单帧数据的局部相关性和帧间数据的全局相关性的特点，分别采取了 LZW 和游程编码两种算法进行压缩处理，相对来说比运用单一算法能提高遥测数据压缩的去除率，同时，游程编码在处理过程中，只需要对所有的输入字符串序列扫描一次，即随着字符串序列的不断输入，压缩编码码字随即输出，算法实现简单，并且压缩处理的延时比较短，能够使得算法总体有更高的实时性。

分帧综合算法主要利用遥测数据单帧局部重复性和帧间全局重复性，对单帧冗余度和帧间冗余度进行压缩处理。但是这个算法也有其局限性。如果在航天数据变化明显的一段时间内对遥测信号进行采样，则数据会在多帧后才具有全局重复性。在这种特殊情况下，采取组长为 25 的分帧综合算法可能会出现 P 帧和 I 帧无任何相关性，这样采用游程编码对帧间冗余度进行处理反而会增加冗余度。

5.4.3 基于线性预测与 BP 神经网络的数据压缩技术的分析与评价

针对模型的评价主要基于两个指标：压缩去除率与压缩解压时间。

一方面，基于线性预测与 BP 神经网络的数据压缩模型，结合两种方法，能够很好地去掉数据间的线性相关性与非线性相关性。在压缩去除率方面，可通过描绘原始数据、线性第一次预测误差和 BP 神经网络第二次预测误差的概率密度分布曲线可以看出，误差值的标准差值逐渐减小，且误差值集中在 0 附近，因此通过这种压缩手段可以大大提高压缩去除率。压缩去除率在实际运用中可以通过文件大小来进行比较。

另一方面，基于线性预测与 BP 神经网络的数据压缩模型，压缩方向在空间维度内，可以更好的适应航空遥感数据的实时性要求。但由于两次预测过程的存在，压缩解压时间较其余方法稍有增长。

但本方法的局限性在于，仅对连续性较好且相关性大的数据有较好的压缩效果，对于一些不连续的数据压缩效果差，甚至有可能增加编码量。

5.4.4 两种压缩算法的比较分析

表 3 两种压缩算法的比较分析

压缩算法	优点	局限性
分帧综合算法	①压缩去除率高于经典算法 ②算法简单, 运行时间短, 实时性高	对某种特殊情况处理结果差
基于线性预测与 BP 神经网络的 数据压缩技术	误差值的标准差小, 压缩去除率高	①两次预测, 压缩解压时间较长 ②对不连续数据处理结果差

六、问题二模型的建立与求解

6.1 航天遥测速变数据特性分析

遥测系统测量和传送航天器内部的工程参数和用敏感器测得的空间物理参数, 按照一定的时间间隔进行记录。对于遥测数据参数分类的方法有多种^[5]。按性质可分为模拟信号和数字信号。按频带范围可分为缓变参数(0Hz~10Hz)和速变参数(10Hz~8000Hz)^[6]。例如温度参数就属于缓变型数据, 在整个航天器飞行过程中数据变化很缓慢、波动很小。缓变信号数量多, 但只占数据的少部分, 速变参数是相对缓变参数而言的, 速变信号数量少, 它具有变化快, 信号频带宽的特点, 由于采样率高, 占了遥测数据的大部分。分析数据采样结果, 由于其采样频率远高于实际变化的频率, 从而使得数据的变化显得较为缓慢, 这为数据的压缩提供了可行性。

遥测系统将反映各分系统工作情况将其转化成在规定范围内变化的电信号, 然后由遥测系统采集、编码、调制后传输得到遥测数据。遥测系统的帧格式是典型的 PCM 帧格式。遥测系统以固定的采样率对每路模拟量进行采样, 并依次对每路信号进行编码。为了既不丢失信号的信息又能合理利用信道, 对于不同速率的参数, 应在主帧基础上用不同的副帧来实现^[7]。一个主帧由若干个副帧构成, 副帧分成多个波道, 存储不同信息。

具体数据结构, 其中包括 N 路速变信号和 N+M 路缓变信号, EB90 为帧标识。如图 9 所示

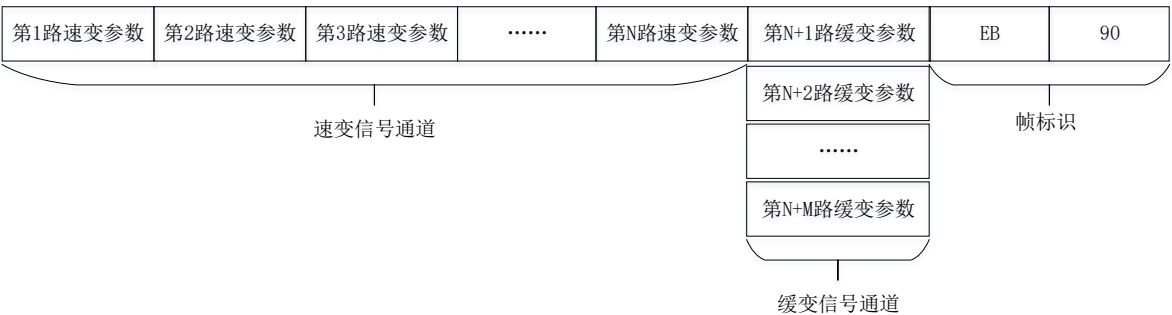


图 9 遥测数据的数据结构

6.2 多指标综合评价模型

6.2.1 适应速变数据的数据压缩技术评价指标确定

基于速变数据的数据压缩技术可以达到在一定容量的通道内传输更多珍贵、有效遥测数据信息的目的, 有助于降低带宽要求和传输功率, 提高通信效率。不同的数据压缩

技术在不同的指标衡量下具有不同的优势，因此建立适应速变数据的多指标综合评价模型至关重要。

首先，应当选择适应速变数据的评价指标。经过综合分析，计划从以下两个层次对于数据压缩技术进行评价：一方面为数据压缩效果，另一方面为算法本身特性与应用潜能。

1. 数据压缩效果

数据压缩效果是衡量数据压缩技术最为直观的体现，因此，我们选取压缩去除率、压缩解压时间两个指标，来评价数据压缩技术的压缩效果。两项指标前文已进行定义，此处不再赘述。

2. 算法本身特性与应用潜能

算法本身特性与应用潜能同样为数据压缩技术的评价层次，其对方方法的后期改进、维护、推广、应用有重要影响，因此，我们选取时间复杂度、空间复杂度、误码率、与硬件结合难度、优化空间 5 个指标来评价算法本身特性与应用潜能。

综上所述，适应速变数据的数据压缩技术评价指标体系如图 10 所示：

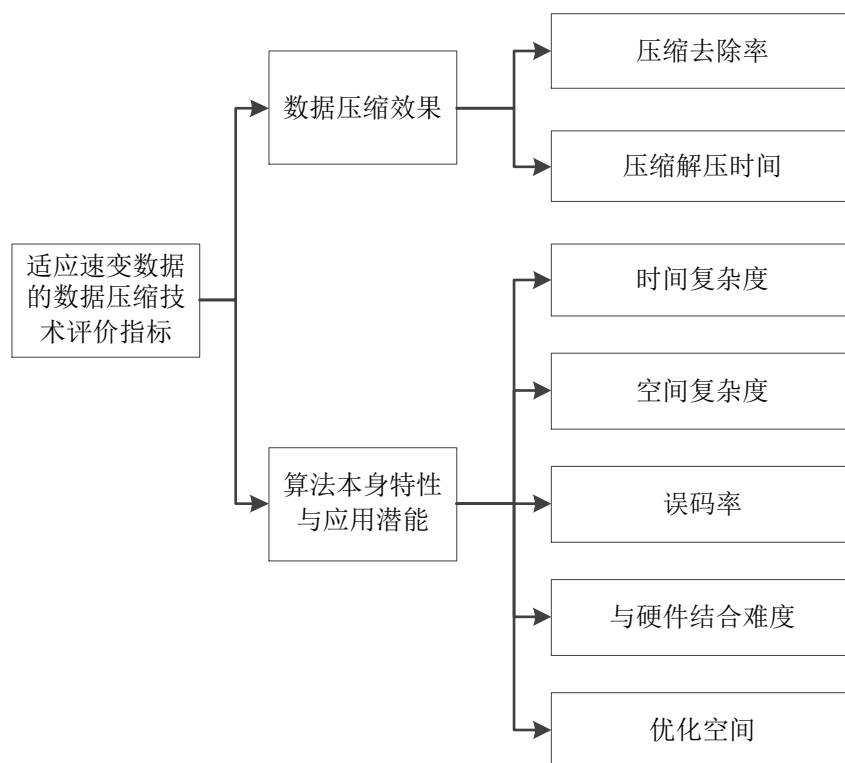


图 10 适应速变数据的数据压缩技术评价指标图

6.2.2 多指标综合评价模型的建立

本多指标综合评价模型，首先利用熵权法确定权重，由于客观赋权不利于体现评判者对不同属性指标的重视程度，因此针对熵权法确定的权值进行二次修正后，判定何种数据压缩方法更适合速变数据。

由材料力学中对于应变的定义可知：应变指在外力和非均匀温度场等因素作用下物

体局部的相对变形，即荷载对材料变形程度的影响。其中，线应变为某一物体长度变化量与原长度的比值，线应变的计算公式为：

$$\varepsilon = \frac{\Delta l}{l} \quad (6-1)$$

类比应变概念，在本模型中引入变形程度概念，描述应用某种数据压缩技术前后，评价指标的改变程度：

$$\mu = \frac{u - u'}{u'} \quad (6-2)$$

其中 μ 为变形程度， u 为应用某方法后该指标的观测/评价值， u' 为应用某方法前该指标的观测/评价值。以此为依据在后续评价中确定某种方法的某项指标得分状况。

1. 多指标的确定

由上式可知应用某种方法对于各指标的变形程度。由上文分析知，适应速变数据的数据压缩技术评价指标分为两层：数据压缩效果 U_1 和算法本身特性与应用潜能 U_2 。因此，一级指标集为 $U = \{U_1, U_2\}$ 。

其中数据压缩效果 U_1 包括压缩去除率 u_1 ，压缩解压时间 u_2 ；算法本身特性与应用潜能 U_2 包括算法的时间复杂度 u_3 ，算法的空间复杂度 u_4 ，算法使用过程中产生的误码率 u_5 ，算法与硬件结合进一步应用的难度 u_6 ，算法未来的优化空间 u_7 。因此二级指标集 $U_1 = \{u_1, u_2\}$ ， $U_2 = \{u_3, u_4, u_5, u_6, u_7\}$ 。

2. 权重的确定

在经济学评价中，通过计算“熵”确定权重，即根据各社会科学检测指标的差异程度，确定各指标权重，熵用于反映数据所包含的信息量。由于熵是系统无序程度的度量，当各评价对象的某项指标相差较大时，熵值较小，说明该指标提供的有效信息量大，相应其权重也较大；反之当各评价对象的某项指标相差较小时，熵值较大，说明该指标提供的有效信息量小，相应其权重也较小。当各被评价对象的某项指标完全相同时，熵值达到最大，此时该指标无有用信息，可以予以剔除。

运用本模型，对问题一中的分帧综合算法、基于线性预测与 BP 神经网络的数据压缩方法与经典压缩方法——游程编码、字典编码 4 种方法进行综合评价，每个方法有 7 项评价指标。因此现有 m 个待评价项目， n 个待评价指标，本问题中 $m=4, n=7$ 。

Step1: 形成原始数据矩阵 $R = (r_{ij})_{m \times n}$ 。各指标需进行标准化处理后得决策矩阵 $R' = (r'_{ij})_{m \times n}$ 。

$$R' = \begin{pmatrix} r'_{11} & r'_{12} & \cdots & r'_{1n} \\ r'_{21} & r'_{22} & \cdots & r'_{2n} \\ \cdots & \cdots & \cdots & \cdots \\ r'_{m1} & r'_{m2} & \cdots & r'_{mn} \end{pmatrix}_{m \times n} \quad (6-3)$$

Step2: 定义熵。定义第 j 个指标下第 i 个项目的指标值的比重 p_{ij} ：

$$p_{ij} = \frac{r_{ij}}{\sum_{i=1}^m r_{ij}} \quad (6-4)$$

定义第 j 个指标的熵 h_j :

$$h_j = -k \sum_{i=1}^m p_{ij} \cdot \ln p_{ij} \quad (k = \frac{1}{\ln m}) \quad (6-5)$$

Step3: 定义熵权。

第 j 个指标的熵权 w_j :

$$w_j = \frac{(1-h_j)}{\sum_{j=1}^n (1-h_j)} \quad (0 \leq w_j \leq 1, \sum_{j=1}^n w_j = 1) \quad (6-6)$$

Step4: 熵权的二次修正。

由于客观赋权不利于体现评判者对不同属性指标的重视程度, 因此针对熵权法确定的权重进行二次修正, 得到修正后的权: w_j' 。

6.2.3 模型求解结果

利用程序解得各项指标的熵权数并进行修正, 得到修正权重系数:

$$W' = \{0.183, 0.255, 0.148, 0.093, 0.144, 0.132, 0.045\}$$

对各方法的各指标值线性加权求和, 得各方法最终评分 Z_i , 分数越高说明该方法更适应速变数据进行数据压缩:

$$Z_i = \sum_{j=1}^n w_j' r_{ij} \quad (6-7)$$

经计算的综合评估分数向量为 $Z = \{15.378, 13.462, 8.349, 10.754\}$ 。方法一即分帧综合算法得分最高。

因此, 针对速变数据, 选择分帧综合算法较为合适。

七、问题三模型的建立与求解

7.1 分帧综合算法的 FPGA 实现分析

航天技术的发展对遥测系统提出了很高的要求, 遥测设备的增多和对传感器参数精度的提高使得数据量飞速增长, 这对数据的无线传输和储存带来了极大的挑战。解决这一问题的有效途径就是对遥测数据进行无损压缩, 因此, 本文通过设计分析基于线性预测与 BP 神经网络的数据压缩技术和分帧综合算法, 认为分帧综合算法在遥测数据压缩的实时性上较有优势。由上文对分帧综合算法进行仿真的结果可以看出, 压缩去除率与压缩延时成反比关系, 即如果要针对分帧综合算法进一步提高无损压缩的实时性, 通过硬件实现具有较好效果, 能够在提高实时性的同时保持较高的压缩去除率。

本文提出的分帧综合压缩算法结合了游程编码和 LZW 两种算法,算法基本原理为:在压缩的过程中,对输入的待压缩数据每隔 25 帧划分为一组,组内第一帧为 I 帧,组内其余帧为 P 帧。对 I 帧采用 LZW 压缩,其中, LZW 算法采用多字符并行搜索方式查询字典词条。对 P 帧与 I 帧的差值采用游程编码,编码输出采用游程标志与单字符混合输出的方式;解压时,对 I 帧进行 LZW 解压缩,对 P 帧差值进行游程解压缩,再将游程解压缩后的帧间差值与 I 帧相加,得到原始的 P 帧数据。算法通过本文的仿真验证了其正确性,为采用嵌入式硬件方案奠定了模型基础。

7.2 硬件方案选择

对于同一个算法,采用不同结构的数字逻辑电路来实现,在运算速度和性价比上会有所不同。目前,比较常见的能够实时完成算法运算的硬件方案有 DSP、FPGA 等,本文主要对这两种系统电路进行分析比较。

DSP (Digital Signal Processor) 是一种数据微处理芯片,芯片内包括运算单元、控制单元、各种寄存器以及内部储存单元等,其还可以在外围外接储存器,能与一定数量的外部设备相互通信。DSP 运行能力强,能实现复杂算法,而且速度快。

FPGA (Field Programmable Gate Array) 是一种半定制 ASIC。FPGA 由逻辑单元阵列构成,内部包括可配置逻辑模块 CLB、输出输入模块 IOB 和内部连线三个部分。用户可对 FPGA 内部的逻辑模块和 I/O 模块重新配置,以实现用户的逻辑。它还具有静态可重复编程和动态在系统重构的特性,使用灵活,通用性好。

DSP 代码的执行是顺序的,而 FPGA 是并行的。DSP 内部总线宽度不能随意改动,要实现分帧综合算法只能改变程序,而 FPGA 中数据宽度是可以随意改动的,具有更好的灵活性。因此对于航天遥测速变数据而言, FPGA 采用的并行结构能提高数据处理速度。

7.3 算法结构

如图 11,分帧综合算法结构主要包括五大部分:数据缓存单元、输入控制单元、输出控制单元、LZW 单元和 RLE 单元,对于 I 帧数据还需要 RAM 缓存。数据输入端 LZW 单元执行压缩算法, RLE 单元执行游程编码;数据输出端也与之对应, LZW 单元执行解压缩算法, RLE 单元执行游程译码。

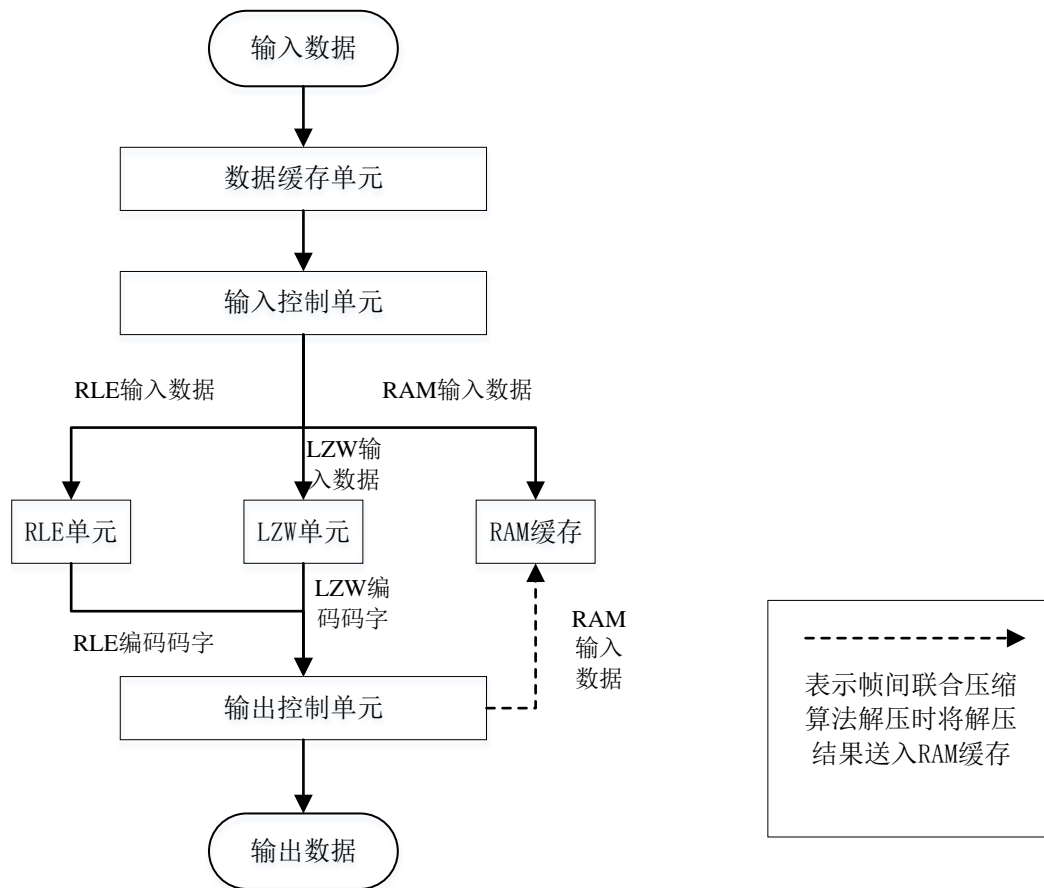


图 11 分帧综合算法的结构框图

7.4 游程编码 FPGA 实现

游程编码单元主要实现两个功能：在发送端实现游程编码，在接收端实现游程译码，即解压缩运算。该单元采用改进的游程编码算法，将游程标志和字符混合后输出。具体如下：

（1）压缩次数计数器、游程长度计数器、游程标志寄存器游程标志寄存器全部初始化为零；

（2）统计输入数据中连续字符出现的长度，此时，游程长度计数器不断累加。当游程长度计数器为 1 时，输出编码码字 $\{X_1, X_2\}$ ， X_1, X_2 分别为当前字符和下一个字符，同时置游程标志寄存器相应标志位为‘0’；当游程长度计数器在 2~255 范围内时，输出编码码字 $\{X, RL\}$ ， X 表示当前字符， RL 表示游程长度计数器的值同时置游程标志寄存器相应标志位为‘1’，令游程长度计数器重新初始化；

（3）每完成一次游程编码的统计和输出后，压缩次数计数器进行一次累加。当压缩次数计数器为 16 时，输出游程标志寄存器的值，令压缩次数计数器和游程标志寄存器重新初始化。

游程编码单元的解压缩过程 FPGA 实现具体如下：

（1）游程码字寄存器 array_RLE 初始化为零；

(2) 连续读入 17 个游程编码码字, 并且存储入游程码字寄存器 array_RLE 中, 根据 array_RLE(16) 的值分别处理 array_RLE(0)~array_RLE(15)。如果 array_RLE(16)(i) 为 '1', 则输出 array_RLE(i) 对应的游程串, 如果 array_RLE(16)(i) 为 '0', 则直接输出 array_RLE(i) 中的字符;

(3) 当前 17 个游程编码码字处理完成后, 重新读入 17 个编码码字, 完成上述的处理过程。

7.5 LZW 单元压缩过程的 FPGA 实现

LZW 单元压缩过程的 FPGA 实现流程具体如下:

- (1) 控制模块把三组待匹配的字符串输入字典存储模块;
- (2) 字典存储模块同时并行搜索三组待匹配字符串, 并且将匹配地址和匹配结果反馈到控制模块; 在并行搜索过程中, 若第一组字符串匹配成功, 则停止这次搜索, 如果失败, 则需要匹配查找字典中存储的所有字典词条;
- (3) 控制模块根据三组匹配搜索结果和匹配地址, 输出对应的编码码字, 同时更新下一次匹配搜索的起始字符位置, 将该次的匹配结果和更新内容送入字典更新模块;
- (4) 字典更新模块根据匹配结果、更新内容和当前字典更新地址指针, 将字典更新地址和字典更新内容送入字典存储模块, 由字典存储模块完成对新的字典词条的存储;
- (5) 在一次多字符并行搜索且输出编码码字后, 由下一次匹配搜索起始字符开始接着进行下一次三字符组并行搜索的过程。

LZW 单元解压缩过程的 FPGA 实现流程具体如下:

- (1) 控制模块将编码码字送入字典存储模块;
- (2) 字典存储模块根据编码码字直接由字典相应地址读取对应字典词条, 并且将字典词条反馈给控制模块;
- (3) 控制模块由反馈的字典词条输出解压缩后的字符串, 同时将更新内容送入字典更新模块;
- (4) 字典更新模块根据更新内容、当前字典更新地址指针, 将字典更新地址和字典更新内容送入字典存储模块, 完成对字典的更新操作;
- (5) 当一次编码码字解压缩完成后, 控制模块接着对下一个编码码字进行解压缩处理。

八、模型的评价与推广

8.1 模型的评价

本文建立的模型简单易懂, 建立过程自然、流畅。并随着问题的深入, 不断加以改进, 从经典方法入手, 并对经典方法进行融合、改进, 设计数据压缩技术, 达到了数据压缩的目的, 同时最大限度寻求压缩去除率与压缩解压时间之间的平衡。将如神经网络等新方法新技术应用于数据压缩的过程中, 为数据压缩技术发展提供新思路。

但在模型建立的过程中仍存在缺点：比如利用计算机模拟的手段进行实验，较真实航天遥测数据仍有很大差距，比如综合评价模型系数仍有较大主观成分且变化程度不易量化和进行标准化处理。这些缺点还有待进一步完善。

总体来说，设计达到了题目预设的目的，并有推广、应用的潜力。

8.2 模型的推广

本文设计数据无损压缩技术、选择适合速变数据的数据无损压缩技术，并通过方法改进以提高实时性。模型建立适应当前航天遥测数据的传输要求，随着大数据时代的到来，大量数据的存储问题将为人们所关注。因此本模型可以推广应用到与之类似的高频、高相关性数据的传输与存储，并以之为启发进行进一步深入研究。

九、参考文献

- [1] 吴乐南, 《数据压缩的原理与应用》, 电子工业出版社, 1995。
- [2] Afzal M K, Ki B S, Kim S W. Efficient and Reliable MPEG-4 Multicast MAC Protocol for Wireless Networks[J], IEEE Transactions on Vehicular Technology, 64(3):1026-1035 2015.
- [3] Sadek R, Youssif A, Elaraby A. MPEG-4 Video Transmission Over IEEE 802.11e Wireless Mesh Networks Using Dynamic-Cross-Layer Approach[J]. National Academy Science Letters, 38(2):113-119, 2015.
- [4] 赵苏璇,罗坚,杨成荫, 基于 BP 神经网络的气象格点数据无损压缩方法[J], 地球科学进展; 第 23 卷第 2 期 2008 年 2 月: 206-212。
- [5] Altera Corporation, Altera Digital Library[Z], Altera, 2002.
- [6] 李锦明,张文栋.实时无损数据压缩算法硬件实现的研究[J], 哈尔滨工业大学学报, 2006(2):165-167。
- [7] 马宁, 朱福荫, 尹志军,改进游程编码在天气雷达数据压缩中的应用[J], 解放军理工大学学报, 5(6):88-90, 2004。

附 录

Python 代码:

```
import numpy as np
import matplotlib as mpl
import matplotlib.pyplot as plt
from pylab import *
mpl.rcParams['font.sans-serif'] = ['SimHei']
column=512
row=100
data=np.zeros([row,column])
data[0,:]=np.random.randint(0,20,column)
ratio1=[]
width1=[]
for i in range(1,row):
    for j in range(column):
        a=np.random.randint(1,6)
        if a<=4:
            data[i,j]=data[i-1,j]
        else:
            data[i,j]=data[i-1,j]+np.random.randint(-5,6)
delta=np.zeros([row-1,column])
for i in range(1,row):
    for j in range(column):
        delta[i-1,j]=data[i,j]-data[i-1,j]

for width in range(5,row,5):
    propare=np.zeros_like(data)
    for i in range(row):
        if i%width==0:
            propare[i,:]=data[i,:]
            b=i
        else:
            for j in range(column):
                propare[i,j]=data[i,j]-data[b,j]
                if propare[i,j]<0:
                    propare[i,j]=256+propare[i,j]

    #print(propare)
    #####原始状态#####
```

```

#print('原始状态:',width=,width,'count=',row*column)
####整体代换状态####
count=0
for i in range(row):
    if i%width!=0:
        if prepare[i,0]!=prepare[i,1]:
            count+=2
        else:
            count+=1
    for j in range(2,column):
        if prepare[i,j]!=prepare[i,j-1] and prepare[i,j-1]!=prepare[i,j-2]:
            count=count+1
        elif prepare[i,j]!=prepare[i,j-1] and prepare[i,j-1]==prepare[i,j-2]:
            count=count+2
    else:
        count=count+column
ratio=(row*column-count)/(row*column)
ratio1.append(ratio)
width1.append(width)
plt.plot(width1,ratio1,color='r',lw=2)
plt.xticks(np.arange(0,100,5))
plt.yticks(np.arange(0,0.3,0.02))
plt.xlabel('分组宽度')
plt.ylabel('压缩去除率')
plt.title('压缩去除率-分组宽度图')
plt.grid(True, linestyle = "--", color = "lightgrey", lw=1)
plt.show()
#print(ratio1)
#print('重复编码=',width=,width,'ratio=',ratio)

```