

# 无线传感器网络基于时空差分阈值 Huffman 数据压缩

吴佳杰, 朱荣钊, 钟志峰

(湖北大学 计算机与信息工程学院, 武汉 430062)

**摘要:** 针对无线传感网络 (WSN) 中节点采集数据后传输消耗能量的问题进行研究, 该文提出了一种基于时空差分阈值 Huffman 模型对时空的数据进行压缩的算法; 该算法通过节点在当前时刻采集的数据与节点的上一个时刻的数据进行差值, 同时对同一簇内空间的节点差分后数据进行对比分析和减少空间的数据量, 再对差值进行动态的 Huffman 编码; 与传统 WSNs 的数据压缩算法相比, 数据的压缩比提高到了 30%, 网络的生命周期提升了 80% 和降低了节点的死亡率, 同时也保证了节点数据收集的准确性。

**关键词:** 无线传感器网络; 差分阈值; 动态 Huffman 编码; 数据压缩; 网络生命周期

## Wireless Sensor Network Based on a Spatio-temporal Differential Threshold Huffman Data Compression

Wu Jiajie, Zhu Rongzhao, Zhong Zhifeng

(School of Computer Science and Information Engineering, Hubei University, Wuhan 430062, China)

**Abstract:** In view of the energy consumption of data transmission in wireless sensor network (WSN), a temporal and spatial differential way threshold Huffman model is proposed to compress spatio-temporal data in the article. The algorithm makes a difference between the data collected at this time and the data stored in the node and has relative differences in the data of the nodes in a certain space, then encoding dynamically Huffman of the difference. Compared with the traditional WSNs data compression algorithm, the way make the compression ratio of the data mentioned 30%, while the life cycle of the network is increased by 80% and the death rate of the nodes is also reduced, and the accuracy of the data acquisition is ensured.

**Keywords:** wireless sensor network; a spatio-temporal differential threshold; dynamic Huffman coding; data compression; network life cycle

### 0 引言

传感器网络<sup>[1]</sup>由大量的传感器节点随机部署在空间中组成的, 单个节点主要组成模块包括数据采集模块、通信模块、电源管理模块。由于节点的能量、存储、通信距离有限, 为了减少节点在数据传输的过程中消耗的能量<sup>[2]</sup>和节点无休眠的工作消耗的能量, 采用了对节点有限时间内自适应采集的数据与数据压缩的方式。该方式不仅减少了相对应数据的传输量<sup>[3]</sup>, 而且减少了节点能量的消耗; 此时通过传感器网络中节点不同时刻和不同位置采集的数据具有一定的时空相关性<sup>[4]</sup>, 以及传感器节点对自身采集的数据可以通过硬件功能进行处。然而无线传感器网络中的节点在同一片区域采集的数据具有相对变化较小的特性, 对该处理的数据将进行无损压缩<sup>[5]</sup>, 并要求传送的数据(如: 温度, 湿度)精确到一定程度。本文提出了一种基于

时空差分后 Huffman 的数据无损压缩算法。该方案采用单个节点自身对数据采集和处理, 处理后进行编码传输, 节点的传输数据大量减少<sup>[6]</sup>, 同时也能将压缩的数据进行还原。因此减少了节点发送的数据量和网络中的通信流量<sup>[7]</sup>, 并延长网络生命。

Steffen<sup>[8]</sup>提出了一种自适应压缩大型矢量的方案, 该方案对节点自身的处理数据要求非常高, 然而节点采用的处理方式速度较慢, 加大了处理数据的时间, 数据不能及时发送, 也未达到节约能耗。Ikjune Yoon<sup>[9]</sup>提出了无线传感器网络能量采集的自适应传感和压缩率选择方案, 通过电池的能量情况选择不同的压缩模式。该方案在开始就需要比较对压缩方案进行选择判断, 在节点能量较低的情况下采用低压缩率方式, 对节点的能量并没有太多的节省。Sunyong Kim<sup>[10]</sup>提出使用能量回收的无线传感器网络中的数据压缩延长网络寿命, 该方案通过簇头对数据进行多级压缩, 然而多级压缩过程容易导致树列后节点的能量消耗过快。Francesco Marcelloni<sup>[11]</sup>提出了一种无线传感器网络监测小节点的高效无损压缩算法, 该方案采用的 LZW 编码这种扩展会产生更高的内存占用, 但不会明显影响压缩比, 而且对应于较长代码的差异具有非常接近 0 的概率。

收稿日期: 2018-11-21; 修回日期: 2018-12-17。

**作者简介:** 吴佳杰(1992-), 男, 湖北荆州人, 硕士研究生, 主要从事无线传感器网络方面的研究。

**通讯作者:** 朱荣钊(1981-), 男, 理学硕士, 主要从事系统集成方向的研究。

上述文献都考虑到了压缩节点采集的数据，然而并没有真正在将数据量减少的情况下进行无损压缩数据<sup>[12]</sup>，同时保证数据采集的时效性和数据的无损性。本文通过时空性使用差分器将数据差值，再将差值后的数据与阈值比较后进行 Huffman 编码，最后节点将无损压缩的数据传递给簇头，再发送给服务器，服务器接收到数据后通过叠加的方式将数据还原。由于服务器是一直供电，当对收到的数据进行还原时并不用考虑服务器的能耗问题，通过实验数据分析，采用差分 Huffman 编码减小了节点的功耗<sup>[13]</sup>，同时也提高了数据压缩的比率和网络生命周期，并保证了节点在能量消耗保持相对稳定。

## 1 数据压缩理论

### 1.1 数据的时空性

无线传感器网络节点通过传感器采集数据，并通过 A/D 模型将采集的模拟信号转换成数字信号，节点发送的数据包含节点 ID 和节点所采集数据（图 1）。通过节点采集的数据具有一定的时间相关性，同一节点相邻的两个时刻采集的数据具有一定的关联，当相邻时刻采集的数据不变时，可以减少大量的数据传输；同时，不同的节点间的数据具有一定的空间相关性，当在一定的区域中，不同节点采集的数据具有一定的差异时，根据区域内多个传感器采集的数据进行比较分析，可以将大量一致的数据在矩阵中转换成 0 后，再进行传输到网关，并可以减少簇头在传输时的数据。此时可以将数据放置在三维空间表示（图 2），节点通过当前时刻的数据与前一时刻的数据进行差值，并将同一簇类节点的数据进行比较，之后采用无损压缩将数据传输给服务器。由于无线传感器网络的数据传输消耗的能量与距离和放大系数有关，传输数据的能量远大于处理数据所用的能量，此时传输数据消耗的能量<sup>[14]</sup>公式为：

$$E_{TX} = mE_{dec} + E_{amp}(m, d) = mE_{dec} + m\epsilon d^\chi$$

$$\begin{cases} \epsilon = \epsilon_{fs}; \chi = 2 & d < d_0 \\ \epsilon = \epsilon_{mp}; \chi = 4 & d \geq d_0 \end{cases} \quad (1)$$

$E_{dec}$  为采集 1bit 数据消耗的能量； $E_{amp}$  为传输时消耗的能量， $d$  表示节点与簇头的距离， $\epsilon$  表示传输时与距离相关的放大倍数；

节点ID	箭头ID	能量	温度	...	湿度
------	------	----	----	-----	----

图 1 节点数据结构

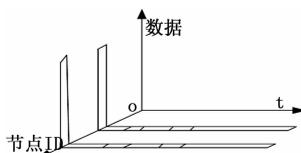


图 2 节点的时空性

### 1.2 一阶差分方程

节点中的每一个传感器采集到的数据不相互影响，通过差分器将现在时刻采集的数据与前一个时刻的数据进行差分（图 3）。多个数据同时进行并行处理并获得一个差分值。该差分器的公式为：

$$y_i(n) = x_i(n) - x_i(n-1) \quad (2)$$

而此时  $x_i(n)$  为单个节点传感器某时刻所采集的数据  $x_i = \{x_{i1}, x_{i2}, x_{i3}, \dots, x_{im}\}$ ，并通过单个节点现在采集的数据与前一时刻  $x_i(n-1)$  的数据进行差分，并获得  $y_i(n) = \{y_{i1}, y_{i2}, y_{i3}, \dots, y_{im}\}$ ， $y_i(n)$  为单个节点所有传感器采集的数据差分后保存的数据；为了保证数据的有效性，同一簇内将对不同的节点进行比较分析，同时消除数据的冗余；这时再将所获得的数据  $y_n$  进行编码，同时根据  $y_n$  所产生的数据进行 Huffman 编码，并确定该值以确定编码长度。

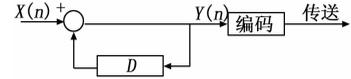


图 3 差分器数据模型

此时为单个节点的数据差分，由于多个节点具有多个传感器可以采集多个数据，此文将根据 LEACH 分层协议<sup>[15]</sup>，普通的节点将数据传输给簇头，簇头将数据收集后传送给网关，最后在传给服务器。节点将采集的数据转换在矩阵中进行存储，初始数据都为  $H_0$ 。（为了方便处理数据已经在服务器接收到数据应为同等大小矩阵，簇内低于规定的矩阵的列数采用零补充，大于列数的簇将舍弃簇内数据差异性很大的节点）。当所有节点部署完毕后并正常工作，第  $i$  个节点初次开始采集的数据  $x_i = \{x_{i1}, x_{i2}, x_{i3}, \dots, x_{im}\}$  并存储在内部的存储器中并传递给簇头，同时将一个簇内所有节点采集的数据直接传递给簇头，并在簇头中以矩阵  $H_1$  方式存储，并进行编码传递给网关；此时，每个节点都有了初始的采集数据，而后每一次采集的数据通过与前一时刻采集的数据进行差分（式 2），单个节点差分后的数据为  $y_i(n)$ ，同时簇内有多个节点，并将保存数据存储为  $H_n$ 。为了保证数据有准确性和数据压缩效率高，并在此设定一个标准的高斯分布（ $\alpha=0.05$ ）作为数据置信区间的阈值矩阵为  $H_T$ （图 4），并以此判断差分后的数据是否在阈值范围内；当  $y_i(n)$  的差值大于阈值，则此时刻节点判断差值大于阈值，节点将重新装载数据  $x_i$ ，同时节点将数据传递给簇头，并重新载入新的节点数据给  $H_1$ 。

$$H_0 = \begin{pmatrix} 0 & \dots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \dots & 0 \end{pmatrix} \quad H_1 = \begin{pmatrix} x_{11} & \dots & x_{1n} \\ \vdots & \ddots & \vdots \\ x_{m1} & \dots & x_{mn} \end{pmatrix}$$

$$H_n = \begin{pmatrix} y_{11} & \dots & y_{1n} \\ \vdots & \ddots & \vdots \\ y_{m1} & \dots & y_{mn} \end{pmatrix} \quad H_T = \begin{pmatrix} \alpha_{11} & \dots & \alpha_{1n} \\ \vdots & \ddots & \vdots \\ \alpha_{m1} & \dots & \alpha_{mn} \end{pmatrix}$$

图 4 数据的存储

而此时的  $H_n$  是在某个时刻在一个簇内不同的节点产生的数据并具有一定的空间性，通过于上一时刻的数据差分后，簇头存储的数据可以根据空间性，将矩阵中的数据大部分转换成 0，但并不是所有的差分后数据都是 0，再次之后对簇头的数据进行数据压缩，这样将大量减少了簇头的

数据量, 所以将采用时间和空间相结合对数据进行处理。

## 2 差分后的 Huffman 数据压缩

### 2.1 动态 Huffman 编码

如果基于静态 Huffman 编码则需要对字符需要两次扫描, 而动态 Huffman 编码是不需要事先构造哈夫曼树和输入扫描符号流; 而是一边进行编码, 同时开始构造哈夫曼树; 这样可以减少处理器消耗的能量和时间。

当初始化编码数时, 由于只能对数据流进行一次扫描, 所以不能预先知道每一个符号的出现次数和频率。为了对所有符号一致对待, 编码树的初始状态只包含一个叶节点, 包含符号 NYT (Not Yet Transmitted, 尚未传送), 权重值为 0。NYT 是一个逸出码 (escape code), 不同于任何一个将要传送的符号。当有一个尚未包含在编码树中的符号需要被编码时, 系统就输出 NYT 编码, 然后跟着符号的原始表达并重新给予新的编码树。

当输入一个字符时, 将会构造一个空的新子树, 子树包含 NYT 符号和新符号的两个节点, 并用新的节点取代旧的 NYT 节点在子树的位置 (最初的 NYT 权重为 0), 将旧的 NYT 取代后, 新符号节点的叶节点的权重为 1, 同时将试图对该父结点执行权重加一的操作。同时, 每读取一个字符, 首先检查字符是否已经在编码树中; 如果不在, 则先对空叶结点进行编码, 再生成一颗子树, 其右分支结点为刚的读入的字符, 其左分支结点为新的空叶结点, 并用这个子树代替原来的空叶结点; 如果在已经编码的树中, 则以静态哈夫曼编码的方式进行编码, 如图 5 所示。

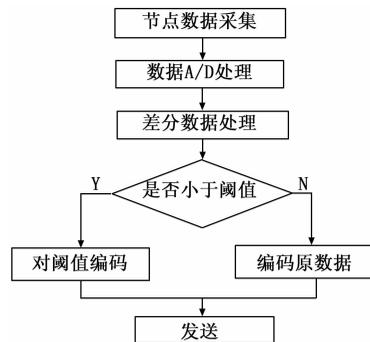


图 6 数据处理过程

为了保证数据传输的没有太多的冗余的数据, 每一次数据都要进行差分阈值的比较, 而此时采用的阈值使用高斯分布置信区间验证是否能正常通过  $H_0$  进行动态 Huffman 编码。对该差分后的数据进行高斯分布  $X \sim N(0, 1)$  的检验思想方式, 假设差分后的  $H_0$  能通过置信区间, 则将数据传输给编码器进行动态 Huffman 编码; 反之, 差分后  $H_0$  通过不了置信区间, 此时, 节点将采集到的原始数据进行动态 Huffman 编码, 并将此刻采集到的源数据压缩后发送给簇头, 同时簇头收到数据后, 也将采用动态 Huffman 编码进行传输。然而置信区间往往可以由某参数的显著性水平为  $\alpha$  ( $\alpha=0.05$ ) 的检验, 得到该参数的置信度为  $1-\alpha$  的置信区间<sup>[2,22]</sup>, 当在置信区间内是, 反之。显著性水平  $\alpha$  的均值  $\mu$  的双侧检验出现问题, 则高斯分布标准化公式:

$$Z = \frac{\bar{X} - \mu}{\sigma / \sqrt{n}} \quad (3)$$

对数据进行差分后减少了大量的数据, 在同一区域内分析数据的可靠性之后, 并通过动态的 Huffman 编码对矩阵  $H_n$  进行数据压缩, 由于每一次的数据都是未知的, 通过 Huffman 编码的动态原则, 先编码在调整编码数进行压缩。动态的 Huffman 编码以默认的概率进行编码, 并不断更新元素的概率, 从而可以达到高概率出现的分配给小规模的数据流位。

动态 Huffman 编码是基于动态变化的哈夫曼树, 也就是说, 对第  $t+1$  个字符编码是根据原始数据中前  $t$  个字符得到的哈夫曼树来进行的。压缩和解压子程序具有相同的初始化树, 每处理完一个字符, 压缩和解压方使用相同的算法修改哈夫曼树, 因而该方法不需要为解压而保存树的有关信息。压缩和解压一个字符所需的时间与该字符的编码长度成正比, 因而该过程可以实时进行。

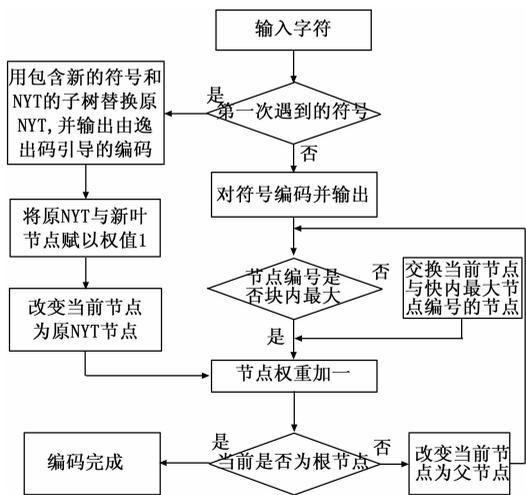


图 5 动态 Huffman 编码过程

### 2.2 时空差分阈值 Huffman 编码

当节点采集好的数据在自身的内存中具有一定的时效性, 通过该内存存储初始的数据作为一个基准, 初始时刻的数据直接基于动态 Huffman 编码进行传输, 此后的数据基于前一时刻的数据先进行差分, 并存储在矩阵中  $H_n$ 。对差分后的数据设定一个阈值  $H_T$ , 当进行差分后的数据小于阈值, 则节点直接将差分器差分的差值进行编码, 如果节点的某一个传感器差分的值大于阈值, 则节点将此时的数据直接编码进行传输, 并以此数据为基准进行差分, 如图 6 所示。

## 3 实验仿真

通过 Matlab 对数据压缩进行处理, 为了让实验效果更明显, 将采用 LEACH 协议对节点采集的数据进行压缩, 同时引入 SINK 中继站尽量节省节点的能量。与原有不采用压缩对比, 能量消耗相对减少、数据传输效率和网络寿命都相对提高, 同时也保证了数据的准确性; 并主要分析了 WSNs 网络的数据传输量, 对比网络生命周期等数据; 将 300 个节点随机部署在  $400\text{ m} \times 400\text{ m}$  的正方形内, BS 的坐标为  $(0, 0)$ 。在仿真中, 本节通过大量的实验比较同一领

域的压缩算法；主要分析 WSNs 网络寿命和数据传输。WSNs 主要参数如表 1 所示。

表 1 网络仿真参数

参数	取值
区域大小	400m×400m
节点数量	300
基站位置	(0,0)
节点初始能量	0.3J
数据包长度	4000bit
控制包长度	100bit
多路衰减模型功率放大系数	$10\text{pJ} \cdot (\text{b} \cdot \text{m}^2)^{-1}$
自由空间模型功率放大系数	$0.0013\text{pJ} \cdot (\text{b} \cdot \text{m}^2)^{-1}$
数据融合消耗能量	$5\text{nJ} \cdot \text{b}^{-1}$

通过实验对比分析，当节点采集数据后进行 Huffman、动态 Huffman、差分动态 Huffman 三种压缩算法，三种无损压缩算法都对节点的数据进行了压缩，然而前两种算法的压缩效果并未达到理想效果，通过图 7 可以看出，差分动态 Huffman 算法在压缩前对数据进行了相对应的融合，在减少了数据的传输同时反而增大了数据量传输，明显可以看出 Huffman 编码的压缩效果低于差分动态 Huffman 无损压缩。

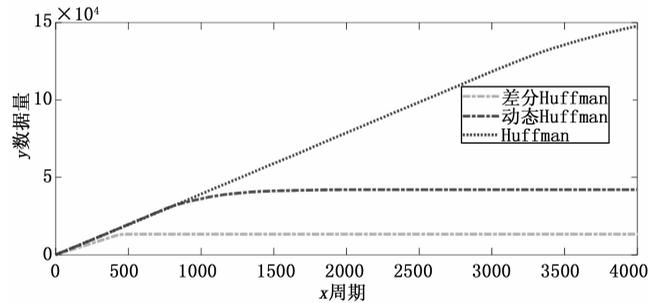


图 7 三种算法的数据传输

图 8 呈现了三种算法的能量的消耗，当节点与簇头之间的距离较远时，节点在数据采集后进行数据压缩消耗的能量低于数据传输的能量，当节点采集数据开始时，节点第一次的数据将直接进行动态 Huffman 编码传输，此后节点通过前一次时刻的数据与现采集的数据进行差分，并将大量的冗余数据抵消，只用将差值后的数据进行动态 Huffman 编码，再将编码后的数据进行传输。数据传输的数据量越少，则网络消耗的能量就越少。此时整个系统中的能量消耗减少，则总体的剩余能量将相应的减少，图 9 表示为三种压缩算法的剩余能量对比。图 10 显示，当采用差分 Huffman 算法时可以大大减少节点的死亡率并保证了节点的相对稳定性，同时也提高了数据的压缩率，这样将会让整个网络更好地监控该区域的环境。

在仿真中，为了保证该方案可行性，采用了节点数为 100/200/300/400/500 个节点部署进行了数据压缩比的分析。通过压缩算法的训练，当节点数越多时，三种算法的压缩率都会提高，而差分 Huffman 算法的压缩也随之提高。

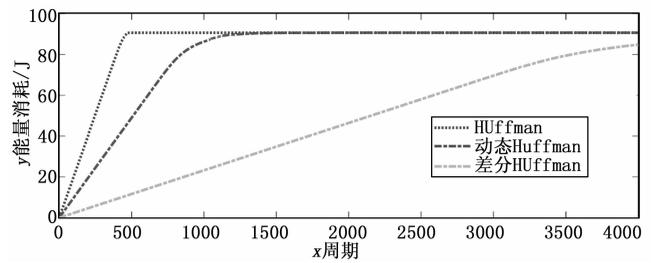


图 8 三种算法的网络能量消耗对比

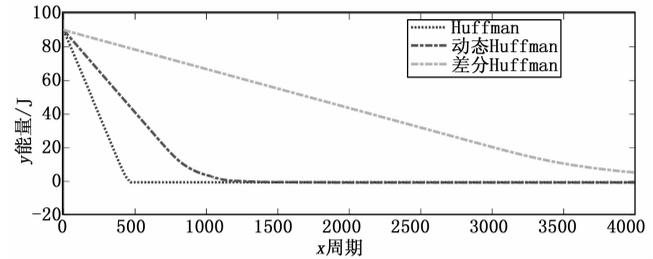


图 9 三种算法的网络的剩余能量

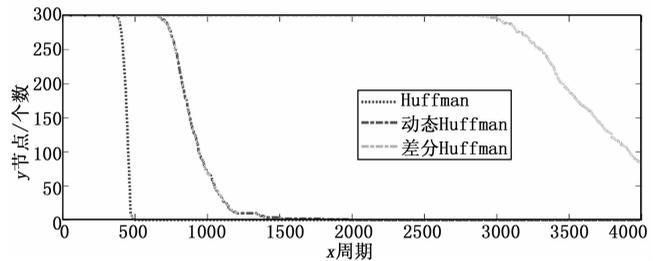


图 10 三种算法节点的存活数

压缩率公式为：

$$CR = \frac{\text{压缩后的数据大小}}{\text{压缩前的数据大小}} \times 100\% \quad (4)$$

表 2 三种算法的压缩率对比

压缩率	100	200	300	400	500
Huffman	87	83	78	75	74
动态 Huffman	81	76	70	66	65
差分动态 Huffman	80	73	64	54	48

通过表 2 的数据压缩比率统计当节点数增多时，差分动态 Huffman 算法的压缩比率是不断增大，传输的数据量相对是不断减小。当直接采用 Huffman 编码时当节点数增加到某一个值时节点的压缩率将会保持并不在减小，而差分动态 Huffman 算法确保了节点的数据压缩率、数据的无损性、数据的准确性、节点的节能性，同时保证了网络系统的稳定性。

#### 4 总结

本文主要针对无线传感器网络数据压缩算法进行研究，采用差分器再对数据进行动态 Huffman 编码。当数据进行减法时将大量的相同数据抵消，只用将差值编码进行传输，极大的节省了数据传输的能量，并将数据传输的准确性也得到了提高，也减少了簇内成员的通信开销，提升了网络的生命周期。然而本文的不足在于没考虑到 SINK 的运行速

度和路径规划。后期将会结合 SINK 与数据压缩, 并提高网络生命周期。

参考文献:

[1] Dener M. WiSeN. A new sensor node for smart applications with wireless sensor networks [J]. Computers & Electrical Engineering, 2017, 64.

[2] Srisooksai T, Keamarungsi K, Lamsrichan P, et al. Practical data compression in wireless sensor networks: A survey [J]. Journal of Network & Computer Applications, 2012, 35 (1): 37 - 59.

[3] 尹震宇, 赵海, 徐久强. WSN 中基于分簇路由的多维度数据压缩算法研究 [J]. 电子学报, 2009, 37 (5).

[4] Liang Y. Efficient temporal compression in wireless sensor networks [C]. Local Computer Networks. IEEE, 2011: 466 - 474.

[5] Aanchal, Kumar S, Kaiwartya O, et al. Green computing for wireless sensor networks: Optimization and Huffman coding approach [J]. Peer - to - Peer Networking and Applications, 2017, 10 (3): 1 - 18.

[6] 基于自适应睡眠机制的 WSN 能量高效协议 [J]. 计算机科学, 2015, 42 (4): 65 - 67.

[7] 瞿双, 等. 无线传感器网络中基于序列相关性的数据压缩算法 [J]. 电子与信息学报, 2016, 38 (3): 714 - 718.

[8] Börm S. Adaptive compression of large vectors [J]. Mathemat-

ics of Computation, 2017, 87 (309).

[9] I Yoon I, Yi J M, Jeong S, et al. Adaptive sensing and compression rate selection scheme for energy - harvesting wireless sensor networks [J]. International Journal of Distributed Sensor Networks, 2017, 13 (6).

[10] Marcelloni F, Vecchio M. An Efficient Lossless Compression Algorithm for Tiny Nodes of Monitoring Wireless Sensor Networks [M]. Oxford University Press, 2009.

[11] Kim S, Cho C, Park K J, et al. Increasing network lifetime using data compression in wireless sensor networks with energy harvesting [J]. 2017, 13 (1).

[12] 王怀光, 张培林, 吴定海, 等. 基于动态 LZ W 与算术编码的缓变信号无损压缩 [J]. 计算机应用研究, 2015 (9): 2742 - 2745.

[13] 陈田, 易鑫, 王伟, 等. 一种低功耗双重测试数据压缩方案 [J]. 电子学报, 2017, 45 (6): 1382 - 1388.

[14] Heinzelman W R, Chandrakasan A, Balakrishnan H. Energy - efficient communication protocol for wireless Micro - sensor Networks [A]. System Sciences, 2000. Proceedings of the 33rd Annual Hawaii International Conference [C]. 2000: 1 - 10.

[15] Elkamel R, Cherif A, Elkamel R, et al. Energy - efficient routing protocol to improve energy consumption in wireless sensors networks: Energy Efficient Protocol in WSN [J]. International Journal of Communication Systems, 2017 (6): e3360.

(上接第 272 页)

表 1 双路 TEC 温度控制试验记录

时间	晶体温度	温度误差	泵浦温度	温度误差
0	24.9	-0.1	24.9	-0.1
15	25.0	0	24.9	-0.1
30	24.9	-0.1	24.9	-0.1
45	25.1	0.1	25.0	0
60	25.0	0	25.0	0

注: 室温 20℃, 时间 min, 温度单位 ℃。

表 2 激光器输出光功率试验记录

时间	光功率/mW	LD 电流/A
0	79.9	0.99
15	80.1	1.00
30	79.8	0.98
45	80.1	1.01
60	80.0	1.00

测试结果表明, 激光器电源驱动电流误差不大于 2%, 且激光器光功率稳定度误差不超过 0.5%, 电源驱动性能良好, 长时工作稳定性强。

5 结束语

本文设计并实现了一种带有双路 TEC 温控的功率可调 DFB 激光器驱动电源, 实现了激光器双路温度控制及输出功率连续调节。经充分测试结果表明, 该驱动电源体积小、温度及电流控制精度高、光功率稳定性好。内部集成 TEC 及 LD 驱动模块, 扩展性强、稳定性高。相较传统驱动电

源, 大大减小了设计体积, 进一步提高了温度及电流控制精度, 在激光器驱动领域应用前景广阔。此外, 该设计方案对其他恒流源设计也有参考借鉴意义。

参考文献:

[1] 杨国文. 基于嵌入式系统的半导体激光器驱动设计 [D]. 南京: 南京大学, 2013.

[2] Deaton D J. High power density pulse generator for laser diode applications [C]. DRS Test and Energy Management Inc. Deaton - SSDLTR 2006.

[3] 王启明. 中国半导体激光器的历次突破与发展 [J]. 中国激光, 2010, 37 (9): 2190 - 2197.

[4] 陈雪芳. 基于 STM32F103 的电子相框的设计. 广东: 东莞理工学院学报, 2012, 19 (1): 19 - 52.

[5] 江剑平. 半导体激光器 [D]. 北京: 电子工业出版社, 2000.

[6] 董翠霞. 中小功率半导体激光器脉冲驱动电源的研制 [D]. 哈尔滨: 哈尔滨工业大学, 2007: 15 - 21.

[7] 黄德修, 刘雪峰. 半导体激光器及其应用 [M]. 北京: 国防工业出版社, 1999.

[8] 李适民, 黄维玲. 激光器件原理与设计 [M]. 北京: 国防工业出版社, 2005.

[9] 宋传磊. 半导体激光器驱动电源及其调控 [D]. 哈尔滨: 哈尔滨工业大学, 2006.

[10] 韦文生. 980 nm 半导体激光二极管的温度特性 [J]. 激光与红外, 2006, 36 (7): 558 - 560.

[11] 周以琳, 李金亮, 杨勇, 等. NTC 热敏电阻 R - T 特性的高精度补偿 [J]. 青岛科技大学学报, 2010, 31 (1): 80 - 82.

[12] 刘澄. 半导体激光器的浪涌损坏及其消除方法 [J]. 电力环境保护, 2003, 19 (4): 49 - 51.