



基于BP神经网络模型时钟同步误差补偿算法

魏连锁 李华 吴迪 郭媛

Clock synchronization error compensation algorithm based on BP neural network model

Wei Lian-Suo Li Hua Wu Di Guo Yuan

引用信息 Citation: *Acta Physica Sinica*, 70, 114203 (2021) DOI: 10.7498/aps.70.20201641

在线阅读 View online: <https://doi.org/10.7498/aps.70.20201641>

当期内容 View table of contents: <http://wulixb.iphy.ac.cn>

您可能感兴趣的其他文章

Articles you may be interested in

单光子激光测距的漂移误差理论模型及补偿方法

Theoretical model and correction method of range walk error for single-photon laser ranging

物理学报. 2018, 67(6): 064205 <https://doi.org/10.7498/aps.67.20172228>

不确定分数阶时滞混沌系统自适应神经网络同步控制

Synchronization of uncertain fractional-order chaotic systems with time delay based on adaptive neural network control

物理学报. 2017, 66(9): 090504 <https://doi.org/10.7498/aps.66.090504>

基于经验知识遗传算法优化的神经网络模型实现时间反演信道预测

Prediction of time reversal channel with neural network optimized by empirical knowledge based genetic algorithm

物理学报. 2019, 68(17): 170503 <https://doi.org/10.7498/aps.68.20190327>

基于遗传算法优化卷积长短记忆混合神经网络模型的光伏发电功率预测

A hybrid model for photovoltaic power prediction of both convolutional and long short-term memory neural networks optimized by genetic algorithm

物理学报. 2020, 69(10): 100701 <https://doi.org/10.7498/aps.69.20191935>

基于深度卷积神经网络的大气湍流相位提取

Extracting atmospheric turbulence phase using deep convolutional neural network

物理学报. 2020, 69(1): 014209 <https://doi.org/10.7498/aps.69.20190982>

变频正弦混沌神经网络及其应用

Frequency conversion sinusoidal chaotic neural network and its application

物理学报. 2017, 66(9): 090502 <https://doi.org/10.7498/aps.66.090502>

基于 BP 神经网络模型时钟同步误差补偿算法*

魏连锁 李华[†] 吴迪 郭媛

(齐齐哈尔大学计算机与控制工程学院, 齐齐哈尔 161000)

(2020 年 10 月 5 日收到; 2020 年 12 月 29 日收到修改稿)

误差补偿是保证水下传感器网络时钟同步精度的一个重要保障, 现有研究方法主要采用线性拟合和最小二乘法对时钟同步参数进行误差补偿, 但该方法并未考虑受海流影响时节点移动所导致的时钟同步精度问题. 针对此问题, 本文提出一种基于 BP 神经网络模型的时钟同步误差补偿算法. 首先采用深海拉格朗日洋流模型描述水下节点运动规律, 模拟水下节点运动速度, 进而建立时钟同步参数模型, 最后构建符合水下环境的 BP 神经网络时钟同步误差补偿模型, 通过定义激励函数, 引入正则项因子和补偿性因子避免模型过拟合, 建立误差反向传播的 BP 神经网络模型时钟同步误差补偿算法. 仿真实验表明, 本文提出的算法与 TSHL 算法、MU-sync 算法、MM-sync 算法相比, 在时钟同步精度 (即时钟同步时间与标准时间的误差) 上分别提升了 37.42%, 17.29% 和 21.86%, 并且均方误差得到显著降低.

关键词: 误差补偿, 时钟同步, BP 神经网络, 水下传感器网络**PACS:** 42.81.Pa**DOI:** 10.7498/aps.70.20201641

1 引言

海洋是未来世界各国竞争的核心基地^[1-3], 拥有大量的资源, 对海洋资源的开发利用离不开水下无线传感器网络 (underwater wireless sensor networks, UWSNs), UWSNs 是由许多水下节点构成的无线网络, 该网络在人类获取海洋信息中扮演着不可忽视的角色. 同时时钟同步对水下传感器网络有着非常重要的意义, 时钟同步对传感器网络许多关键技术起着非常重要的作用, 如: 节点定位、数据融合等^[4-6], 但由于水下环境条件恶劣、声波通信受到较多不确定因素影响, 目前现有的时钟同步研究成果在误差补偿方面仍存在如下挑战: 1) 水下节点会随着海洋或者潮汐的运动而发生移动, 故水下节点具有移动性; 2) 单向同步数据报文算法交换少, 时钟频偏估算精度高, 但相偏估算误

差大; 3) 双向同步数据报文算法时钟相偏估计精度高, 但数据报文交换多次, 时钟频偏估算误差较大, 而且传输延迟不一致也降低了时钟相偏的估算精度; 4) UWSNs 普通节点能源有限, 传输距离过长导致能耗高.

水下无线传感器网络节点具有移动性, 导致网络节点始终处于时变状态. 在同步转发过程中, 节点发送和接收数据报文的位置会发生变化, 从而产生相对移动距离, 导致动态时延增大. 如此随着转发次数增多, 时钟的误差 (主要由传送时间、传播时间和接收时间组成) 也逐渐增大, 造成水下传感器无线网络同步精度 (节点估算的参考时间与实际参考时间的偏差) 逐渐降低. 现存的水下无线传感器网络时钟同步算法, 并没有充分考虑到节点随洋流运动而产生的动态时延, 只是利用时间戳机制求解时钟同步参数, 然后利用传统的线性拟合对同步参数进行求精处理, 如 TSHL^[7], MM-sync^[8] 和

* 国家自然科学基金 (批准号: 61872204)、黑龙江省自然科学基金 (批准号: LH2019F037)、黑龙江省教育厅专项 (面上项目) (批准号: 135409312) 和研究生创新科研项目 (批准号: YJSCX2020009, YJSCX2019072) 资助的课题.

[†] 通信作者. E-mail: 1769348608@qq.com

MU-sync^[9] 等算法. 对动态时延的精确求解是同步精度的一个关键因素. 利用传统的优化算法 (如最小二乘法) 对同步参数进行求精, 容易陷入局部最优, 进而使得同步精度不高. 因此现有传统的时钟同步算法研究不能很好地解决由于节点移动性导致的时钟同步精度问题.

因此, 在设计 UWSNs 时钟同步算法时, 时钟同步参数的精度是水下传感器网络时钟同步算法需要着重考虑的因素, 为了提高时钟同步的参数精度, 需要对同步误差进行补偿.

本文的篇章组织结构如下: 第 1 节介绍 UWSNs 时钟同步算法的研究意义及误差补偿的重要性; 第 2 节详细给出有时钟同步算法及研究现状; 第 3 节给出了各种模型的搭建过程; 第 4 节详细介绍 BP 神经网络模型时钟同步误差补偿算法; 第 5 节通过仿真实验证明该误差补偿模型的实用性; 第 6 节对整篇文章进行分析总结.

2 相关工作

目前, 国内外学者对陆地无线传感器网络时钟同步算法的研究取得了一些成果, 由于水下传感器网络独特的传播特点, 使得现有的陆地传感器时钟同步算法 (如: RBS 算法, TPSN 算法, DMTS 算法)^[10–12] 并不适用于水下复杂的环境.

针对水下无线传感器网络时延的随机性, 研究者们提出了一系列适用于水下传感器网络的时钟同步算法. Eidenbenz 等提出了高延迟时钟同步算法 TSHL, 该算法首先利用信标节点发送数据报文, 然后待同步节点将收到报文的时间与本地时间组成元组, 再利用线性拟合计算频偏, 最后根据一次双向数据报文求解相偏, 该算法解决了高时延的问题, 但是因为需要多次发送数据报文, 所以存在能耗较高的问题, 同时该算法忽略了水下节点移动的特点, 因此算法的时钟精度不高. 与此同时, 该算法用的是线性拟合求解方法, 其误差较大. Sajjad 等^[13] 在分析时间间隔对时钟同步精度影响的基础上, 增大了 TSHL 算法的同步信号的时间间隔, 减少了同步信号的发送次数, 这种改进版的 TSHL 算法降低了算法复杂度和能量消耗. 文献^[14] 针对 TSHL 算法同步交互信号多, 能量消耗大的不足, 提出了一个 Tri-Message 同步算法, 首先改用 3 次发送-接收交互计算频偏, 然后使用双向交换计

算相偏, 这可以降低消息开销并保持一定同步精度, 进一步优化了 TSHL 算法的性能. 文献^[15] 提出 TSMU 算法, 利用卡尔曼滤波对多普勒计算的相对速度求其精度, 再用线性回归和校准来计算时钟同步的频偏和相偏. 文献^[16] 提出的 Hybrid-Sync 同步算法, 利用混合时钟交换数据报文的方式计算相偏和频偏, 最后利用计算得到的频偏来进一步校正相偏. 与 TSHL 算法相比, 降低了算法的通信费用, 但是 Hybrid-Sync 的计算费用较高且时钟同步精度不高.

上述算法虽然在一定程度上解决了时钟同步问题, 但都没有考虑节点移动性导致时钟同步精度不高的问题.

MM-sync 算法考虑传感器网络移动性对同步参数的影响, 推出时钟同步参数模型, 进而构建出一种适合浅水域的时间同步算法. 文献提出的一个分簇 UWSNs 时钟同步算法 MU-Sync. 该算法先对网络节点进行分簇, 再利用簇首周期性地广播同步信号来进行簇内时钟同步, 利用簇首节点收集到的同步信息进行两次线性拟合估计往返传播时延, 以此来降低误差, 通过多次双向信息交换, 使 MU-Sync 算法获得了较好的同步精度. Shi 等^[17] 首次将水下时钟同步问题分为授权、簇内同步和簇间同步三个阶段; 在授权阶段, 所有的节点相互授权, 并确认和删除恶意节点; 在簇内同步阶段, 簇首节点通过发送、接收模式和普通节点进行同步. 该算法为了提高同步精度, 利用 CESVM (centered hyperellipsoidal support vector machine based anomaly detection)^[18] 过滤非正常的接收端到接收端的传播延迟, 同时假设随机传输延迟符合方差为 0 的高斯分布, 进行时钟同步求精.

文献^[19] 提出 D-sync 时钟同步算法, 是基于水下通信产生的多普勒效应, 利用该效应计算节点间的相对速度, 估计传播延迟完成时钟同步. 但由于水声环境影响因素过多, 该算法测量的误差会较大. 因此, NU-sync 算法^[20] 加入了更准确的多普勒测量的方法因素, 通过计算节点的相对速度解决延迟多变的问题, 利用潜水器作为信标节点对节点进行同步, 极大降低了误差.

综上所述, 现今水下无线传感器网络时钟同步算法的研究需要综合考虑节点移动对同步精度的影响, 同时也要避免传统时钟同步算法使用线性拟合思想造成误差较大的问题. 因此, 本文综合考虑

节点的移动性、误差补偿等因素构建符合水下网络的 BP 神经网络时钟同步误差补偿算法, 从而设计出一个精度高、误差小的水下传感器网络时钟同步算法.

3 时钟同步误差参数补偿模型的建立与学习

3.1 洋流模型

本节首先分析洋流运动对节点运动的影响, 然后根据速度计算公式计算待同步节点的移动速率.

由于水下环境复杂多变, 水下物体运动并不是完全无规律的, 而是呈现出半周期的运动特性. 文献 [21] 中提出一种权威的洋流模型, 本文用此模型对节点的运动进行分析. 模型公式为

$$\begin{cases} \varphi(x, y, t) = \\ -\tanh\left[\frac{y - B(t)\sin(k(x - ct))}{\sqrt{1 + k^2 B(t)^2 \cos^2(k(x - ct))}}\right] \\ + ly, \\ B(t) = R + \varepsilon \cos(\omega t), \end{cases} \quad (1)$$

其中, l 表示位移率, $B(t)$ 表示曲线的宽度, ω 表示洋流场的频率, ε 表示振幅. $R = 1.2$, $l = 0.12$, $\omega = 0.4$, $\varepsilon = 0.3$, $k = 2\pi/7.5$ [22]. 一般情况下只考虑节点在 x 和 y 方向上的位移, 节点在水平方向的速率为

$$\begin{cases} u = -\frac{\partial \varphi}{\partial y} + \sigma u(t), \\ v = \frac{\partial \varphi}{\partial x} + \sigma v(t). \end{cases} \quad (2)$$

节点在 x 和 y 方向的运动情况如图 1、图 2 所示.

对普通节点来说, 难与浮标节点通信, 但利用

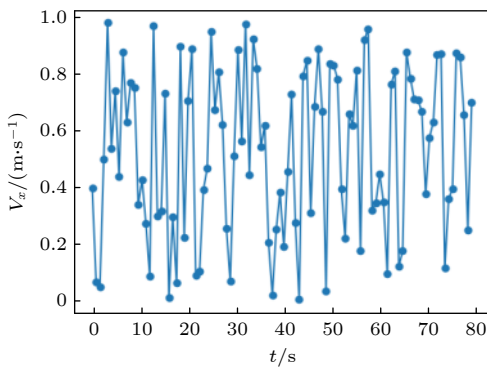


图 1 x 方向的速率

Fig. 1. The velocity in the x direction.

自身的相关性, 可以获取移动速度. 假定节点 i 想要取得自身的速度 $[v_x(i), v_y(i)]$, 其中 $v_x(i), v_y(i)$ 分别是 x 轴和 y 轴上的速度分量. 如果节点 i 能够获取附近簇首节点的速度, 就可以使用 (3) 式计算得到自身速度:

$$\begin{cases} v_x(i) = \sum_{j=1}^m \xi_{ij} v_x(j), \\ v_y(i) = \sum_{j=1}^m \xi_{ij} v_y(j), \end{cases} \quad (3)$$

其中, m 为邻近簇首节点的数量, ξ_{ij} 为差值系数, 可以用 (4) 式得到:

$$\xi_{ij} = \frac{1}{\sum_{i=1}^m \frac{1}{r_{ij}}}, \quad (4)$$

其中 r_{ij} 为簇首节点与普通节点的距离, 可以使用欧几里得 [23] 计算.

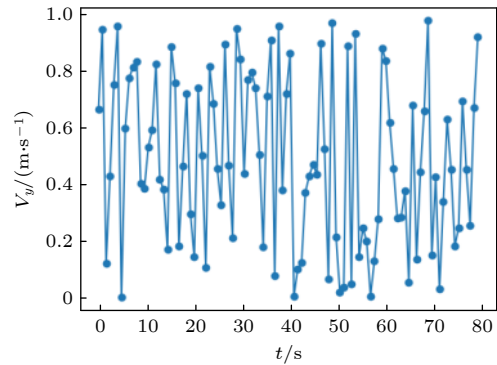


图 2 y 方向的速率

Fig. 2. The velocity in the y direction.

3.2 时钟同步参数模型

本文传感器节点以线性模型为例, 根据文献 [24] 节点的时钟 t 与标准时间 T 的关系为

$$T = at + b, \quad (5)$$

其中, a 表示时钟漂移, b 表示时钟偏移.

本文采用先分簇后对簇群进行时钟同步, 分为浮标节点、簇首节点、普通节点. 簇首节点可以直接获取浮标节点的标准时间, 故只需要考虑簇群的时钟同步. 报文交换过程如图 3 所示.

普通节点分别向各个节点发送数据报文进行时间同步, 簇首节点收到后向普通节点发送多个数据分组. 信息交互过程如图 3 所示. C 在 T_1 时刻向

簇首节点发起同步进程, Q 在 T_2 时刻收到该信息, 并随即向节点 C 发送多个数据包, 其时间戳用 $t[i]$ 表示, C 收到数据报文的时间戳用 $T[i]$ 表示. 当节点 Q 第一次发送数据报文的发送时间是 $t[1]$, 经过时间 d_1 后, 到达节点 C . 节点 C 记录下到达本地的时间 $T[1]$. 过一段时间后, 节点 C 再向节点 Q 发送同步数据报文, 由于节点受到洋流的影响会发生移动, 故数据报文传播延迟不再是 t_1 而是 t_2 . 则可以

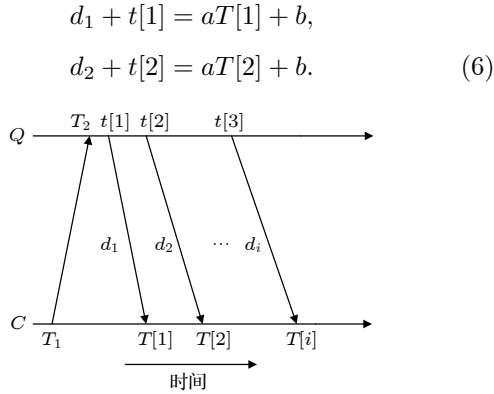


图 3 数据报文交换过程

Fig. 3. Data message exchange process.

由于水下节点具有移动性, 故假设簇首 Q 的移动速率为 V_{mi} , 待同步节点 C 的移动速率为 V_{Ai} , 水声传播速度为 u , 则可得

$$ud_i = ud_{i-1} + V_{mi}(d_{i-1} + d_i) + V_{Ai}(d_{i-1} + d_i), \quad (7)$$

整理后可得:

$$d_i = \frac{ud_{i-1} + v_{mi} + v_{Ai}d_{i-1}}{u - v_{mi} - v_{Ai}}, \quad (8)$$

其中, V_{mi} 和 V_{Ai} 可以由 (3) 式计算得到. 求出 d_i 后, 漂移可由 (8) 式求得:

$$\begin{aligned} \mathbf{a} &= [a_1, a_2, a_3, \dots, a_m]^T \\ &= \begin{bmatrix} a_1 = \frac{t[1] - t[2] + d_1 - d_2}{T[1] - T[2]} \\ a_2 = \frac{t[2] - t[3] + d_2 - d_3}{T[2] - T[3]} \\ a_3 = \frac{t[3] - t[4] + d_3 - d_4}{T[3] - T[4]} \\ \vdots \\ a_i = \frac{t[i] - t[i+1] + d_i - d_{i+1}}{T[i] - T[i+1]} \end{bmatrix}. \end{aligned} \quad (9)$$

收集的时间戳为 T_a, T_b, T_c, T_d , 如图 4 所示, 并使用

下式计算出频偏:

$$\begin{cases} aT_a + b_1 + t_1 = T_{b1}, \\ T_c + t_2 = aT_d + b, \\ t_2 - t_1 = aT_d + aT_a + 2b_1 - T_{b1} - T_c, \\ t_2 - t_1 = va(t_d - t_a)/u. \end{cases} \quad (10)$$

其中 \mathbf{a} 是 (9) 式中的列矩阵, 考虑到水下传感器网络时钟同步的时效性精准性, T_{b1} 和 b_1 只有在计算频偏的时候才会使用. 由 (10) 式可推出:

$$\begin{aligned} \mathbf{b} &= [b_1, b_2, b_3, \dots, b_m]^T = \\ &= \begin{bmatrix} b_1 = \frac{u(T_{a1} + T_{d1}) - a_1[u(T_{b1} + T_c) + v(T_{d1} - T_{a1})]}{2u} \\ b_2 = \frac{u(T_{a2} + T_{d2}) - a_2[u(T_{b2} + T_c) + v(T_{d2} - T_{a2})]}{2u} \\ b_3 = \frac{u(T_{a3} + T_{d3}) - a_3[u(T_{b3} + T_c) + v(T_{d3} - T_{a3})]}{2u} \\ \vdots \\ b_i = \frac{u(T_{ai} + T_{di}) - a_i[u(T_{bi} + T_c) + v(T_{di} - T_{ai})]}{2u} \end{bmatrix}, \end{aligned} \quad (11)$$

其中 T_{bi} 和 T_{ai} 是收集的一系列剪短时间戳, 主要是为了更精准地求出频偏. 本次时钟同步通过单向数据报文交换来减少双向交换的次数, 从而减少数据报文能耗, 但没有考虑时延带来的误差, 导致 $\{a, b\}$ 精度不高. 本文将针对该误差提出补偿模型.

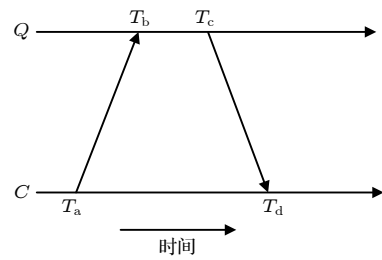


图 4 数据报文交换过程

Fig. 4. Data message exchange process.

3.3 BP 神经网络时钟同步误差补偿模型

基于 BP 神经网络的时钟同步误差补偿模型, 是将根据时钟同步参数模型中的 (9) 式和 (11) 式计算得到的 m 个同步参数估计值作为网络学习样本, 经过大量重复不断地学习, 沿着误差减小的方向, 不断调整各层神经元之间的权值和阈值, 目的是得到一个最佳预测补偿模型.

针对 3.2 节时钟同步参数模型精度不高提出的 BP 神经网络时钟同步误差补偿模型如图 5 所示。

3.3.1 激活函数的设定

激活函数对本文算法网络尤为重要, 常用的 sigmoid 和 tanx 虽然易于实现, 但存在不可避免的缺陷. 基于此本文为避免陷入局部计算以及综合节点移动对水下传感器时钟同步产生的影响, 本文提出 $f(x)$ 作为激活函数:

$$f(x) = \begin{cases} x, & x \geq 0, \\ \phi x, & x < 0, \end{cases} \quad (12)$$

其中, $\phi \in (0, 1)$ 通过先验知识赋值.

3.3.2 代价函数、误差补偿模型和误差方向传播更新过程

1) 代价函数

本文有 m 个输入样本 $\{t = [t_1, t_2, t_3, \dots, t_i, \dots, t_m]\}$ 、 $\{a = [a_1, a_2, a_3, \dots, a_i, \dots, a_m]\}$ 、 $\{b = [b_1, b_2, b_3, \dots, b_i, \dots, b_m]\}$, $d(i)$ 为期望输出, 输出向量 $g = \{g_1, g_2, \dots, g_k, \dots, g_m\}$, 某层神经元输出: $h^{(l)} = \{h_1^{(l)}, h_2^{(l)}, \dots, h_j^{(l)}, \dots, h_{sl}^{(l)}\}$. 本文属于线性回归的问题, 故采用平方误差 $E = \frac{1}{m} \sum_{t=1}^m E(i)$ 作为代价函数.

2) 时钟同步误差补偿模型

为避免过拟合现象造成的误差无法达到极小值, 参数的更新过程采用随机梯度下降算法. 考虑到水下传感器节点资源有限并结合 BP 神经网络结构, 构造时钟同步误差补偿模型为:

其中

$$P = \frac{\lambda}{2} P_1 + \sigma P_2 = \kappa \sum_{l=1}^{L-1} \sum_{j=1}^{s_l} \sum_{i=1}^{s_{l+1}} w_{ij}^{(l)} + \eta \sum_{k=1}^{h_k} \sum_{j=1}^{s_k} KL(\chi || \hat{\chi}_j^k), \quad (14)$$

其中, P_1 表示正则化项惩罚因子, P_2 表示补偿性因子, $||$ 表示交叉熵运算.

$$E(i) = \frac{1}{2} \sum_{k=1}^n (d_k(i) - g_k(i))^2, \quad (15)$$

$$P = \frac{\lambda}{2} P_1 + \sigma P_2, \quad (16)$$

P_1, P_2 计算方法如下:

$$P_1 = \sum_{l=1}^{L-1} \sum_{j=1}^{s_l} \sum_{i=1}^{s_{l+1}} (w_{ij}^{(l)})^2, \quad (17)$$

$$P_2 = \sum_{k=1}^{h_k} \sum_{j=1}^{s_k} KL(\chi || \hat{\chi}_j^k) = \sum_{k=1}^{h_k} \sum_{j=1}^{s_k} \tau \log \frac{\tau}{\hat{\tau}_j^{(k)}} + (1 - \tau) \log \frac{1 - \tau}{1 - \hat{\tau}_j^{(k)}}, \quad (18)$$

其中, h_k 为隐藏层个数, τ 为稀疏性因子, 满足 $\tau \in (0, 1)$,

$$\hat{\tau}_j^{(k)} = \frac{1}{m} \sum_{i=1}^m a_j^{(k)}(x_i). \quad (19)$$

3) 误差方向传播参数迭代更新过程

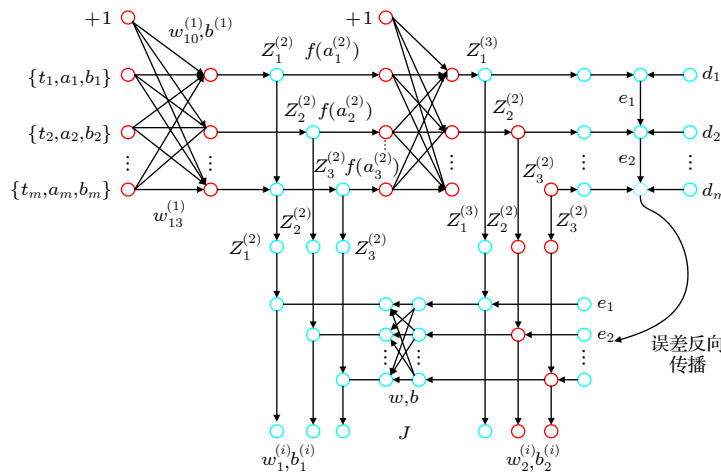


图 5 BP 神经网络时钟同步误差补偿结构图

Fig. 5. BP neural network clock synchronization error structure diagram.

$$w_{ij}^{(l)} = \frac{\partial J}{\partial w_{ij}^{(l)}} = w_{ij}^{(l)} - \alpha \frac{\partial E}{\partial w_{ij}^{(l)}} + \frac{\partial P}{\partial w_{ij}^{(l)}}, \quad (20)$$

$$b_i^{(l)} = \frac{\partial J}{\partial b_i^{(l)}} = b_i^{(l)} - \alpha \frac{\partial E}{\partial b_i^{(l)}} + \frac{\partial P}{\partial b_i^{(l)}}, \quad (21)$$

其中 $\alpha \in (0, 1)$ 是学习率. 为提高收敛速度, 从局部极小值中跳出来, 根据文献 [25] 提出优化学习率的变换公式:

$$\alpha = Ge^{-\lambda r}. \quad (22)$$

式中, r 是迭代次数, G 和 λ 是常数, 根据本文的学习样本, 选取 $G = 20$, $\lambda = 0.002$.

另外,

$$\begin{aligned} \frac{\partial E(i)}{\partial w_{kj}^{(l)}} &= \frac{\partial}{\partial w_{kj}^{(l)}} \left\{ \frac{1}{2} \sum_{k=1}^n [d_k(i) - g_k(i)]^2 \right\} \\ &= -[d_k(i) - g_k(i)] \frac{\partial g_k(i)}{\partial Z_k^{(L)}} \frac{\partial Z_k^{(L)}}{\partial w_{kj}^{(L)}} \\ &= -[d_k(i) - g_k(i)] f(x)' \Big|_{x=Z_k^{(L)}} \frac{\partial Z_k^{(L)}}{\partial w_{kj}^{(L)}} h_j^{(L-1)} \\ &= \sum_{k=1}^{s_{l+1}} w_{kj}^{(l+1)} [d_k(i) - g_k(i)] f(x)' \Big|_{x=Z_k^{(l+1)}} \\ &\quad \times f(x)' \Big|_{x=Z_j^{(i)}} h_i^{(l-1)}. \end{aligned} \quad (23)$$

同理可得:

$$\begin{aligned} \frac{\partial E(i)}{\partial b_j^{(l)}} &= \sum_{k=1}^{s_{l+1}} w_{kj}^{(l+1)} [d_k(i) - g_k(i)] f(x)' \Big|_{x=Z_k^{(l+1)}} \\ &\quad \times f(x)' \Big|_{x=Z_j^{(i)}}, \end{aligned} \quad (24)$$

$$\frac{\partial P}{\partial w_{ij}^{(l)}} = \frac{1}{m} \delta^{(l)} a^{(l)\top} + \lambda w^{(l)}, \quad (25)$$

$$\frac{\partial P}{\partial b^{(l)}} = \frac{1}{m} \delta^{(l)} a^{(l)\top}. \quad (26)$$

4 BP 神经网络模型时间同步误差补偿算法

本文构建的 BP 神经网络时钟同步误差补偿模型是一种正向传递和误差反向传播的网络 [26], 采用随机梯度下降原理对参数进行调节, 经过多次反复训练就可以得到满足补偿条件的 BP 神经网络. 具体过程为: 模型搭建阶段、模型学习阶段、模型预测阶段.

4.1 模型搭建阶段

如表 1 所示, 输入层节点个数设置为 2, 输出层节点的个数设置为 1, 隐藏层的神经元个数使用了经验公式 [27]

$$l = \sqrt{q+s} + c, \quad (27)$$

式中 q 是输入层神经元的个数, s 是输出层神经元的个数, $c \in [1, 10]$, 根据本文实际需求与经验公式 (27) 式可知, 隐藏层的层数设置为 2, 神经元个数为 6 个. 收集学习样本, 利用时钟同步参数模型输出得到的 $m = \{a_i, b_i\}$ 组学习样本作为数据集, 再对学习样本进行数据归一化.

表 1 模型搭建阶段
Table 1. Model construction phase.

算法: 基于BP神经网络模型时钟同步误差补偿算法	
$\mathbf{a} = [a_1, a_2, a_3, \dots, a_i, \dots, a_m], i = 1, 2, \dots, m$	
输入: $\mathbf{b} = [b_1, b_2, b_3, \dots, b_i, \dots, b_m], i = 1, 2, \dots, m$	
$\mathbf{t} = [t_1, t_2, t_3, \dots, t_i, \dots, t_m], i = 1, 2, \dots, m$	
阶段1: BP模型搭建阶段	
01: 算法	
02: 算法阶段一开始	
03: 数据收集	
	$m \leftarrow \{\mathbf{a} = [a_1, a_2, a_3, \dots, a_i, \dots, a_m],$
04: $\mathbf{b} = [b_1, b_2, b_3, \dots, b_i, \dots, b_m],$	
	$\mathbf{t} = [t_1, t_2, t_3, \dots, t_i, \dots, t_m]\}$
05: 数据归一化	
06: $\tilde{m} \leftarrow m$	
07: for $c = 1, c \leq 10, c++$	
08: 根据 $l = \sqrt{q+s} + c$ 计算隐藏层的神经元的个数	
09: BP神经网络模型	
10: end for	
11: end	

4.2 模型训练阶段

如表 2 所示, 分别对 $w \in (0, 1), b \in (0, 1), \alpha \in (0, 1), \lambda \in (0, 1), \sigma \in (0, 1)$ 随机赋初值, 同时设迭代次数 $r = 3000$, $\rho = 0.001$. 使用随机分成的训练集对误差补偿网络模型进行训练, 然后再用测试集对网络模型进行预测检测. 结合水下传感器网络时钟同步的特点对激活函数重新定义, 最终设为 (12) 式. 设定时钟同步误差补偿模型 J . 然后开始对误差补偿网络进行训练, 将训练集输入到输入层神经元, 然后逐层将信号正向传播并计算输出, 直到

有输出层的结果并计算输出层的误差,再利用误差反向传播并根据 (20) 式、(21) 式更新各层神经元的权重和阈值. 反复进行该过程直到满足迭代终止条件才停止训练,得到的误差补偿模型参数几乎不再发生变化. 此时得到一个满足误差补偿条件且精度度较高的 BP 神经网络时钟同步模型 J .

表 2 模型训练阶段
Table 2. Model training stage.

阶段2: BP网络模型训练阶段	
01:	算法开始
02:	开始训练网络
03:	参数初始化
04:	随机设置 $w \in (0, 1), b \in (0, 1), \alpha \in (0, 1), \lambda \in (0, 1), \sigma \in (0, 1)$
05:	$r = 3000, \rho = 0.001$
06:	create $f(x), P$
07:	\tilde{m} 分成训练集和测试集
08:	start training
09:	for all $\{t_i, a_i, b_i\} \in \tilde{m}$ do
10:	if $(E > \rho)$ do
11:	w, b 被更新 $w_{ij}^{(l)}, b_i^{(l)}$ according to Eq. (20) and Eq. (21)
12:	end if
13:	end for
14:	输出 J
15:	end

4.3 模型预测阶段

如表 3 所示, 模型预测阶段具体过程如下: 当网络达到迭代终止条件, 即 E 小于期望误差 ρ 时, 得到一个满足条件的补偿模型 J . 使用测试集对

表 3 模型预测阶段
Table 3. Model prediction stage.

阶段3: BP网络预测	
01:	开始
02:	算法预测
03:	当满足条件时, 执行以下
04:	if $(E < \rho)$ do
05:	停止训练
06:	得到 J
07:	输入 $[a_{i+1}, b_{i+1}]$ to J
08:	输出 $[\tilde{a}_{i+1}, \tilde{b}_{i+1}]$
09:	end if
10:	end

J 进行预测, 最后只需将从时钟同步参数模型得到的第 $i+1$ 个同步估计参数 $[a_{i+1}, b_{i+1}]$ 输入 J 中, 通过输出得到时钟同步参数补偿值 $[\tilde{a}_{i+1}, \tilde{b}_{i+1}]$.

5 实验与结果分析

本文使用 python 语言, pycharm 集成开发环境, 做了算法时间复杂度分析和两种仿真实验描述. 实验一: 将本文提出的算法与传统线性拟合在误差控制和预测方面做对比, 分别计算均方误差并分析它们对水下传感器时钟同步误差补偿的效果. 实验二: 本文算法与 TSHL^[7] 算法、MM-sync^[8] 算法、MU-sync^[9] 算法在同步精度和数据报文能耗方面进行对比分析.

5.1 算法时间复杂度分析

时间复杂度是衡量一个算法的度量指标, 本节主要分析基于 BP 神经网络模型时钟同步误差算法. 本文算法的核心是网络的前向传播和误差反向调整权重和阈值部分. 前向信号传播时间复杂度为 $O(n)$, 误差反向传播的时间复杂度为 $O(m)$. 故本文算法的时间复杂度为 $O(n^2)$.

5.2 实验设置

本文仿真实验的相关参数如表 4 所示.

表 4 实验参数
Table 4. Experimental parameter setting.

仿真实验参数	符号表示	数值
浮标节点	N	2
节点布置区域/ m^3	O	$400 \times 400 \times 400$
普通传感器节点	m	200
迭代次数	r	3000
正则惩罚因子	λ	0.01
稀疏性惩罚因子	σ	0.03
稀疏性因子	τ	0.6
激活因子	ζ	0.4

5.2.1 本文算法和线性拟合作同步时间与标准时间的误差对比分析

为了比较 BP 神经网络时钟同步误差补偿的效果, 引入利用传统线性拟合对数据报文交换进行误差补偿^[28], 同时使用传统线性拟合对 m 组训练样本进行拟合.

由图6可知,精度在0.001时几乎看不出变化,因为使用BP神经网络模型的补偿效果几乎与标准时间一致,为了更好地看出BP补偿效果与传统补偿效果的区别,将其扩大 10^4 倍.开始时BP神经网络误差大于线性拟合,是因为开始时网络参数设置是随机设定.当节点个数达到5时,BP神经网络的补偿效果明显优于线性拟合.当节点个数达到90时,BP神经网络的误差补偿效果相比于线性拟合更加具有优势.

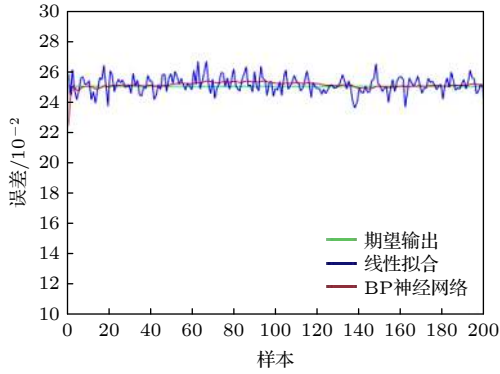


图6 误差对比分析

Fig. 6. Error comparison and analysis.

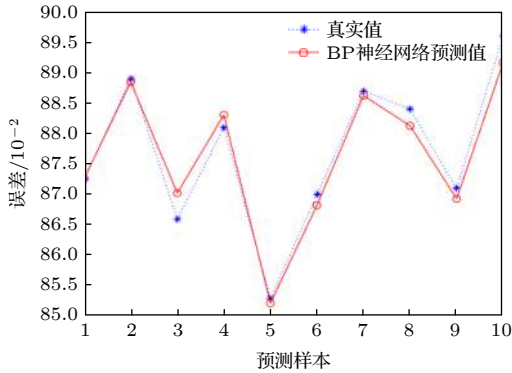


图7 BP神经网络预测

Fig. 7. BP neural network prediction.

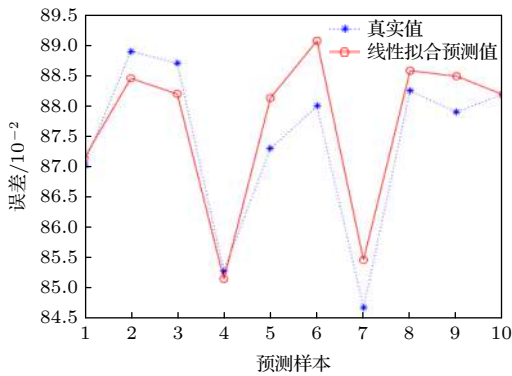


图8 线性拟合预测

Fig. 8. Linear fitting prediction.

同时本文使用10个样本对已经训练好的补偿模型和线性拟合作对比预测,对比它们之间的均方误差.

如图7和图8所示,节点一开始是随机抛洒,故需要将训练好的BP神经网络误差补偿模型和传统线性拟合去逼近节点真实值,通过比较并计算可得图7的均方误差 $R^2 = 0.6972$,图8的均方误差 $R^2 = 0.9116$.比较可得,使用BP神经网络时钟同步误差补偿模型的效果远高于线性拟合.

5.2.2 本文算法与其他时间同步算法的对比分析

本节针对时钟同步精度和数据报文能耗,将本文算法与TSHL^[7]算法、MU-sync^[9]算法、MM-sync^[8]算法进行对比实验分析.

图9和图10分别表示时钟频偏 a 计算偏差随节点移动速度的变化情况和时钟漂移 b 计算偏差随节点移动速度的变化情况.

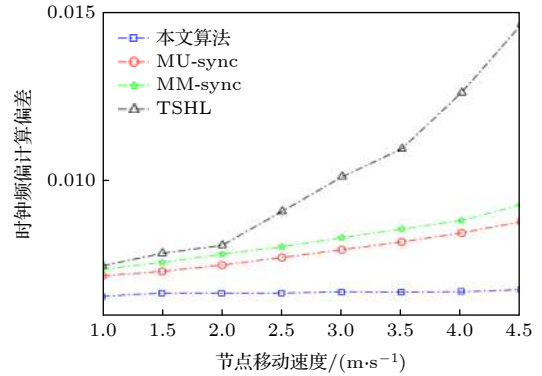


图9 时钟频偏计算偏差随节点移动的变化

Fig. 9. The variation of clock-frequency offset calculation deviation with node move.

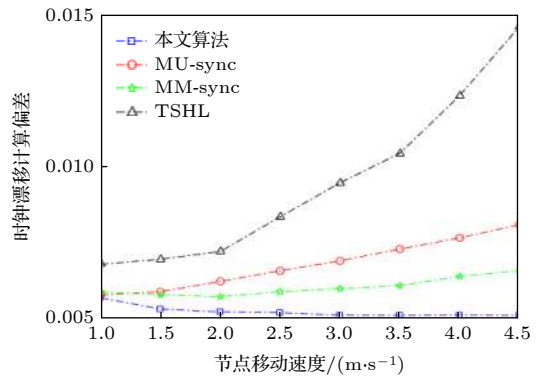


图10 时钟漂移计算偏差随节点移动变化情况

Fig. 10. The clock drift calculation deviation changes with the node movement.

由图 9 可知, 本文算法的误差明显小于其他算法, 分别比 TSHL^[7] 算法、MU-sync^[9] 算法、MM-sync^[8] 算法降低 37.42%, 17.29% 和 21.86%。因为随着节点速度增大, 而 TSHL 忽视了节点的移动性问题, 在时钟同步期间假定节点位置不变, 即传播过程时延没有发生变化, 导致速度增大时, TSHL 的误差会异常明显。MU-sync 是针对节点移动性提出的算法, 但忽视了传播时延, 采用不变的值去计算频偏, 再用线性拟合去除相应的传播时延, 从而求得时钟漂移, 当速度明显增大时, 算法表现就会减弱。MM-sync 算法在误差补偿中, 假设节点做匀速运动, 当节点速度增大时, 误差补偿效果减弱。本文算法基于 BP 神经网络的时钟同步误差补偿模型, 考虑了节点移动性带来的时延问题, 同时通过 BP 神经网络反复训练得到一个精度较高的误差补偿模型。

由图 10 可以看出, 本文算法明显优于其余三种算法, 当节点速度越大, 误差几乎不在发生变化, 其运动轨迹和移动模型更加符合实际, 并且时钟同步精度也更高, 因而计算偏差越小。

图 11 给出本文算法、MU-sync^[9] 算法、MM-sync^[8] 算法和 TSHL^[7] 算法在完成同步后, 本地时钟增长的误差趋势。

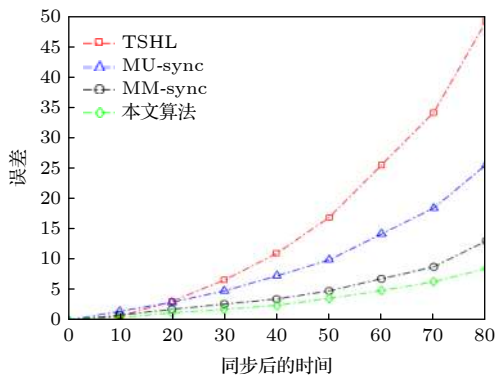


图 11 时钟同步后本地时钟误差增长趋势变化

Fig. 11. The increasing trend of local clock error changes after clock synchronization.

由图 11 可知, 本文算法的误差明显小于其他算法, 分别降低 47.12%, 31.37% 和 12.89%。其他算法误差较大的主要原因是: TSHL 法假设网络为静态网络, 这势必会导致更大的同步误差; MU-sync 使用两次线性回归来估计时钟频偏和相偏, 但该算法假设每一轮消息交换到的传播时延是一致的, 没有考虑往返时间不一致的传播时延, 这会

导致更大的同步误差; MM-sync 算法虽然考虑节点移动性, 但是在误差估计阶段, 将节点运动状态单一化的归结为匀速运动, 如此随着网络的动态变化, 该算法的性能也随之下降; 这三种算法并没过多考虑节点随洋流运动而产生的动态时延, 导致误差较大。本文算法通过引入洋流模型估计待同步节点的速率, 提高了动态传播延时的精度, 且使用 BP 神经网络反复训练, 得到一个误差补偿精度很高的模型。

图 12 给出本文算法与其余三种算法在同步 35 次后产生的数据报文数, 以此来分析数据报文能耗的高低。

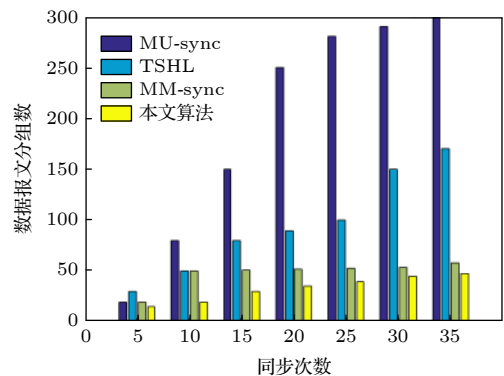


图 12 同步次数与数据报文的变化

Fig. 12. Changes in synchronization times and datagram.

由图 12 可知, 本文算法在能耗方面少于其他三种算法^[7-9]。分别比其余三种算法降低了 37.46%, 61.26% 和 18.75%。本文算法只需要 BP 神经网络反复训练即可, 与其余三种算法的原理不同, 需要的数据报文较少。随着同步次数的增加, TSHL 算法在同步过程的第二阶段需要参考节点和普通节点进行一次双向数据报文交换, 用来计算时钟漂移。MU-sync 算法为追求更高的精度, 通过增加了双向数据报文的交换次数, 导致当同步次数增加时, 数据报文量快速增加。MM-sync 需要提高线性拟合的次数来提高精度, 导致数据报文的消耗大。

6 总 结

水下传感器网络具有移动性、能耗有限、高时延和随机动态时延的特点, 给设计水下传感器网络节点时钟同步带来巨大挑战。本文创新性地提出利用 BP 神经网络对时钟同步误差进行补偿, 从而提高时钟同步精度。本文算法首先考虑节点随洋流运

动的影响, 采用深海拉格朗日洋流模型描述水下节点运动规律, 并算出待同步节点的运动速率, 进而建立时钟同步参数模型, 最后利用 BP 神经网络对时钟同步参数模型进行误差补偿. 与现有同步算法对比, 本文算法能高效补偿同步误差提高时钟同步精度, 且避免了节点的双向数据报文交换, 从而也降低了数据能耗. 与此同时, 由于本文算法考虑了节点受洋流运动的影响, 故节点在运动的情况下, 依然能有效进行误差补偿, 从而保持时钟一定的同步精度. 实验表明, 算法在水下复杂环境时候, 本文算法对水下传感器网络时钟同步具有很好地适用性, 同时也为时钟同步误差补偿的理论提供了新思路.

参考文献

- [1] MacKenzie A B, DaSilva L A 2019 *Sensors* **57** 36
- [2] Hao X C, Zhang Y X, Jia N, Liu B 2018 *Wirel. Pers. Commun.* **69** 1289
- [3] Li N, Hou J C, Sha L 2019 *Wirel. Commun.* **4** 1195
- [4] Peng H X, Zhao H, Li D Z, Lin C 2014 *Acta Phys. Sin.* **63** 090206 (in Chinese) [彭海霞, 赵海, 李大舟, 林川 2014 物理学报 **63** 090206]
- [5] Sun D Y, Qian Z H, Han M F, Wang X 2014 *Acta Electron. Sin.* **42** 1601 (in Chinese) [孙大洋, 钱志鸿, 韩梦飞, 王雪 2014 电子学报 **42** 1601]
- [6] Zhang C, Fei S M, Zhou X P 2018 *Chin. Phys. B* **21** 120101
- [7] A. A. Syed and J. Heidemann 2006 *25th IEEE International Conference on Computer Communications* Barcelona, Spain April 23–29, 2006 p161
- [8] Wang H Q, Weng X X, Lin J Y, Feng G S 2016 *Journal of Communications*. **17** 620 (in Chinese) [王慧强, 温秀秀, 林俊宇, 冯光升 2016 通信学报 **17** 620]
- [9] Chirdchoo, N, W. S. Soh, K. C. Chua 2008 *Proceedings of the Third Workshop on Underwater Networks (WUWNET)*, San Francisco, California, USA, September 15, 2008 p201
- [10] Yaghoubi F, Abbasfar A A, Maham B 2019 *IEEE Commun. Lett.* **18** 973
- [11] Wang X, Sheng M, Liu M, Zhai D, Zhang Y 2018 *Wireless Communications and Networking Conference (WCNC)* Shanghai, China, April 7–10, 2018 p1009
- [12] Sajjad Z, Nasser Y, Amir N 2018 *Comput. Netw.* **56** 902
- [13] Zou Y L, Zhu J, Zheng B Y, Tang S L 2010 *IEEE Global Telecommunications Conference (GLOBECOM 2010)* Miami Florida, USA, December 6–10, 2010 p1
- [14] Komali R S, MacKenzie A B 2016 *Proceedings of IEEE CCNC Las Vegas Conference* Arizona, USA, January 8–10, 2016 p563
- [15] Kim S, Yoo Y 2019 *INT J DISTRIB SENS N.* **8** 18
- [16] Yu K, Guo Y J, Hedley M 2018 *IET Signal Proc.* **3** 106
- [17] Laska J, Bradley W, Rondeau T 2011 *IEEE Symposium on New Frontiers in Dynamic Spectrum Access Networks* Aachen, Germany, May 3–6, 2011 p156
- [18] Shi M Z, Wang N Z, Lei B, Ying J J, Zhu C S, Sun Z L, Cui J H, Meng F B, Shang C, Ma L K, Chen X H 2018 *New J. Phys.* **20** 123007
- [19] Iyo A, Kawashima K, Kinjo T, Nishio T, Ishida S, Fujihisa H, Gotoh Y, Kihou K, Eisaki H, Yoshida Y 2016 *J. Am. Chem. Soc.* **138** 3410
- [20] Blomberg E, Tanatar M, Fernandes R, Mazin I, Shen B, Wen H H, Johannes M, Schmalian J, Prozorov R 2017 *Nat. Commun.* **4** 1914
- [21] Fang L, Luo H Q, Cheng P, Wang Z S, Jia Y, Mu G, Shen B, Mazin I I, Shan L, Ren C, Wen H H 2018 *Phys. Rev. B* **80** 140508
- [22] Shimojima T, Sakaguchi F, Ishizaka K, Ishida Y, Kiss T, Okawa M, Togashi T, Chen C T, Watanabe S, Arita M, Shimada K, Namatame H, Taniguchi M, Ohgushi K, Kasahara S, Terashima T, Shibauchi T, Matsuda Y, Chainani A, Shin S 2016 *Science* **332** 564
- [23] Ho K C, Lu X, Kovavisaruch L 2017 *IEEE Trans.* **55** 684
- [24] Wang C, Li Y, Zhu Z, Jiang S, Lin X, Luo Y, Chi S, Li L, Ren Z, He M, Chen H, Wang Y T, Tao Q, Cao G H, Xu Z A 2017 *Phys. Rev. B* **79** 054521
- [25] Newman M E J, Strogatz S H, Watts D J 2011 *Phys. Rev.* **64** 026118
- [26] Analytis J G, Kuo H, McDonald R D, Wartenbe M, Hussey N, Fisher I 2016 *Nat. Phys.* **10** 194
- [27] Kleinrock L, Silvester J 2008 *Proceedings of the National Telecommunications Conference* Birmingham Ala, USA, March 27, 2008 p10
- [28] Hayes I M, McDonald R D, Breznay N P, Helm T, Moll P J, Wartenbe M, Shekhter A, Analytis J G 2016 *Nat. Phys.* **12** 916

Clock synchronization error compensation algorithm based on BP neural network model^{*}

Wei Lian-Suo Li Hua[†] Wu Di Guo Yuan

(School of Computer and Control Engineering, Qiqihar University, Qiqihar 161000, China)

(Received 5 October 2020; revised manuscript received 29 December 2020)

Abstract

Error compensation is an important guarantee method to ensure the accuracy of clock synchronization in underwater sensor networks. Existing research methods mainly use linear fitting and least square method to compensate for clock synchronization parameters. Underwater wireless sensor network nodes are mobile, which leads the network nodes to be always in a time-varying state. In the process of synchronous forwarding, the position where the node sends and receives data packets will change, resulting in a relative moving distance, leading the dynamic delay to an increase in. In this way, as the number of forwarding nodes increases, the error of the clock gradually increases, causing the synchronization accuracy of the underwater sensor wireless network to gradually decrease. The existing underwater wireless sensor network clock synchronization algorithm does not fully consider the dynamic time delay caused by the movement of the node with the ocean current. It only uses the time stamp mechanism to solve the clock synchronization parameters, and then uses the traditional linear fitting to refine the synchronization parameters. The accurate solution of dynamic time delay is a key factor of synchronization accuracy. The use of traditional optimization algorithms to refine the synchronization parameters can easily fall into a local optimum, which makes the synchronization accuracy not high. Therefore, the existing traditional research on clock synchronization algorithms cannot well solve the problem of clock synchronization accuracy caused by node mobility. However this type of method does not consider the clock synchronization accuracy of node movement affected by ocean currents. To solve this problem, this paper proposes a clock synchronization error compensation algorithm based on BP neural network model. First, the deep-sea Lagrangian ocean current model is used to describe the movement of underwater nodes and simulate the movement speed of underwater nodes, and then a clock synchronization parameter model is established, and finally a BP neural network clock synchronization error compensation model is build, which conforms to the underwater environment, and the excitation function is defined, and regular term factor and compensatory factor are introduced to avoid model over-fitting. The BP neural network model clock synchronization error compensation algorithm is established for error back propagation. Simulation experiments show that compared with the comparison algorithm TSHL, MM-sync, and MU-sync, the accuracy of clock synchronization, namely the error between clock synchronization time and standard time, increased by 37.42%, 17.29% and 21.86%, and the mean square error is significantly reduced.

Keywords: error compensation, underwater sensor network, BP neural network, clock synchronization

PACS: 42.81.Pa

DOI: 10.7498/aps.70.20201641

^{*} Project supported by the National Natural Science Foundation of China (Grant No. 61872204), the Natural Science Foundation of Heilongjiang Province, China (Grant No. LH2019F037), the Special (General) Project of Heilongjiang Provincial Department of Education, China (Grant No. 135409312), and the Postgraduate Innovation Research Project, China (Grant Nos. YJSCX2020009, YJSCX2019072).

[†] Corresponding author. E-mail: 1769348608@qq.com