

# A comparison of ultrasonic temperature monitoring using machine learning and physics-based methods for high-cycle thermal fatigue monitoring

Laurence Clarkson , Yifeng Zhang  and Frederic Cegla

Structural Health Monitoring

2024, Vol. 23(3) 1560–1577

© The Author(s) 2023



Article reuse guidelines:

[sagepub.com/journals-permissions](https://sagepub.com/journals-permissions)

DOI: 10.1177/14759217231190041

[journals.sagepub.com/home/shm](https://journals.sagepub.com/home/shm)

## Abstract

Failure of pipe network components in so-called mixing zones due to high-cycle thermal fatigue (HCTF) can occur within nuclear power plants where fluids of different thermal and hydraulic properties interact. Given that the consequences of such failures are potentially deadly, a method to monitor HCTF non-invasively in real-time is expected to be of great use. This method may be realised by a technique to determine the inaccessible temperature distribution of a component since thermal gradients drive HCTF. Previous work showed that a physics-based method called the inverse thermal modelling (ITM) method can obtain the temperature distribution from external temperature and ultrasonic time of flight (TOF) measurements. This study investigated whether the long-short-term memory (LSTM) machine learning architecture could be a faster alternative to the ITM method for data inversion. On experimental data, a 25-member ensemble of LSTM networks achieved an ensemble median root mean square error (RMSE) of 1.04°C and an ensemble median mean error of 0.194°C (both relative to a resistance temperature device measurement). These values are similar to the ITM method which achieved a RMSE of 1.04°C and a mean error of 0.196°C. The single LSTM network and the ITM method achieved a computation-to-real-world time ratio of 0.008% and 14%, respectively demonstrating that both methods can invert data in real-time. Simulation studies revealed that LSTM performance is sensitive to small differences between the training and real-world parameters leading to unacceptable errors. However, these errors can be detected via an ensemble of independent networks and, corrected by simply adding a correction factor to the TOF prior to being input into the networks. The results show that LSTM has the potential to be an alternative to the ITM method; however, the authors favour ITM for temperature distribution monitoring given its interpretability.

## Keywords

ITM, LSTM, machine learning, thermal fatigue, ultrasonic thermometry

## Introduction

### Motivations

Pipe networks within nuclear power plants (NPPs) are susceptible to high-cycle thermal fatigue (HCTF) in so-called mixing zones where fluids of different thermal and hydraulic properties interact.<sup>1,2</sup> This susceptibility is, in part, due to the high thermal expansion coefficient and low thermal conductivity of austenitic stainless steels (SSs)<sup>3</sup> that are used throughout different types of NPPs.<sup>4,5</sup>

In May 1998, a crack of a pipe elbow within the reactor heat removal system (RHRS) caused a leak at the French Civaux 1 pressurised water reactor after just 1500 h of operation.<sup>6</sup> The leak caused the release

of radioactive steam at a rate of 30 m<sup>3</sup> h<sup>-1</sup> into the reactor building.<sup>7</sup> The location of the 180 mm through-wall crack on the pipe elbow is shown in Figure 1. Following this incident, Civaux 1 and three other reactors of the same design were defueled, and a failure analysis was performed.<sup>7</sup> The analysis identified that the failure was caused by thermal-fatigue-initiated

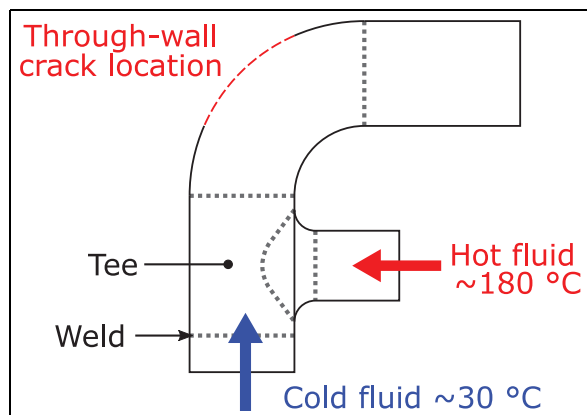
---

Non-Destructive Evaluation Group, Department of Mechanical Engineering, Imperial College London, London, UK

### Corresponding author:

Frederic Cegla, Non-Destructive Evaluation Group, Department of Mechanical Engineering, Imperial College London, Exhibition Road, London SW7 2AZ, UK.

Email: [f.cegla@imperial.ac.uk](mailto:f.cegla@imperial.ac.uk)



**Figure 1.** Schematic of the Civaux 1 RHRS pipe elbow illustrating the location of the fatigue crack location.  
Source: Geometry replicated from Cipière et al.<sup>9</sup>  
RHRS: reactor heat removal system.

cracks.<sup>8</sup> This HCTF phenomenon had not been considered during the design of the NPPs since it was not captured in the design standards of the time.<sup>6</sup> Hence, redesign, requalification and replacement of the affected components of the RHRS in all four reactors had to be performed. Subsequent ultrasonic inspection in 1999 of all NPPs in France revealed that thermal fatigue cracking was not unique to the Civaux 1 reactor design (due to the limitations of the design standards<sup>6</sup>). Further research identified that mixing zone HCTF is primarily caused by repeated exposure to temperature fluctuations affected by differences  $>50^{\circ}\text{C}$  between hot and cold fluids.<sup>6</sup> It was also found that the fatigue is most severe for temperature fluctuations with frequencies in the range from 0.1 to 1 Hz.<sup>9,10</sup>

Given the susceptibility of austenitic steels to thermal fatigue,<sup>3</sup> and that safety is paramount in the nuclear industry, it is desirable to investigate techniques that can monitor the progression of HCTF and as a result, relax the requirements on the inspection interval and remove operators from the hazardous environment.

### Article structure

The remainder of the article is structured as follows: the section ‘Limitations of current HCTF monitoring methods’ details the short-comings of current HCTF monitoring methods, and the physical reasons for this. A brief explanation of a physics-based ultrasonic temperature inversion method is presented in the section ‘Inverse thermal modelling method’. The section ‘Long short-term memory’ describes the machine learning network architecture considered in this work, the process for generating (simulated) training and testing

data and the steps for training networks. The section ‘Simulation studies’ defines an initial test case and two additional test cases concerning data that have never been seen before by the trained networks. The section ‘Experimental studies’ introduces experimental data used to evaluate the machine learning networks on real-world data. The results of the machine learning networks are shown and discussed first for the simulated test data followed by the experimental test data, with results for the physics-based inversion method included for comparison. Finally, a summary of the key findings are provided in the conclusions.

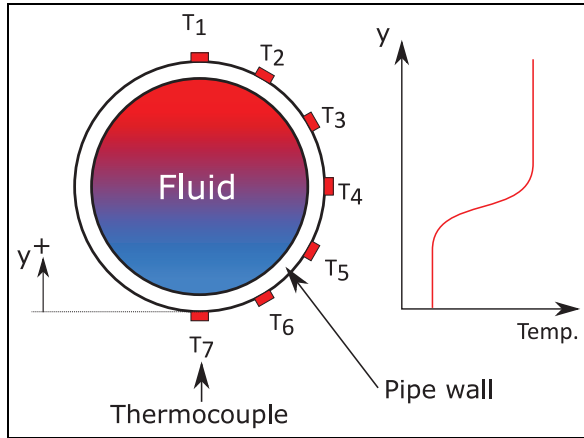
### Limitations of current HCTF monitoring methods

A component exposed to temperature fluctuations will develop thermal gradients that will generate thermal stresses. If sufficiently large, these stresses will impart damage leading to crack initiation/propagation and eventually cause the component failure. This failure mechanism is known as thermal fatigue.<sup>11</sup> For a pipe carrying a thermally varying fluid, the inaccessible interior surface will experience the largest (compressive or tensile) stresses.

Since thermal fatigue progression is driven by thermal gradients, knowing how the through-thickness temperature profile evolves over time is vital for monitoring thermal fatigue.<sup>10</sup> However, this is a difficult task because traditional temperature measurement equipment (e.g. thermocouples) can only measure surface temperatures. Several techniques to overcome this issue have been developed; the following two sections will introduce and evaluate these techniques.

**Embedded thermocouples.** The obvious method to obtain the temperature profile in a component is by embedding sensors into a component. This method is demonstrated in two studies<sup>12,13</sup> by embedding thermocouples throughout the thickness of a component via drilled holes. However, these holes will create stress-raising features that will accelerate thermal fatigue progression.<sup>14</sup> Furthermore, RTDs resistance temperature detector have been shown to lag true temperature in the order of seconds due to thermal conduction into the device<sup>15</sup> causing measurement errors.

**FAMOSi.** The integrated FATigue MOonitoring System (FAMOSi) was developed by Siemens in the 1980s and later updated by Areva (a French multinational group with a focus on nuclear power) for thermal fatigue monitoring following the discovery of fatigue cracks in NPPs.<sup>16</sup> The resulting non-invasive system, called Integrated FAMOS (FAMOSi), comprises seven temperature sensors mounted around one half of a pipe’s



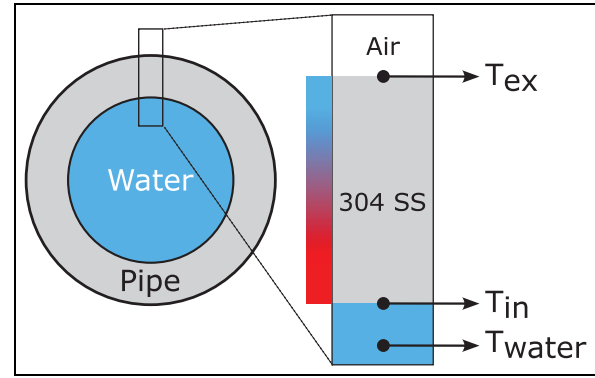
**Figure 2.** Schematic diagram of FAMOSi.

Source: Adapted from Bergholz et al.<sup>16</sup> and Rudolph et al.<sup>17</sup>

circumference as shown in Figure 2. The system is based on a comparison of the outer surface temperature-time history to a pre-compiled reference database of ‘responses’ which are computed via a finite element model (FEM). It is likely to be a significant task to validate the FEM. Furthermore, the system is incapable of detecting thermal fluctuations  $>1$  Hz.<sup>17</sup> Although the reason for this limitation is not explicitly stated in the literature, it is assumed to be due to the low thermal conductivity of austenitic steels preventing thermal energy from diffusing to the component’s outer surface rather than a limitation in computational or data acquisition capabilities. This effect is demonstrated in the next section.

**Thermal conduction: a low-pass filter.** This section presents simulations to demonstrate that materials with low thermal conductivity effectively act as low-pass filters of temperature. This low-pass effect implies that the previously introduced HCTF monitoring methods, based on externally mounted temperature sensors, will be unsuitable for resolving sub-surface temperatures that fluctuate rapidly, especially for thick components.

An explicit 1D, finite difference heat transfer model with convective boundary conditions was developed based on the model used by Zhang et al.<sup>18</sup> The model simulated the temperature profile evolution of a section of 304 SS pipe carrying water at constant pressure and the outer surface exposed to air as shown in Figure 3. The fluctuation frequency of the water temperature was set to be a linear up-chirp: the instantaneous frequency increases linearly with time. The properties and parameters of the simulation are summarised in Table 1 where  $T$ ,  $h$ ,  $k$ ,  $\alpha$ ,  $L$ ,  $\rho$  and  $\Delta x$  are temperature, convective heat transfer coefficient, thermal conductivity, thermal diffusivity thickness, density and nodal spacing,



**Figure 3.** Schematic of the simulation setup. The colour bar denotes an arbitrary temperature gradient due to the difference in temperature between the air and water.

**Table 1.** Parameters and material properties used in the simulation (304 SS).

Parameter	Value	Unit
Simulation time	500	min
$T_{\text{air}}$	50	$^{\circ}\text{C}$
Mean $T_{\text{water}}$	50	$^{\circ}\text{C}$
Chirp peak-to-peak amplitude	40	$^{\circ}\text{C}$
Chirp frequency range	0.01–5	Hz
$h_{\text{air}}$	45.2	$\text{W}/\text{m}^2/\text{K}$
$h_{\text{water}}$	15,000	$\text{W}/\text{m}^2/\text{K}$
$k_{304}$	18.5	$\text{W m}^{-1} \text{K}^{-1}$
$\alpha_{304}$	4.64	$\text{mm}^2 \text{s}^{-1}$
$L$	10	mm
$\rho_{304}$	7850	$\text{kg m}^{-3}$
$\Delta x$	0.625	mm

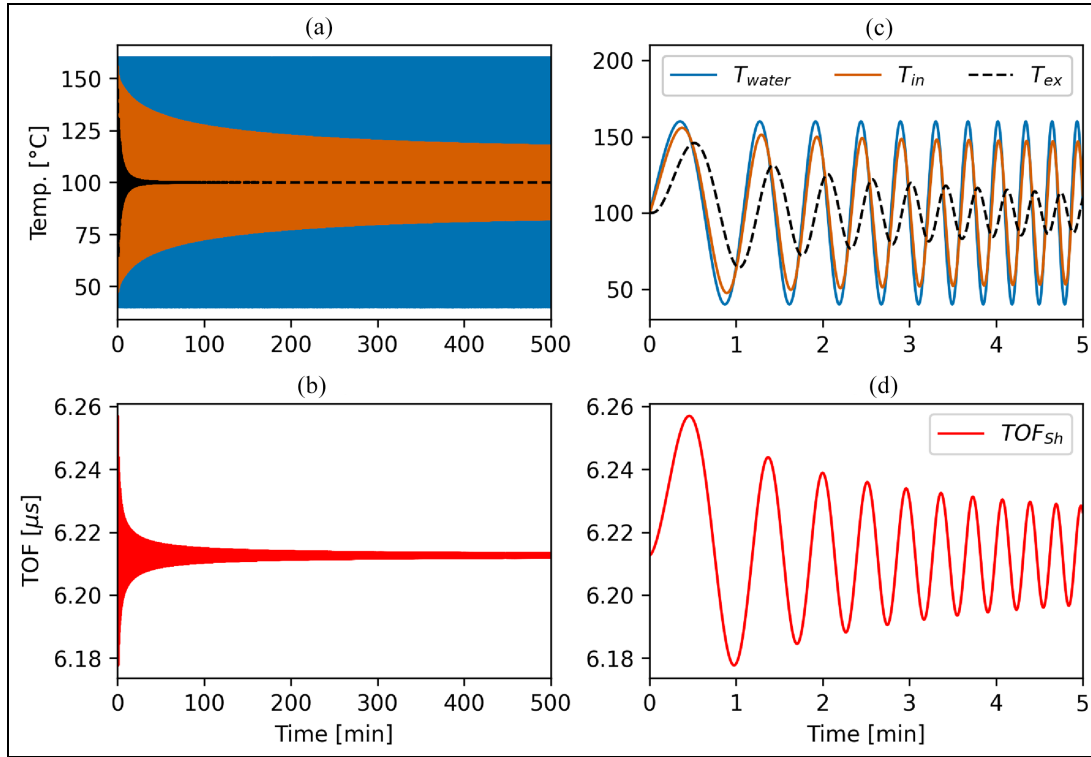
SS: stainless steel.

respectively. The material properties of 304 SS were taken from the CES Edupack materials database (Granta Design Limited).<sup>19</sup> The values for thickness and  $h_{\text{water}}$  match the conditions at Civaux 1 that lead to the tee-joint failure.<sup>20</sup> The minimum time step for stable computation is given by  $\Delta t_{\text{min}}$  in Equation (1).

$$\Delta t_{\text{min}} = \frac{\Delta x^2}{2\alpha \left(1 + \frac{h_{\text{in}} \Delta x}{k}\right)} \approx 0.027 \quad (1)$$

The 500 min of simulated time yields a rate of change of chirp frequency of  $\approx 0.17 \text{ mHzs}^{-1}$  which is quasi-stationary compared with the sampling rate  $\approx 0.027 \text{ s}$ . Following the computation of the temperature profile, shear wave time of flights (TOFs) were calculated using the trapezoidal approximation<sup>18</sup> given in Equation (2).

$$\text{TOF} = \int_0^L \frac{2}{v(T(x))} dx \approx \Delta x \left[ \left( \frac{1}{v_1} + \frac{1}{v_N} \right) + 2 \sum_{i=2}^{N-1} \frac{1}{v_i} \right] \quad (2)$$



**Figure 4.** Results of the finite difference simulation for a 10 mm block of 304 SS exposed to temperature-varying water that fluctuates according to a sine wave with linearly increasing frequency. Temperatures of the water, and at the inner and outer surface of the component are shown in the top row. The bottom row shows the shear TOF. Panels (c) and (d) are zoomed to show the first 5 mins of panels (a) and (b), respectively. The left column shows full 500 min of the finite difference simulation whilst the plots in the right column show only 5 min. SS: stainless steel; TOF: time of flight.

In Equation (2)  $\Delta x$ ,  $v_i$  and  $N$  denote the nodal spacing, wave velocity at the  $i^{\text{th}}$  node, and the total number of nodes respectively. A quadratic fit was used to describe the velocity–temperature ( $v(T)$ ) relationship. Previous work by Gajacsi<sup>21</sup> provided an experimentally determined fit given in Equation (3).

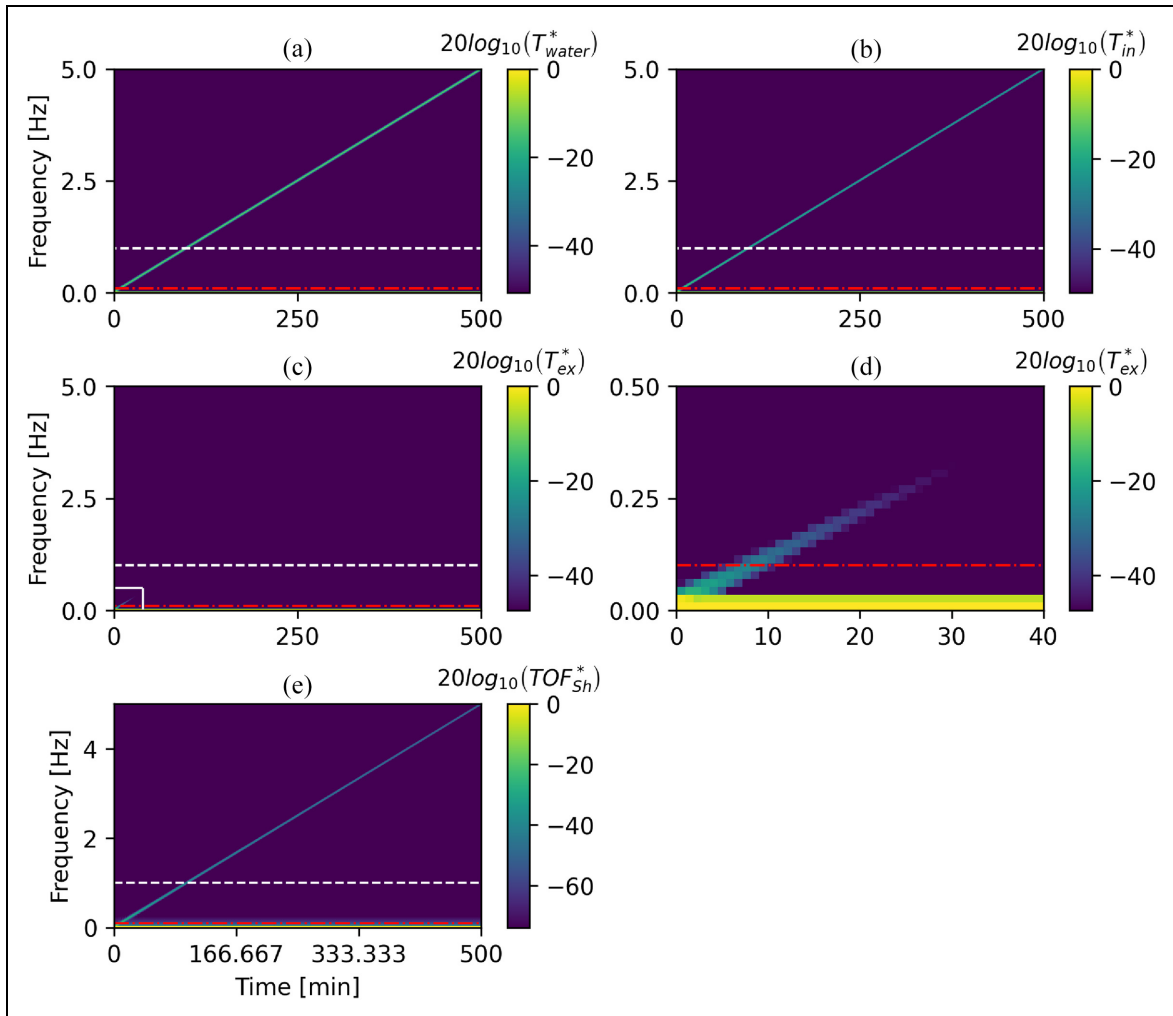
$$v_{\text{Sh},304}(T) = -10^{-5} T^2 - 0.775T + 3188.73 \quad (3)$$

Figure 4 shows the water temperature ( $T_{\text{water}}$ ), the internal ( $T_{\text{in}}$ ) and external ( $T_{\text{ex}}$ ) surface temperatures of the steel and shear ( $\text{TOF}_{\text{Sh}}$ ) TOFs computed in the simulation. Panel (a) demonstrates the low-pass effect: the amplitude of the fluctuations of  $T_{\text{ex}}$  rapidly decreases as the frequency of  $T_{\text{water}}$  increases and quickly drops below the measurement repeatability (standard deviation (SD)) of a class A RTD<sup>22</sup> at 100°C. The reduction in amplitude of  $T_{\text{in}}$  is less relative to  $T_{\text{ex}}$  and remains above the class A RTD measurement repeatability. The amplitude of the fluctuations of the shear wave TOF also decreases over time. However, the peak-to-peak amplitudes both remain above the measurement repeatability (SD) of the piezoelectric transducers used by Zhang et al.<sup>18</sup>

A spectrogram of the full 500 min of simulation for each variable is shown Figure 5. The horizontal lines at 0.1 Hz (red dash-dot) and 1 Hz (white dashed) are superimposed on Figure 5 to show the range of critical fluctuation frequencies for HCTF of the tee-joint at Civaux.<sup>9,10</sup> The spectrograms were computed to identify the maximum resolvable fluctuation frequency for each variable. To create a spectrogram, a given variable was segmented into periods of 2048 time steps. The Fourier transform was computed for each period and stacked along the  $x$ -axis, that is, a visual representation of the frequency content of a signal for each consecutive 2048 time step period. Each variable was normalised by its absolute range before the spectrogram was computed, that is, Equation (4).

$$X^* = \frac{X}{|X_{\text{max}} - X_{\text{min}}|} \quad (4)$$

The minimum of the colour bar ( $\hat{X}$ ) for each panel in Figure 5 was set according to Equation (5) where  $X$  denotes temperature or TOF.  $\sigma_X$  denotes the experimental SD.



**Figure 5.** Spectrograms for the temperatures (a)–(d) and shear TOF (e) computed by the finite difference simulation (Figure 4). Each variable was normalised according to Equation (4) before computing the spectrogram. The horizontal red dash-dot and white dashed lines denote 0.1 and 1, respectively. The solid white lines in panel (c) bound the zoom extents for panel (d). TOF: time of flight.

$$\hat{X} = 20 \log_{10} \left( \frac{\sigma_X}{|X_{\max} - X_{\min}|} \right) \quad (5)$$

For temperature,  $\sigma_X$  was set to be 0.35°C (class A RTD<sup>22</sup> at 100°C). For TOF,  $\sigma_X$  was set to 16 ps (SDs of the piezoelectric transducers used by Zhang et al.<sup>18</sup>).

Figure 5(d) shows that a temperature sensor mounted to the external surface of a component cannot differentiate temperature fluctuation frequencies greater than approximately 0.29 Hz. Furthermore, the practical frequency limit is likely lower than 0.29 Hz due to the lag time of typical temperature sensors<sup>15,18,23</sup> which is not considered in these simulations. In contrast, the shear TOFs (Figure 5(e)) remain sensitive up to 5 Hz. TOF should remain sensitive well beyond 5 Hz although the theoretical upper limit of the sensitivity

was not investigated in this simulation. The upper limit is expected to be governed by the sampling rate of the acquisition hardware.

### Inverse thermal modelling method

A feasibility study by Zhang et al.<sup>18</sup> utilised the thermal sensitivity of (shear) ultrasonic TOF (as shown in the previous section), demonstrating internal surface temperature estimation within 2°C. The method couples TOF and outer (accessible) surface temperature measurements with a physics-based inversion model to obtain temperature estimations. Of the two inversion methods presented by Zhang et al., only the inverse thermal modelling (ITM) method based on earlier work by Ihara et al.<sup>24</sup> will be considered in this article.

The ITM method is based on iterative optimisation of an explicit finite difference formulation of the inverse heat conduction problem enabling it to obtain the full temperature profile. The explicit formulation requires that the time step of the data must be sufficiently small to ensure a stable solution. Hence, interpolation of the data is usually necessary to ensure stability which increases computational expense.

### Long short-term memory

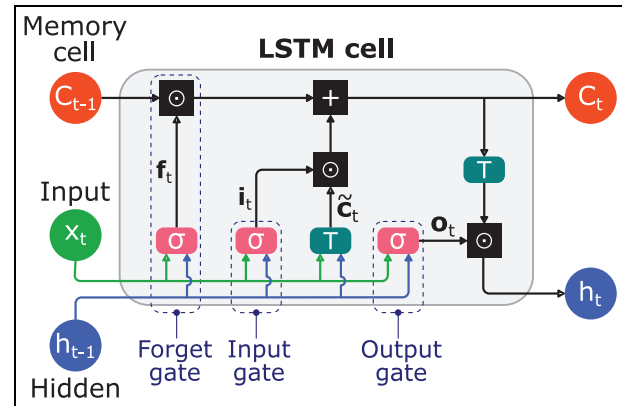
Machine learning is a technique that is being increasingly implemented in non-destructive evaluation scenarios, including:

- Ultrasonic flaw classification<sup>25</sup>
- Deconvolution of ultrasonic signals<sup>26</sup>
- Artefact identification and suppression in ultrasonic images<sup>27</sup>
- Defect detection in guided wave signals<sup>28</sup>
- Noise quantification in ultrasonic images<sup>29</sup>
- Ultrasonic crack characterisation<sup>30</sup>

Machine learning has several beneficial characteristics including:

- A physics-based model of an underlying system is not required<sup>31</sup>
- The ability to implicitly create complex non-linear relationships

One drawback is that it is difficult to explore the explainability of the mathematical operations inside the model leading to machine learning networks to often be described as a ‘black box’ method.<sup>32</sup> Very broadly, deep learning – a subset of machine learning – can use all the available information embedded within the data set by simply using the raw data as the input. In contrast, shallow learning requires hand selection of input features but requires less training data than deep learning.<sup>33</sup> Within deep learning, recurrent neural networks (RNNs) are well suited for problems involving time-series data; however, they can be susceptible to gradient vanishing and gradient exploding problems.<sup>34</sup> To overcome this issue, a novel and efficient gradient-based method, called long short-term memory (LSTM) was developed by Hochreiter and Schmidhuber.<sup>35</sup> This article explores whether machine learning networks can replace the (single shear wave) ITM inversion method for real-time temperature gradient monitoring using networks trained using simulation data only. Given the available scope of this article, it would not be possible to explore all potential machine learning architectures. The LSTM architecture was selected for



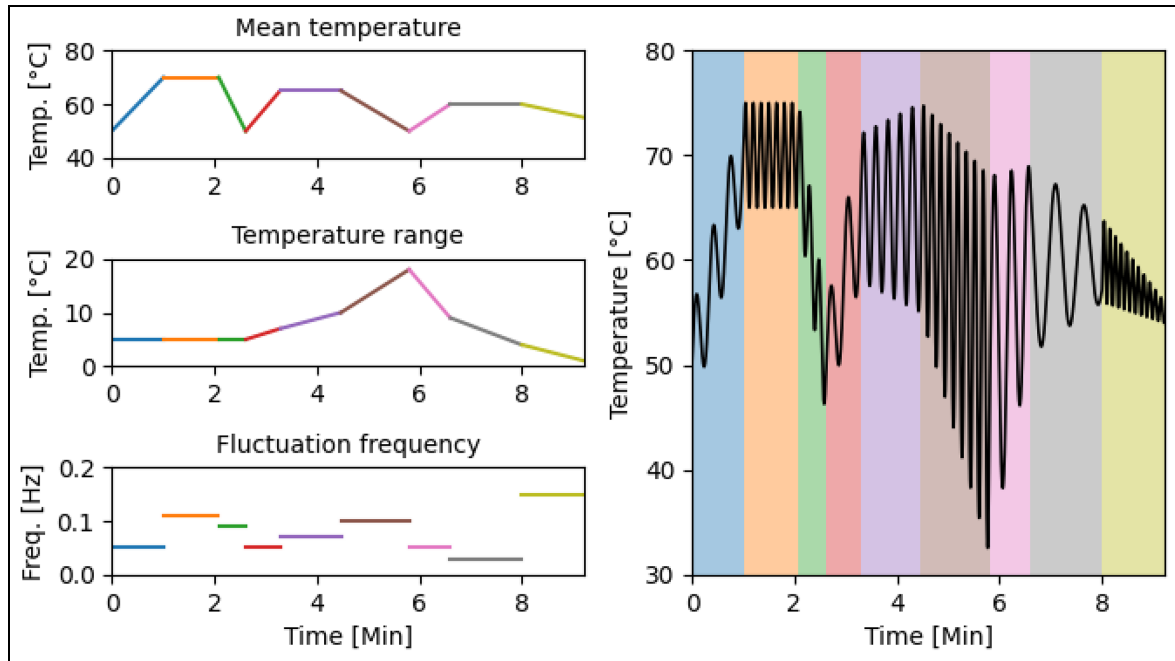
**Figure 6.** Graphical illustration of an LSTM cell. The weights are omitted for clarity.  $\sigma$ ,  $T$ ,  $\odot$  and  $+$  denote sigmoid and tanh layers, element-wise multiplication and addition, respectively. LSTM: Long-short-term memory.

investigation as it seemed a prominent candidate and showed promising results in initial evaluations. For a detailed discussion of other types of architectures that have been proposed for a range of non-destructive evaluations applications, beyond time-series data, the review paper by Cantero-Chinchilla et al.<sup>36</sup> is suggested to the interested reader.

Figure 6 shows a schematic of an LSTM cell that contains four layers. The four layers comprise three logistic sigmoid and one hyperbolic tangent (tanh) functions that interact to produce the output and the state of the cell which are then passed onto the next hidden layer. An LSTM cell has three inputs:  $h_{t-1}$ ,  $c_{t-1}$  and  $x_t$  and two outputs:  $h_t$  and  $c_t$ . Subscripts  $t$  and  $t-1$  denote the current and previous time steps, respectively.  $h$  is the hidden state,  $c$  is the cell state (or memory) and  $x$  is the input. The output of the first sigmoid layer (forget gate  $f_t$ ) defines the amount of information of the previous cell to be maintained via an element-wise multiplication with  $c_{t-1}$ . The tanh layer yields a vector  $\tilde{c}_t$  of the new candidate values. The element-wise multiplication of the second sigmoid layer (input gate,  $i_t$ ) and  $\tilde{c}_t$  determines the amount of information to be added to the cell state. This result is then added to the output of the forget gate multiplied with  $c_{t-1}$  to produce  $c_t$ . The final sigmoid layer (output gate,  $o_t$ ) is multiplied in an element-wise fashion with a tanh layer to produce the output  $h_t$  of the cell. Further details on the LSTM architecture can be found in Hochreiter et al.<sup>35</sup> and DiPietro and Hager.<sup>37</sup>

### Train/test data generation and network training

The previously introduced finite difference heat transfer code was used to generate data sets for training and



**Figure 7.** Representative example showing a region for each of the possible nine mean temperature and temperature range combinations used when generating  $T_{\text{water}}$  using the sine method for training data. The fluctuation frequencies and time for each region were chosen purely for clear visualisation. A frequency range of 0.1–1 Hz was used to generate simulated training or test data.

testing LSTM networks. However, the method for defining the water temperature fluctuations differed. The first method (later referred to as the sine method) defined the fluctuations to be sinusoidal (rather than a linear chirp) and were defined by three parameters:

1. Mean temperature (increasing, decreasing, or constant)
2. Temperature range (increasing, decreasing, or constant)
3. Fluctuation frequency ( $0.1 \leq f \leq 1$  Hz, critical frequency range for HCTF at Civaux<sup>9,10</sup>)

Each of the parameters was selected for each ‘region’ in the data set using uniform random distributions; discrete forms of the distribution are used for the mean temperature and temperature range. Figure 7 shows nine representative regions demonstrating each of the possible mean temperature and temperature range combinations for the sine method. The training data set used to train all LSTM networks was generated using the parameters given in Table 2.

The second method (later referred to as the square method) used to create a test set that mimics the experimental data introduced in a later section, defined the fluctuations as square waves. In both methods, the upper limit of  $T_{\text{water}}$  was set as the saturation temperature at the chosen (constant) pressure, calculated using

**Table 2.** Parameters and material properties used to generate the training data.

Parameter	Value	Unit
Generation method	Sine	–
Regions	100	–
Fluctuation frequency range	0.1–1	Hz
Region time range	5–20	min
Time steps	$1.01 \times 10^7$	–
Train-validation split	85–15	%
Thickness	30	mm
Nodes	61	–
Nodal spacing	0.5	mm
Time step	7.01	ms
Material	EN32B steel	–
Thermal diffusivity	17.7	$\text{mm}^2 \text{s}^{-1}$
Thermal conductivity	70.2	$\text{W m}^{-1} \text{K}$
Density	7890	$\text{kg m}^{-3}$
Water heat transfer coefficient	1100	$\text{W/m}^2/\text{K}$
Air heat transfer coefficient	45.2	$\text{W/m}^2/\text{K}$
Pressure	0.10	MPa
Saturation temperature	99.6	$^{\circ}\text{C}$

the properties of water and steam based on the formulation coordinated by the International Association for the Properties of Water and Steam,<sup>38</sup> to ensure physical sense. Finally, the generated data were down-sampled to a 0.5 Hz sampling rate which is the Nyquist frequency<sup>39</sup> of the maximum critical frequency of HCTF at Civaux.<sup>9,10</sup>

The networks were created in Python 3<sup>40</sup> using the Keras deep learning API.<sup>41</sup> Each network comprised a single LSTM layer with 180 neurons followed by a one-unit dense layer. An 85:15 train-validation split was used on training data sets, and separate unseen data sets were used for testing. All training sets were generated using the sine method. The input data were assembled such that for a given time step, the data passed to the network contained two values: the value at the current, and one previous time steps. The Adam optimizer<sup>42</sup> was used in conjunction with Equation (6) which describes the exponentially decaying learning rate,  $L_r$ , where  $n$  refers to the current training epoch. The root mean squared error (RMSE) and mean error were used as quantitative metrics to assess performance on the test sets.

$$L_r[n] = 0.01 e^{-0.05n} \quad (6)$$

The mean square error was used as the loss function. Early stopping based on 25 epochs without reduction of the validation set loss was used to prevent the networks from over-fitting to the training data. The networks were trained with a 12th Gen Intel core i7 processor CPU of a desktop PC. This PC was used throughout the work presented in this article.

A manual ‘trial and error’ approach was used over formal optimisation strategies of the LSTM networks for two reasons. Firstly, because this work set out to determine whether machine learning might be a less computationally expensive alternative to physics-based methods rather than finding the best (highest accuracy) machine learning method. Secondly, the first trial implementation yielded good results (single 50-neurons LSTM layer followed by a one-unit dense layer). Furthermore, better performance (defined as lower RMSE/mean error) should only be considered on experimental data. However, this poses a limitation due to the lag of RTDs<sup>15,18,23</sup> – the LSTM predictions are compared to a reference measurement rather than a ground truth during experiments. Hence lower RMSE/mean error on experimental data by LSTM would mean that the inherent lag of the RTDs is being learnt, this is not beneficial and hence a sensitivity study of changes in performance metrics was not attempted.

## Simulation studies

It was expected that if a LSTM network could predict the temperature at a single distance from the water–metal interface, a more complex network would be able to predict the full temperature profile of a component. In this work, a single distance network was investigated to confirm whether the LSTM architecture was a suitable choice for data inversion. The single distance was

**Table 3.** Properties used to generate the simulated training and test data (EN32B mild steel).

Parameter	Value	Unit
$T_{\text{air}}$	20	°C
$h_{\text{air}}$	45.2	$\text{m}^2 \text{s}^{-1}$
$h_{\text{water}}$	1100	$\text{m}^2 \text{s}^{-1}$
$k_{\text{EN32B}}$	70.2	$\text{W m}^{-1} \text{K}^{-1}$
$\alpha_{\text{EN32B}}$	17.7	$\text{mm}^2 \text{s}^{-1}$
$L$	30	mm
$\rho_{\text{EN32B}}$	7890	$\text{kg m}^{-3}$

chosen to be 5 mm from the water–metal interface to match the reference RTD distance in the experimental data (introduced later). A training set that simulated an EN32B mild steel block exposed to water on one side and the other exposed to constant temperature air was created using the sine method comprising 100 regions. A single shear wave was used since the changes in TOF were due to temperature changes only (the component thickness remained constant). The shear wave velocity–temperature relationship of the sample is given in Equation (7).

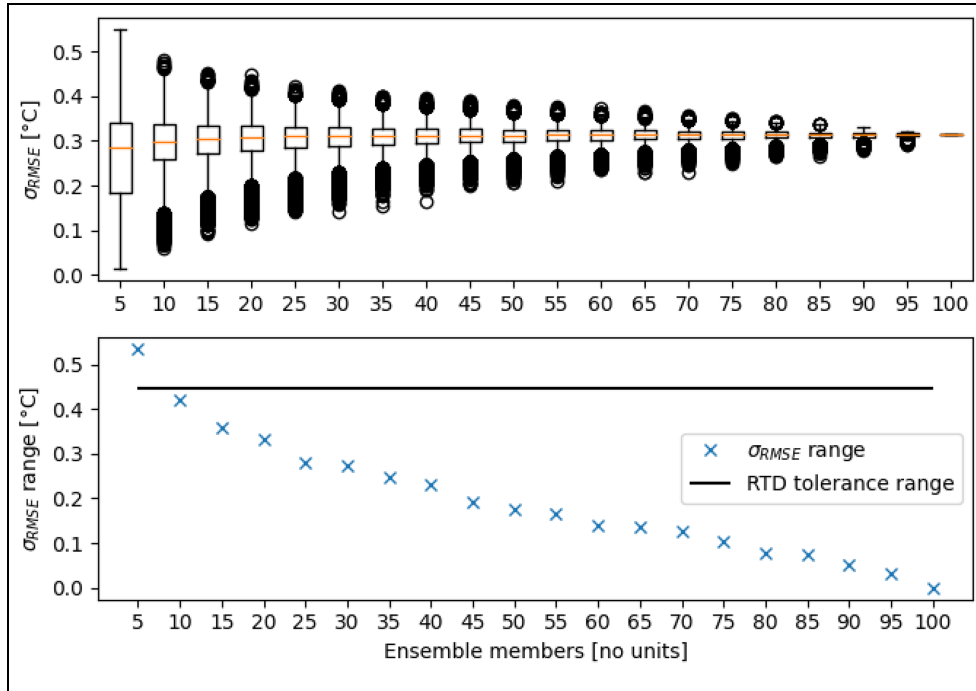
$$v_{\text{Sh, EN32B}}(T) = AT + B = -0.48894T + 3237.61 \quad (7)$$

LSTM networks were then trained with this data set. Following training, the performance of the networks was evaluated using an initial simulated test set with the water temperature defined with the square method to switch between hot and cold, with the magnitudes and periods of exposure matching the experimental data set. The simulation parameters are shown in Table 3 where the symbols have the same definitions as in Table 1.

## Deep ensemble

To observe the influence and minimise the impact of the random initialisation of the network weights each time a new network was trained, 100 networks (with identical architectures) were trained using the same training set. These networks were used to create a deep ensemble. Deep ensembles of machine learning networks have been shown to increase prediction accuracy and provide a measure of uncertainty.<sup>33</sup> To determine a sufficient number of networks (members) in the ensemble, increasing numbers of members were included in the ensemble and the SD of the RMSE was computed. This process was repeated with random (unique) shuffles of the order in which the networks were included in the ensemble. However, only 20,000 shuffles were considered since for 100 networks it would be unrealistic to consider all  $1.26 \times 10^{30}$  possible combinations, as given in Equation (8). In Equation (8),  $n$  and  $r$  are the total





**Figure 8.** The upper panel shows the standard deviation of RMSE of the ensembles for increasing numbers of members in the LSTM ensemble for predictions on the baseline simulation test set. The standard deviations of the RMSE are shown for 20,000 unique random shuffles of the order in which networks are considered for inclusion in the ensemble. Values that were more than  $1.5 \times$  the interquartile range are shown by the circles. The lower panel shows the range of standard deviations of the RMSE (which are shown in the upper panel). In the lower panel, the solid black line denotes the tolerance range of a class A RTD<sup>22</sup> as defined by Equation (9). LSTM: long-short-term memory; RMSE: root mean square error.

number of possible members and the current number of members considered, respectively.

$$\sum_{r=1}^n \frac{n!}{r!(n-r)!} = 2^n - 1 = 2^{100} - 1 \approx 1.26 \times 10^{30} \quad (8)$$

The range of the RMSE SD was compared to the (peak-to-peak) range tolerance, at the mean temperature of the simulated data set ( $\approx 36.6^{\circ}C$ , which closely matches that of the experimental data set), of a class A RTD<sup>22</sup>:  $Tol \approx 0.45^{\circ}C$  (Equation (9)).

$$Tol = \pm 0.15 + 0.002 \times 36.6 \approx \pm 0.223 \rightarrow 0.446^{\circ}C \quad (9)$$

This comparison is shown in Figure 8. As expected, as the number of ensemble members reaches the maximum number, the range of the SD of the RMSE reduces to zero since a change in the inclusion order of networks becomes less significant. The ensemble SD range falls below that of the RTD after 10 members. While 10 members could be argued to be sufficient the authors decided to be conservative by including 25 members so that variations due to simulations would be smaller than those that are expected in experimental measurements. For the avoidance of any doubt, 25 of

the 100 networks were randomly selected and the same 25 networks were subsequently used throughout this article for any and all simulation or experimental studies.

### Robustness against out-of-distribution data

Out-of-distribution data (OODD) are data that a trained network has never been exposed to because the training set does not capture it. The response of the LSTM ensemble to OODD was explored to assess two areas. Firstly, assess the magnitude of the impact of OODD on the ensemble, that is, how badly wrong do the predictions get on previously unseen data. Secondly, whether the deep ensemble could detect when the model is working outside of the predefined domain of operation using the SD of the ensemble predictions. The second area encompasses quantification of the epistemic uncertainty. Epistemic uncertainty arises from a lack of knowledge about data generation method, resulting in uncertain network parameters. The uncertainty can be reduced by increasing the amount of relevant training data, provided that the training data aligns closely with the test data. However, it is important to note that in this work the epistemic

**Table 4.** Values of the velocity–temperature relationship coefficients used to generate the velocity OOD test sets. The percentage changes are relative to the values used when generating the base test set.

OODD case	Velocity ( $V = AT + B$ )		Deviation (%)
	A ( $\text{m s}^{-1} \text{ } ^\circ\text{C}^{-1}$ )	B ( $\text{m s}^{-1}$ )	
1	−0.48845	3234.37	−0.10
2	−0.48869	3235.99	−0.05
3	−0.48889	3237.29	−0.01
<b>Base</b>	<b>−0.48894</b>	<b>3237.61</b>	<b>0.00</b>
4	−0.48898	3237.93	0.01
5	−0.48918	3239.23	0.05
6	−0.48942	3240.85	0.10

OODD: out-of-distribution data. The base dataset is highlighted in bold.

**Table 5.** Values of the thicknesses used to generate the thickness OOD test sets. The percentage changes are relative to the values used to generate the base test set using absolute magnitudes.

OODD case	Thickness L (mm)	Deviation (%)
1	29.90	−0.33
2	29.95	−0.17
<b>Base</b>	<b>30.00</b>	<b>0.00</b>
3	30.05	0.17
4	30.10	0.33

OODD: out-of-distribution data. The base dataset is highlighted in bold.

uncertainty cannot be fully eliminated because simulation data has been used to approximate real-world data leading to an inherent difference between training and test data.<sup>33</sup>

Two independent OOD scenarios that were deemed most likely to occur in the real world were explored. The OOD scenarios were incorrect component thickness, and  $v(T)$  coefficients, referred to as thickness OOD and velocity OOD, respectively. All parameters used to generate the base test set in both OOD scenarios exactly matched those used to generate the training data apart from the random seed. The OOD test sets were derived from the base test set using the variation in parameters given in Tables 4 and 5 – all other parameters remained the same across the OOD sets (including the random seed). The OOD test sets including the base set were generated using the sine method.

## Experimental studies

The ‘temperature fluctuations only’ experiment data were provided by Zhang and Cegla<sup>23</sup> for use as a test

set to evaluate the real-world performance of the 25 previously trained LSTM networks. The ITM MATLAB<sup>43</sup> code was also provided and then rewritten in Python. The experiment comprised a sample of EN32B mild steel ( $L \approx 30\text{ mm}$ ) exposed to alternating hot and cold water by a purpose-built setup. A reference measurement was provided by an RTD embedded in the sample 5 mm from the water–metal interface using thermally conductive epoxy. Data were recorded at a sampling rate of 0.25 Hz, or one point every 4 s. The interested reader is directed to the original paper for in-depth details of the materials and methods.<sup>23</sup> The shear wave velocity–temperature relationship of the sample is given in Equation (7).

## Prediction time

To investigate the computational time of the LSTM ensemble, single LSTM network and ITM method, their respective Python codes were run 20 times using the experimental test set. Their respective computation times were measured (using the `perf_counter()` function<sup>44</sup>) and averaged. The snippets of code that were timed only included instructions explicitly related to making predictions.

## Results and discussion

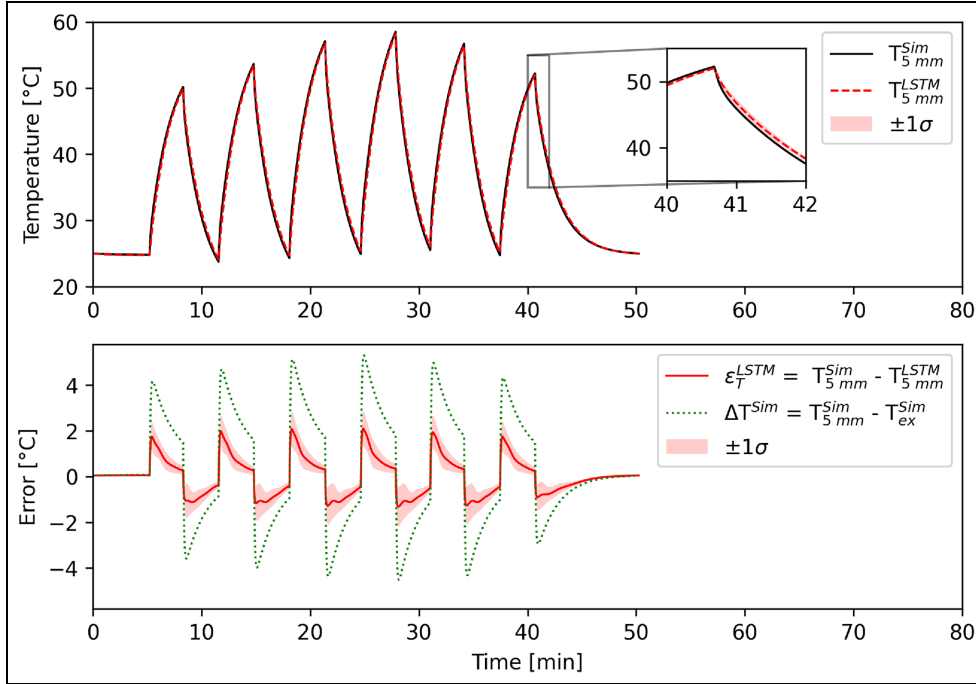
Throughout this section, Equation (10) was used to define the error between the true (simulated) or reference (RTD) measurement and the inversion method (LSTM or ITM) predictions whilst Equation (11) was used to define the error between the true (simulated) or reference (RTD) measurement and the external surface temperature. The subscript 5 mm refers to the distance from the water–metal interface that the simulated or experimental temperature measurements are taken from as the true and reference measurement, respectively.

$$\epsilon_T^{\text{Method}} = T_{5\text{mm}}^{\text{Source}} - T_{5\text{mm}}^{\text{Method}} \quad (10)$$

$$\Delta T^{\text{Source}} = T_{5\text{mm}}^{\text{Source}} - T_{\text{ex}}^{\text{Source}} \quad (11)$$

## Simulation studies

Figure 9 shows the results of the initial simulated test set. The top panel of Figure 9 shows the true (simulated) temperature 5 mm from the water–metal interface and the mean temperature predictions of the 25 LSTM networks at the same spatial location. One SD above and below the mean of the 25 networks is also superimposed to demonstrate the aforementioned impact of the random initialisation of each network.



**Figure 9.** Initial simulated test set results. Top panel: simulated temperature 5 mm from the water–metal interface ( $T_{5\text{mm}}^{\text{Sim}}$ ) and the ensemble mean predictions across the LSTM networks ( $T_{5\text{mm}}^{\text{LSTM}}$ ). Bottom panel: errors of the LSTM mean predictions ( $\varepsilon_T^{\text{LSTM}}$ ) and external simulated temperature ( $\Delta T^{\text{Sim}}$ ), both relative to the true simulated temperature. One standard deviation across the LSTM networks above and below the mean prediction is superimposed in both panels ( $\pm 1\sigma$ ). LSTM: long-short-term memory.

The bottom panel shows the error relative to the true temperature for the mean LSTM predictions as well as for the temperature at the steel block’s outer surface (the air–metal interface).  $\varepsilon_T^{\text{LSTM}}$  is approximately half that of  $\Delta T^{\text{Sim}}$  demonstrating that the LSTM machine learning approach can outperform the most basic non-invasive temperature sensing method. The largest values of SD occur when the water temperature switches between hot and cold which lead to the largest temperature gradients at the water–metal interface.

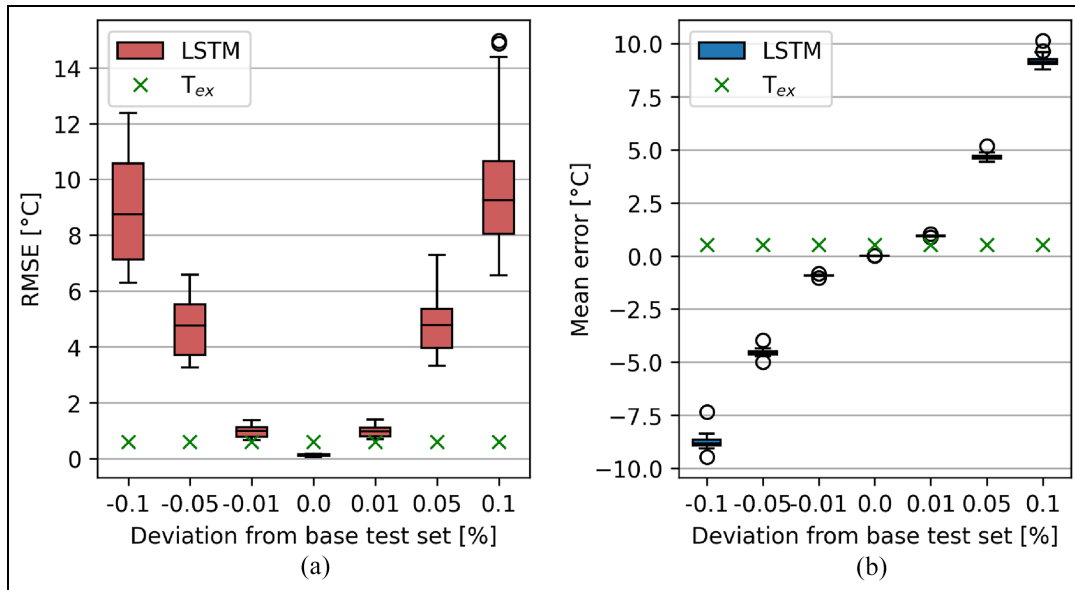
Figure 10 shows the RMSE and mean error for the velocity OOD test sets taken over all time steps for each network. The metrics based on the external temperature are also shown as a reference. As both  $v(T)$  coefficients diverge from the values used during training (and in the base test case), the absolute magnitude of both the RMSE and mean error increase. For the avoidance of doubt, OOD cases 4–6 of Table 4 are referred to as positive deviations of the  $v(T)$  coefficients. Essentially, the absolute magnitude of each coefficient in the test set is larger than the values used during training;  $A$  becomes *more* negative and  $B$  becomes *less* negative. In a similar fashion, OOD cases 1–3 of Table 4 are referred to as negative deviations. It was expected that for the positive deviations of both  $v(T)$  coefficients, the LSTM networks would

underpredict the true temperature and therefore the error (according to Equation (10)) would be positive.

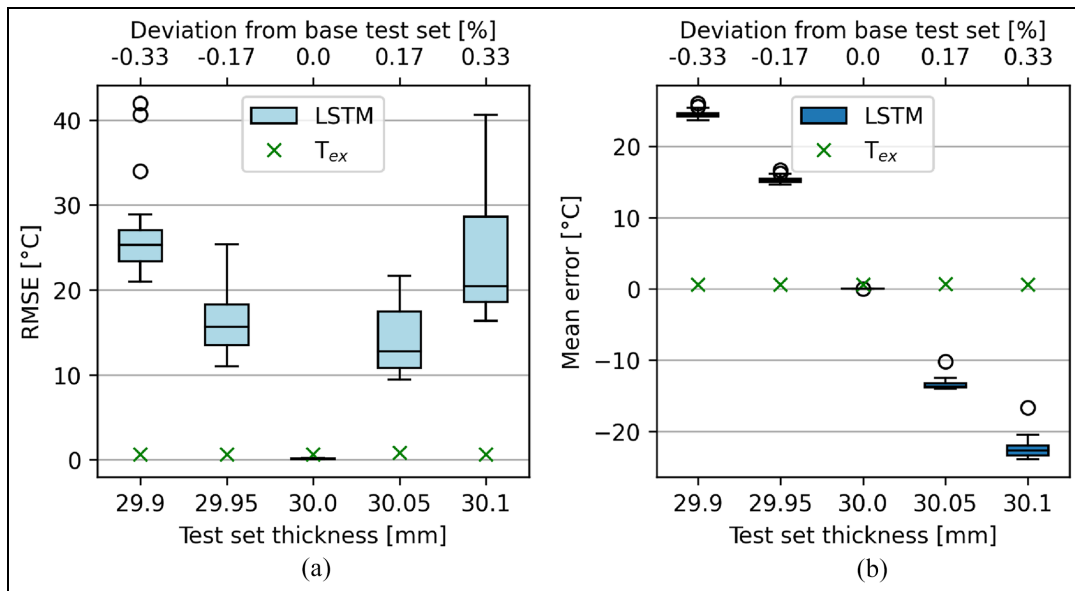
This can be intuitively explained by considering the following scenario. Consider a block of thickness  $L$  where the true  $v(T)$  coefficients are  $A_{\text{true}}$  and  $B_{\text{true}}$  and an LSTM network which is trained using the coefficients  $A_+$  and  $B_+$  where  $|A_{\text{true}}| < |A_+|$  and  $|B_{\text{true}}| < |B_+|$ . Although not strictly necessary, it will be assumed that the block is at some uniform temperature (for simplicity) such that the integral of Equation (2) becomes Equation (12).

$$\text{TOF} = \frac{2L}{v(T)} = \frac{2L}{AT + B} \quad (12)$$

Suppose the TOF is measured at this given temperature, then the apparent temperature could be computed by rearranging Equation (12) using  $A_+$  and  $B_+$ . The apparent temperature would underpredict the true temperature and hence this behaviour will be learnt by the network during training. The inverse of this effect was expected if  $|A_{\text{true}}| > |A_+|$  and  $|B_{\text{true}}| > |B_+|$  (OOD cases 1–3, Table 4). This behaviour is reflected in the positive trend between the mean error and the deviation of the  $v(T)$  coefficients shown in Figure 10(b).



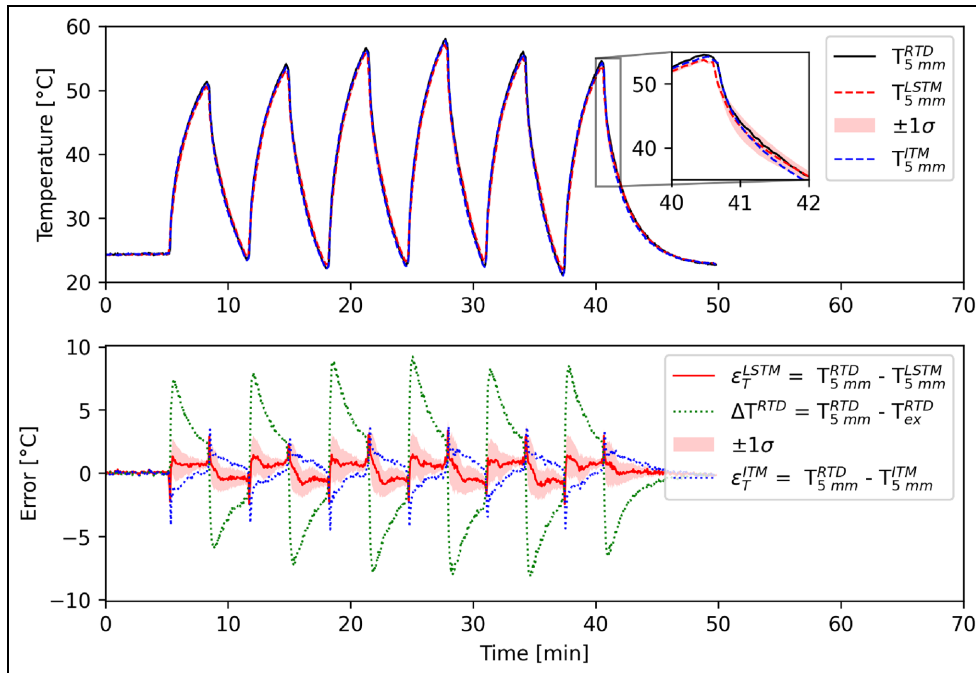
**Figure 10.** Performance metric boxplots for each trained network on the velocity OODD test sets. The percentage deviation refers to the  $v(T)$  coefficients given in Table 4. The box plots and crosses denote the metrics for the predictions by each of the LSTM networks and the external surface temperature, respectively. Prediction metrics for LSTM networks that were more than  $1.5 \times$  the interquartile range are shown by the circles. (a) RMSE, (b) Mean error. LSTM: long-short-term memory; RMSE: root mean square error; OODD: out-of-distribution data.



**Figure 11.** Performance metric boxplots for each trained network on the thickness OODD test sets. The base set (30.0, 0.0%) is included for reference. The box plots and crosses denote the metrics for the predictions by each of the LSTM networks and the external surface temperature, respectively. Prediction metrics for LSTM networks that were more than  $1.5 \times$  the interquartile range are shown by the circles. (a) RMSE, (b) mean error. LSTM: long-short-term memory; OODD: out-of-distribution data.

Figure 11 shows RMSE for the thickness OODD test sets. The metrics based on the external temperature are also shown as a reference. Similarly to the velocity

OODD cases, the absolute magnitude of both the RMSE and mean error increase as the parameters (thickness) of the test sets diverge from the value used



**Figure 12.** Experimental test set results. Top panel: RTD-measured temperature 5 mm from the water–metal interface ( $T_{5\text{mm}}^{\text{RTD}}$ ), LSTM ensemble mean predictions ( $T_{5\text{mm}}^{\text{LSTM}}$ ) and ITM<sup>23</sup> predictions ( $T_{5\text{mm}}^{\text{ITM}}$ ). Bottom panel: error of the LSTM predictions ( $\varepsilon_T^{\text{LSTM}}$ ), external temperature ( $\Delta T^{\text{RTD}}$ ) and ITM predictions ( $\varepsilon_T^{\text{ITM}}$ ). One standard deviation across the LSTM networks above and below the mean prediction is superimposed in both panels ( $\pm 1\sigma$ ). LSTM: long-short-term memory; ITM: inverse thermal modelling.

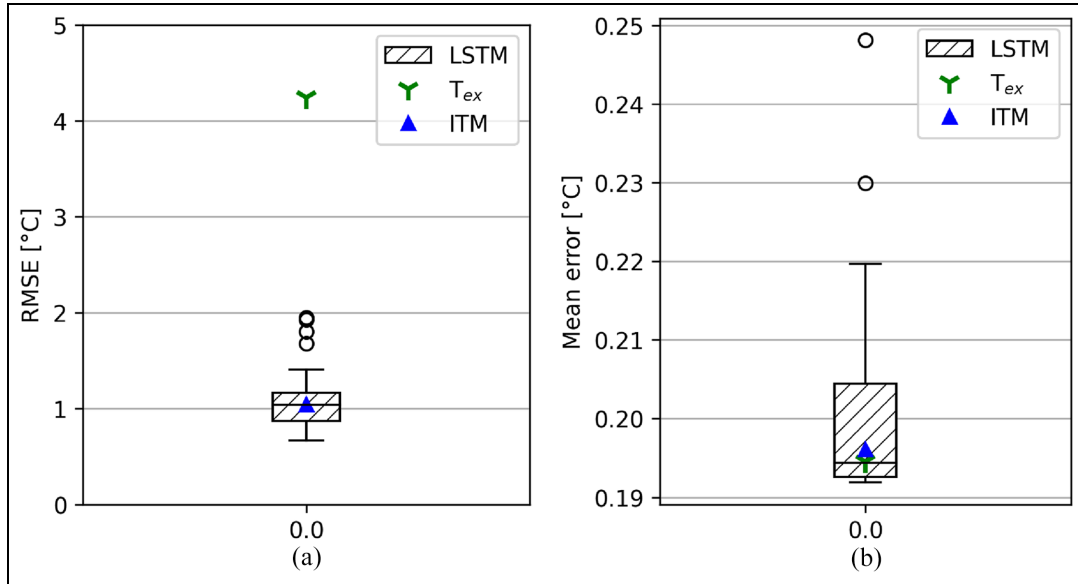
during training. However, it was expected that for the thickness ODD a positive deviation (ODD cases 4–6, Table 5) would cause the LSTM networks to over-predict the true temperature and therefore the error (according to Equation (10)) would be negative. This can again be explained by imagining a similar scenario as that described for the velocity ODD case except this block has true thickness  $L_{\text{true}}$  but an LSTM network is trained using a thickness  $L_-$  where  $|L_{\text{true}}| > |L_-|$ . The same uniform temperature assumption is made to yield Equation (12). Since the true block thickness is less than the assumed thickness during training, the TOF values will be smaller than expected and the computed temperature would over-predict the true temperature. The inverse of this effect was expected for a scenario where  $|L_{\text{true}}| < |L_+|$  (ODD cases 1 and 2, Table 5). This behaviour is reflected in the negative trend between the mean error and the deviation of the thickness shown in Figure 11(b).

In both ODD scenarios, the RMSE and mean error in temperature prediction both grow with the increasing deviation of the training parameters. This error manifests as a constant offset which is shown by the mean error (Figures 10(b) and 11(b)). However, the RMSE box plots for both scenarios (Figures 10(a) and 11(a)) demonstrate that using an ensemble of many networks can help to diagnose this issue since the SD

becomes larger as the deviation of parameters grows. This behaviour has previously been exploited for uncertainty quantification.<sup>33</sup> The influence of the size of the LSTM ensemble was not investigated, a smaller ensemble may be equally as informative whilst reducing computational expense. It should be noted that this issue of over/under prediction is also expected to be suffered by the ITM method and it is not possible to apply the ensemble method to ITM.

### Experimental studies

The top panel of Figure 12 shows the mean predicted temperatures across the 25 LSTM networks with one SD above and below this value superimposed. The predictions by the ITM method as well as the embedded RTD measurements are also shown as references. The bottom panel of Figure 12 shows the errors based on Equations (10) and (11). To achieve the predictions shown in Figure 12,  $0.1414 \mu\text{s}$  ( $\text{TOF}_{\text{adj}}$ ) was added to the TOF values before being passed to the LSTM networks. This correction ( $\text{TOF}_{\text{adj}}$ ) was needed because the networks were trained on a training set that assumed a thickness of  $L_{\text{train}} = 30 \text{ mm}$  but the true thickness was measured to be  $L = 29.77 \text{ mm}$ . The use of  $L_{\text{train}} = 30 \text{ mm}$  was chosen simply to (significantly) reduce the spatial resolution necessary to generate



**Figure 13.** Performance metrics for the LSTM networks, ITM and external surface temperature predictions made on the experimental. The box plots, tri-stars and triangles denote the metrics for predictions by each of the LSTM networks, the external surface temperature and ITM method, respectively. Prediction metrics for LSTM networks that were more than  $1.5\times$  the interquartile range are shown by the circles. (a) RMSE, (b) Mean error. LSTM: long-short-term memory; ITM: inverse thermal modelling.

training data and hence reduce the computation time. In effect, the experimental data was OODD compared with the training data. Using Equations (7) and (13) with the temperature during the first 5 min, for which the block is uniformly at  $24.3^{\circ}\text{C}$ , shows that adjusting the TOF by  $0.1414\ \mu\text{s}$  correctly calibrates the assumed training set thickness ( $L_{\text{train}} = 30\ \text{mm}$ ).

$$\begin{aligned} \text{TOF}_{\text{corr}} &= \text{TOF}_{\text{train}} + \text{TOF}_{\text{adj}} \\ &= \text{TOF}_{\text{train}} + 2 \times \frac{L_{\text{corr}} - L_{\text{train}}}{v(T)} \end{aligned} \quad (13)$$

The corrected thickness,  $L_{\text{corr}}$ , matches the true experimental thickness:  $L = 29.77$  and gave the best results on the experimental data set. Figures 13(a) and (b) show the RMSE and mean error respectively for LSTM, ITM and external surface temperature predictions. The median LSTM ensemble RMSE and mean error are both in close agreement with the ITM values, showing good accuracy. Furthermore, both methods achieved approximately  $4\times$  lower RMSE than if the external surface temperature was used.

The ITM implementation used a similar method to adjust the assumed thickness using the period in which the block temperature is uniform and then rearranging Equation (12) to obtain  $L$ . However, without a period of uniform temperature, this calibration is not possible and the ITM predictions would suffer the same offset experienced by the LSTM networks. Therefore, in the

context of a thickness OODD scenario, the ITM method is superior to the presented LSTM approach because a trained LSTM network has the training thickness ‘baked in’. The current machine learning architecture would typically require retraining the network(s) to change the assumed thickness; however, the adjustment of the TOF data given in Equation (13) can also resolve the thickness issue.

### Prediction time

The percentage ratio,  $R_t$ , between the raw mean computation time and real experimental time was calculated with Equation (14) where  $n_T$  is the number of data points (749) and  $f_s$  is the measurement sampling rate (0.25 Hz).

$$R_t = \frac{\mu_t}{n_T \times f_s} \times 100 \quad (14)$$

If  $R_t < 100\%$  then inversion can be done in real-time because computations are completed before the next data measurements are made. The mean adjusted computation time,  $C$ , was calculated with Equation (15) where  $\mu_t$ ,  $n_S$ , IF, EF are the average computation time, number of spatial points, interpolation factor and the ensemble factor, respectively.

$$C = \frac{\mu_t}{n_T \times n_S \times \text{IF} \times \text{EF}} \quad (15)$$

**Table 6.** Value of each correction factors used in Equation (15) to compute the adjusted mean computation times.

Method	$n_T$	$n_S$	IF	EF
LSTM ensemble	749	1	1	25
LSTM single	749	1	1	1
ITM	749	151	4000	1

LSTM: long-short-term memory; ITM: inverse thermal modelling.

**Table 7.** Raw mean computation time ( $\mu_t$ ), associated standard deviation ( $\sigma_t$ ), computation-to-real-time ratio ( $R_t$ ), mean computation time ( $C$ ) adjusted for a single point temperature estimate at one time instance and associated standard deviation ( $\sigma_C$ ) for 20 repeats of the LSTM network (full 25-network ensemble and a single network) and ITM method codes. Values of  $C$  were calculated using Equation (15) and Table 6.

Method	$\mu_t$ (s)	$\sigma_t$ (s)	$R_t$ (%)	$C$ ( $\mu$ s)	$\sigma_C$ ( $\mu$ s)
LSTM ensemble	5.12	0.0534	0.17	273	2.85
LSTM single	0.237	0.00473	0.008	316	6.32
ITM	423	6.05	14	0.936	0.0134

LSTM: long-short-term memory; ITM: inverse thermal modelling.

As in Equation (14),  $n_T$  is the number of data points. The interpolation factor accounts for the upsampling of the data from 4 to 0.001 s required by the ITM method. The ensemble factor accounts for the LSTM ensemble repeating the calculations for each network. These correction factors ensure the calculated times report the time taken to compute the temperature value at a single spatial location at a single time step. The values for the variables of Equation (15) are given in Table 6.

Table 7 shows both the raw ( $\mu_t$ ) and adjusted ( $C$ ) mean computation times for the LSTM ensemble, a single LSTM network and ITM method. The computation-to-real-time ratio ( $R_t$ ) is also shown as a percentage. The LSTM ensemble contained 25 members, and had a raw computation time approximately 25 $\times$  that of the single LSTM network implying that the computation time scales linearly with ensemble size. Comparing the raw computation times suggests that the single LSTM and ensemble are both significantly faster than ITM. However, the adjusted values ( $C$ ), which account for differences in sampling rates and the number of spatial nodes computed at each time step, show that for each computation of a single temporal and spatial point, the ITM method is approximately 292 and 338 times faster than the LSTM ensemble and single LSTM network, respectively. A multi-output LSTM network – one that makes temperature predictions for different spatial locations – may well be faster

than ITM. However, this hypothetical multi-output network would require a more complex architecture and hence a larger number of computations at each time step (due to an increase in trainable parameters). Given the larger number of computations, this complex LSTM is expected to remain slower than ITM. However, without further work this cannot be answered definitively. Despite this apparent large difference between the speeds of ITM and LSTM, the values of both  $R_t$  and  $C$  suggest either method would be fast enough for real-time computations, assuming similar performance is achieved on field-deployable hardware.

For the prediction of temperature-gradient-induced stresses some form of spatial resolution is required. ITM satisfies this requirement as the full (151-node) spatial grid must be computed at each time step. In contrast, the presented LSTM networks only computed a single spatial point. Therefore, spatial resolution could be achieved using multiple LSTMs for different spatial locations (spatial ensemble). Since the raw computation time scaled for LSTM (approximately) linearly with ensemble size, it is expected that a spatial ensemble would follow a similar behaviour. This is not to say that a spatial LSTM ensemble would need 151 members (to match ITM) since this discretisation is expected to exceed the spatial resolution necessary for predicting the thermo-mechanical stresses that cause HCTF. Hence, the impact of increasing LSTM spatial outputs is not expected to increase LSTM prediction times prohibitively, that is,  $R_t$  would remain below 100%. Further work is necessary to determine suitable spatial grid resolutions for both the ITM and LSTM methods.

Another factor to consider is that the experimental data sampling rate (0.25 Hz) would alias the maximum critical HCTF frequency at Civaux (1 Hz). Therefore, the measurement sampling frequency would have to be increased to at least 2 Hz<sup>39</sup> to properly capture 1 Hz fluctuations. At 2 Hz, the ITM method prediction time and  $R_t$  would remain (approximately) constant because the method already interpolates the data to 1 kHz (for numerical stability) which is capable of resolving 1 Hz. On the other hand, at higher sampling frequencies, the LSTM would need to perform a higher number of computations ( $R_t$  would increase). However, the LSTM method would only be required to predict on data sampled at 2 Hz, not 1 kHz to resolve 1 Hz fluctuations. Hence, the impact of increasing measurement sampling rates is not expected to increase LSTM prediction times prohibitively, that is,  $R_t$  would remain below 100%. Ultimately the choice of ITM versus LSTM depends on the required temporal and spatial resolutions with ITM becoming more favourable for significantly higher sampling rates.

The adjusted computation times for each of the methods are expected to be sufficient for real-time monitoring considering the critical range of HCTF frequencies (0.1–1 Hz) for Civaux. Nevertheless, further investigation is required to determine whether these speeds are achievable on lightweight, field-deployable hardware. During the development of the LSTM and ITM codes, speed and efficiency were not prioritised. Therefore, the execution speed of both codes might be improved through careful programming of the respective methods. For the LSTM ensemble such techniques might include:

- Reduction of the number of ensemble members
- Conversion of the ensemble into a single compact ‘multi-headed’ network<sup>45</sup>
- Network simplification by pruning<sup>46,47</sup> (also applicable to a single network)
- Performing a hyperparameter search to remove network complexity, for example, number of neurons (also applicable to a single network)

It is important to highlight that the same 25-member LSTM ensemble was used throughout the simulation and experimental studies. The networks were all trained using pure simulation data yet were able to make predictions on experimental data which have similar accuracy as the ITM method (with a correction factor to address the difference between the assumed training data thickness and true experimental thickness). Furthermore, the LSTM networks were trained on simulated data sampled at 0.5 Hz. When predictions were made on the experimental data which were sampled at 0.25 Hz, no interpolation was applied meaning the LSTM networks were predicting on sparse data.

## Conclusions

HCTF in NPP mixing zones is driven by large temperature gradients, that is, large differences between the interior and exterior wall temperatures in a pipe. It was previously shown that using the physics-based ITM method the inaccessible pipe wall temperature can be estimated to within 2°C by using the information from an external temperature measurement and the ultrasonic TOF. However, the ITM method was perceived to be relatively slow requiring 423 s to invert the full data set on a 12th Gen Intel core i7 processor CPU of a desktop PC. For field deployment less powerful processors would most likely be available and therefore this study investigated whether LSTM machine learning architecture would be less computationally intensive

than the ITM method, whilst achieving comparable accuracy.

It was found that relative to a resistance temperature detector measurement, the 25-member LSTM ensemble achieved an ensemble median RMSE of 1.04°C and an ensemble median mean error of 0.194°C. This is almost identical to the performance of the ITM method which achieved a RMSE and mean error of 1.04°C and 0.196°C, respectively. These key metrics demonstrate that LSTM networks can perform as well as the ITM method if parameters such as the component thickness and velocity–temperature relationship coefficients used during training are in perfect agreement with the (unseen) test set. However, differences between the training and testing sets as small as  $\pm 0.01\%$  of the velocity–temperature relationship coefficients or  $\pm 0.17\%$  of the thickness caused the accuracy of the predictions to drop significantly. In both cases, these errors cause a simple offset of the predicted temperatures. Similarly to the ITM method, periods of constant temperature can be used to correct the offset caused by thickness discrepancies between the training data and real-world data. SD of the predictions made by an ensemble of 25 independent networks was found to be a clear indicator of the magnitude of errors in the predictions.

The aspects that affect computation time for a temperature prediction using both the LSTM and ITM methods were also discussed. While, for the implementations in this work, the LSTM looked considerably faster for performing temperature estimates for a full data set, the ITM method actually had a lower computation time per temperature estimate. However, for the stability of the ITM method it is required that it performs predictions at very small time steps, which therefore requires many interim computations if the sampling rate is relatively slow, that is, 0.25 Hz. This means that the ITM computation time will be unaffected by an increase in sampling rate, unless the increase exceeds the stable ITM time step. On the other hand, the LSTM method would need to perform more computations for an increased sampling rate, increasing computation time. A similar argument will apply in space; for the prediction of temperature-gradient-induced stresses some form of spatial resolution will be required. The ITM requires the use of a spatial grid of  $N$  points ( $N = 151$  in this work), whereas multiple LSTM networks would need to be run separately to perform another spatial prediction, increasing computation time. Finally, the behaviour of the ITM can be fully interpreted and explained whilst that of the LSTM is a black box. Therefore, the authors conclude that the ITM should be favoured over the LSTM for a high frequency field application.





### Declaration of conflicting interests

The author(s) declared no potential conflicts of interest with respect to the research, authorship, and/or publication of this article.

### Funding

The author(s) disclosed receipt of the following financial support for the research, authorship, and/or publication of this article: The authors would like to acknowledge funding from EPSRC for project funding as part of the FIND CDT (EP/S023275/1).

### ORCID iDs

Laurence Clarkson  <https://orcid.org/0000-0003-0086-2092>  
Yifeng Zhang  <https://orcid.org/0000-0002-9426-7536>

### References

- Courtin S. High cycle thermal fatigue damage prediction in mixing zones of nuclear power plants: engineering issues illustrated on the fatter case. *Procedia Eng* 2013; 66: 240–249.
- Fontes JP, Braillard O, Cartier O, et al. High-cycle thermal fatigue in mixing zones: investigations on heat transfer coefficient and temperature fields in PWR mixing configurations. In: *18th international conference on nuclear engineering*, Xi'an, China, 17–21 May 2010, pp. 179–186.
- George ED. *Mechanical metallurgy*. New York, NY: McGraw-Hill, 1961.
- Was GS and Ukai S. *Austenitic stainless steels*. Amsterdam, The Netherlands: Elsevier, 2019.
- Maziasz PJ and Busby JT. Properties of austenitic steels for nuclear reactor applications. *Compreh Nuclear Mater* 2020; 7: 303–318. DOI: 10.1016/B978-0-12-803581-8.11736-9.
- Couturier J and Schwarz M. Current state of research on pressurized water reactor safety. *EDP Sci* 2018. DOI: 10.1051/978-2-7598-2164-8.
- World Information Service on Energy. France: serious accident at Civaux-1, <https://www.wiseinternational.org/nuclear-monitor/495/france-serious-accident-civaux-1> (1998, accessed 19 November 2021).
- Cipière MF and Le Duff JA. Thermal fatigue experience in French piping: influence of surface condition and weld local geometry. *Weld World* 2002; 46: 23–27.
- Timperi A. Development of a spectrum method for modelling fatigue due to thermal mixing. *Nucl Eng Des* 2018; 331: 136–146.
- Radu V, Paffumi E, Taylor N, et al. A study on fatigue crack growth in the high cycle domain assuming sinusoidal thermal loading. *Int J Press Vessels Pip* 2009; 86(12): 818–829.
- Basu S and Debnath AK. Chapter 2 – Main equipment. In: Basu S and Debnath AK (eds.) *Power plant instrumentation and control handbook*. 2nd ed. Boston, MA: Academic Press, 2019, pp. 41–147.
- Taler J, Taler D, Kaczmarski K, et al. Monitoring of thermal stresses in pressure components based on the wall temperature measurement. *Energy* 2018; 160: 500–519.
- Taler J, Dzierwa P, Jaremkiwicz M, et al. Thermal stress monitoring in thick walled pressure components of steam boilers. *Energy* 2019; 175: 645–666.
- Murakami Y. 2 – Stress concentration. In: Murakami Y (ed.) *Metal fatigue*. 2nd ed. Boston, MA: Academic Press, 2019, pp. 13–27.
- Wang Y, Zou F and Cegla FB. Acoustic waveguides: an attractive alternative for accurate and robust contact thermometry. *Sens Actuator A Phys* 2018; 270: 84–88.
- Bergholz S, Jouan B, Rudolph J, et al. Automatic fatigue monitoring based on real loads and consideration of EAF: live demonstration. In: *ASME 2013 pressure vessels and piping conference*, Paris, France. DOI: 10.1115/PVP2013-97236.
- Rudolph J, Bergholz S, Heinz B, et al. *AREVA fatigue concept: a three stage approach to the fatigue assessment of power plant components*. London, UK: InTech, 2012. DOI: 10.5772/37029.
- Zhang Y, Cegla F and Corcoran J. Ultrasonic monitoring of pipe wall interior surface temperature. *Struct Health Monitor* 2020; 20: 2476–2492.
- Granta Design Limited. Ansys Granta EduPack, <https://www.ansys.com/en-gb/products/materials/granta-edupack> (accessed 20 August 2022).
- Dahlberg M, Nilsson KF, Taylor N, et al. Development of a European procedure for assessment of high cycle thermal fatigue in light water reactors: final report of the NESC-thermal fatigue project 2007, <https://op.europa.eu/s/yfbF> (accessed 27 October 2007).
- Gajdacs A. *High accuracy ultrasonic degradation monitoring*. PhD Thesis, Imperial College London, Mechanical Engineering, London, UK, 2015.
- British Standards Institution. *BS EN IEC 60751:2022. Industrial platinum resistance thermometers and platinum temperature sensors*. UK: BSI, 2022.
- Zhang Y and Cegla F. Co-located dual-wave ultrasonics for component thickness and temperature distribution monitoring. *Struct Health Monit* 2022; 22: 1090–1104.
- Ihara I, Yamada H, Kosugi A, et al. New ultrasonic thermometry and its applications to temperature profiling of heated materials. In: *2011 fifth international conference on sensing technology*, Palmerston North, New Zealand, pp. 60–65. New York, NY: IEEE.
- Munir N, Park J, Kim HJ, et al. Performance enhancement of convolutional neural network for ultrasonic flaw classification by adopting autoencoder. *NDT E Int* 2020; 111: 102218.
- Chapon A, Pereira D, Toews M, et al. Deconvolution of ultrasonic signals using a convolutional neural network. *Ultrasonics* 2021; 111: 106312.
- Cantero-Chinchilla S, Wilcox PD and Croxford AJ. A deep learning based methodology for artefact identification and suppression with application to ultrasonic images. *NDT E Int* 2022; 126: 102575.

28. Tu XL, Pyle RJ, Croxford AJ, et al. Potential and limitations of NARX for defect detection in guided wave signals. *Struct Health Monit* 2023; 22: 1863–1875.
29. Bevan RL, Zhang J, Budyn N, et al. Experimental quantification of noise in linear ultrasonic imaging. *IEEE Trans Ultras Ferroelectr Freq Control* 2019; 66: 79–90.
30. Pyle RJ, Bevan RL, Hughes RR, et al. Deep learning for ultrasonic crack characterization in NDE. *IEEE Trans Ultras Ferroelectr Freq Control* 2021; 68: 1854–1865.
31. Hasan MM, Pourmousavi SA, Ardakani AJ, et al. A data-driven approach to estimate battery cell temperature using a nonlinear autoregressive exogenous neural network model. *J Energy Storage* 2020; 32: 101879.
32. Tu JV. Advantages and disadvantages of using artificial neural networks versus logistic regression for predicting medical outcomes. *J Clin Epidemiol* 1996; 49: 1225–1231.
33. Pyle RJ, Hughes RR, Ali AAS, et al. Uncertainty quantification for deep learning in ultrasonic crack characterization. *IEEE Trans Ultras Ferroelectr Freq Control* 2022; 69: 2339–2351.
34. Hochreiter S. The vanishing gradient problem during learning recurrent neural nets and problem solutions. *Int J Uncertain Fuzziness Knowledge Based Syst* 1998; 6: 107–116.
35. Hochreiter S and Schmidhuber J. Long short-term memory. *Neural Comput* 1997; 9: 1735–1780.
36. Cantero-Chinchilla S, Wilcox PD and Croxford AJ. Deep learning in automated ultrasonic NDE – developments, axioms and opportunities. *NDT E Int* 2022; 131: 102703.
37. DiPietro R and Hager GD. Chapter 21 – Deep learning: RNNs and LSTM. In: Zhou SK, Rueckert D and Fichtinger G (eds.) *Handbook of medical image computing and computer assisted intervention*. Boston, MA: Academic Press, 1990, pp. 503–519.
38. Wagner W and Kretzschmar HJ. *International steam tables*. Berlin, Heidelberg: Springer, 2008.
39. Shannon C. Communication in the presence of noise. *Proc IRE* 1949; 37(1): 10–21.
40. Python Software Foundation. Python, <https://www.python.org/> (accessed 24 October 2022).
41. Chollet F, et al. Keras, <https://github.com/fchollet/keras>; [https://keras.io/getting\\_started/faq/](https://keras.io/getting_started/faq/) (2015, accessed 9 February 2023).
42. Kingma DP and Ba J. Adam: a method for stochastic optimization, <http://arxiv.org/abs/1412.6980> (2014, accessed 15 March 2023).
43. MathWorks. *Matlab*. Massachusetts, USA: The MathWorks, Inc. 2021.
44. Python Software Foundation. Time – time access and conversions, [https://docs.python.org/3/library/time.html#time.perf\\_counter](https://docs.python.org/3/library/time.html#time.perf_counter) (2023, accessed 24 October 2022).
45. Tran L, Veeling BS, Roth K, et al. *Hydra: preserving ensemble diversity for model distillation*. *arXiv:2001.04694*, 2020; <https://arxiv.org/abs/2001.04694>.
46. Blalock D, Ortiz JGG, Frankle J, et al. What is the state of neural network pruning? *arXiv:2003.03033*, 2020; <http://arxiv.org/abs/2003.03033>.
47. Molchanov P, Tyree S, Karras T, et al. Pruning convolutional neural networks for resource efficient inference. *arXiv:1611.06440*, 2016; <http://arxiv.org/abs/1611.06440>.