Electronic Supplementary Materials

For https://doi.org/10.1631/jzus.A2300441

Temperature field prediction of steel-concrete composite decks using TVFEMD-stacking ensemble algorithm

Benkun TAN[1,2✉], Da WANG[1,3], Jialin SHI[3], Lianqi ZHANG[1]

[1]*School of Civil Engineering, Changsha University of Science & Technology, Changsha 410114, China*

[2]*School of Civil and Architecture Engineering, Hunan University of Arts and Science, Changde 415000, China*

[3]*School of Civil Engineering, Central South University of Forestry and Technology, Changsha 410004, China*

✉ Benkun TAN, tanbenkun@yeah.net

Section S1

S1.1 Heat conduction theory

The internal heat transfer process of a SCCDs can be calculated using the Fourier differential equation for thermal conductivity. Assuming that the temperature distribution *T* (°C) of a SCCDs is a function of time and space, denoted by (*x*, *y*, *z*, *t*), then the heat transfer differential equation is represented by Eq. (S1).

$$\rho c\frac{\partial T}{\partial t}=k\left(\frac{\partial^2 T}{\partial x^2}+\frac{\partial^2 T}{\partial y^2}+\frac{\partial^2 T}{\partial z^2}\right). \tag{S1}$$

where ρ is the density of the material (kg/m^3), c is the specific heat capacity (J/kg/°C), k is the thermal conductivity (W/m/K).

S1.2 Boundary conditions and initial conditions

As the structural temperature field of an operational SCCDs is being analyzed, it is reasonable to assume that the structure does not have an internal heat source. The boundary conditions required to solve Eq. (S1) can be determined using the following equation:

$$\rho ck\left(\frac{\partial T}{\partial x}l_x+\frac{\partial T}{\partial y}l_y+\frac{\partial T}{\partial z}l_z\right)+q=0. \tag{S2}$$

where l_x, l_y, and l_z represent the directional cosines in the x, y, and z directions, respectively. Meanwhile, q refers to the heat flow density at the structure's boundary, which changes over time t.

Thermal exchange between the SCCDs and its surrounding environment is primarily influenced by convective, long-wave radiative, and short-wave radiative heat exchange. This phenomenon can be quantified using the following equation:

$$q(t) = q_{\mathrm{c}}(t) + q_{\mathrm{r}}(t) + q_{\mathrm{s}}(t). \tag{S3}$$

where $q_{\mathrm{c}}(t)$ is the heat flow density of convective heat transfer, $q_{\mathrm{r}}(t)$ is the heat flow density of long-wave radiation heat transfer, and $q_{\mathrm{s}}(t)$ is the heat flow density of short-wave radiation heat transfer.

The formula for the convective heat transfer density $q_{\mathrm{c}}(t)$ is shown in Eq. (S4) and the value is determined by the temperature difference between the surface of the structure and the surrounding environment.

$$q_{\mathrm{c}}(t) = h_{\mathrm{c}}[T_{\mathrm{s}}(t) - T_{\mathrm{a}}(t)]. \tag{S4}$$

where $T_{\mathrm{s}}(t)$ represents the surface temperature of the structure, $T_{\mathrm{a}}(t)$ represents the ambient temperature, and the surface heat exchange coefficient, h_{c} (m/s), is related to the orientation of the structure and the wind speed on its surface. Typically, h_{c} is calculated using empirical formulas, but in this study, we corrected the parameters based on references and measured data.

For long-wave radiative heat transfer, calculations can be based on the following equation according to the Stefan-Boltzmann law.

$$q_{\mathrm{r}}(t) = A_{\mathrm{m}} C_0 \left([273.16 + T_{\mathrm{a}}(t)]^4 - (273.16 + T)^4 \right). \tag{S5}$$

where A_{m} represents the coefficient of long-wave radiation, C_0 is the Stefan-Boltzmann constant, 5.67×10^{-8} W m^{-2} K^{-4}.

The thermal environment and SCCDs are depicted in Fig. S1. The relationship between the total shortwave radiation intensity and the shortwave radiation heat exchange value can be expressed by Eq. (S6), and it is generally considered to include direct solar radiation, scattering from the sky, and reflection from the ground, which can be calculated by Eq. (S7).

$$q_{\mathrm{s}}(t) = A_{\mathrm{s}} I_{\mathrm{T}}, \tag{S6}$$

$$I_{\mathrm{T}} = I_{\mathrm{B}} \frac{\cos\theta}{\sin h} + \frac{1+\cos\beta}{2} I_{\mathrm{D}} + \frac{1-\cos\beta}{2} r_{\mathrm{e}} \left(I_{\mathrm{B}} + I_{\mathrm{D}} \right). \tag{S7}$$

where A_{s} is the structure surface of the short-wave radiation absorption rate, I_{T} is the structure surface by the total intensity of radiation, I_{B} and I_{D} indicate the horizontal surface direct solar radiation intensity and horizontal surface solar scattering radiation intensity respectively, r_{e} represents the surface reflectivity, β represents the angle between the inclined surface and the horizontal plane, θ represents the angle of incidence of the sun, h represents the solar altitude angle.

S1.3 Calculation of shadow length caused by flange plate

Under actual sunlight conditions, the radiation received by bridge structures is affected by the structure itself or surrounding objects blocking the radiation. Therefore, it is necessary to consider the shading effect in numerical simulations. The shadow lengths of the steel beam web (d) and bottom flange (d_{bot}) caused by shading of the cantilever part of the SCCDs can be calculated using the Eq. (S8) and Eq. (S9) ,respectively.

$$d = \frac{w_{\text{top}} \tan h}{\cos(\gamma_{\text{s}} - \gamma)\sin\beta + \cos\beta \tan h}, \tag{S8}$$

$$d_{\text{bot}} = \frac{d - d_{\text{web}}}{\tan h}. \tag{S9}$$

where w_{top} is the length of the cantilever section of the bridge deck flange, where d_{web} is the depth of the web, γ_s represents the solar azimuth angle, and γ denotes the angle formed by the normal direction of the inclined surface and the southward direction.

Section S2

Step 1: Using the Hebelt transform on the temperature time series $T(t)$ to derive the instantaneous amplitude $W(t)$ and the instantaneous phase $\psi(t)$.

$$W(t) = \sqrt{T(t)^2 + T(t)^2}, \tag{S10}$$

$$\psi(t) = \arctan[T(t) / T(t)], \tag{S11}$$

where $\tilde{T}(t)$ is the Hebelt transformed form of the original sequence $T(t)$.

Step 2: Calculate the instantaneous amplitude W(t) for the local maximum value $W(\{\max\})$ and the minimum value $W(\{\min\})$ respectively.

Step 3: Based on the curves $\beta_1(t)$ and $\beta_2(t)$ obtained from the interpolation of the set of very large and very small points, the instantaneous mean $w_1(t)$ and the instantaneous envelope $w_2(t)$ are derived, and their corresponding mathematical equations are shown below:

$$w_1(t) = [\beta_1(t) + \beta_2(t)] / 2, \tag{S12}$$

$$w_2(t) = [\beta_2(t) - \beta_1(t)] / 2, \tag{S13}$$

Step 4: Interpolate $\psi_1^{'}(\{t_{\min}\})W^2(\{t_{\min}\})$ and $\psi_2^{'}(\{t_{\max}\})W^2(\{t_{\max}\})$ to estimate $x_1(t)$ and $x_2(t)$, respectively, and the instantaneous frequencies $\psi_1^{'}(t)$ and $\psi_2^{'}(t)$ are calculated as follows:

$$\psi_1'(t) = \frac{\chi_1(t)}{2w_1^2(t) - 2w_1(t)w_2(t)} + \frac{\chi_2(t)}{2w_1^2(t) + 2w_1(t)w_2(t)}, \tag{S14}$$

$$\psi_2'(t) = \frac{\chi_1(t)}{2w_2^2(t) - 2w_1(t)w_2(t)} + \frac{\chi_2(t)}{2w_2^2(t) + 2w_1(t)w_2(t)}, \tag{S15}$$

Step 5: Calculate the local cut-off frequency $\psi_{\text{bis}}^{'}(t)$ from Eq. (S16).

$$\psi'_{\mathrm{bis}}(t)=\frac{\left[\psi'_1(t)+\psi'_2(t)\right]}{2}=\frac{\chi_2(t)-\chi_1(t)}{4w_1(t)w_2(t)},\tag{S16}$$

Step 6: After deriving the local cut-off frequency $\psi'_{\mathrm{bis}}(t)$, the signal $S(t)$ is calculated according to Eq. (S17). A time-varying filter is constructed by using the local extrema of $S(t)$ as nodes, and B-sample interpolation is employed to filter the original time series $T(t)$ and generate the filtering result $c^1(t)$.

$$S(t)=\cos[\int\psi'_{\mathrm{bis}}(t)dt],\tag{S17}$$

Step 7: Define a stopping criterion such that $T(t)$ is considered an IMF component when the stopping condition $\theta(t)\le\varepsilon$ is satisfied. Otherwise, let $T(t)$-$c^1(t)$ be the new input signal and repeat steps one to seven until the stopping criterion is satisfied. The stopping criterion $\theta(t)$ is calculated as follows:

$$\theta(t)=\frac{B_{\mathrm{Loughlin}}(t)}{\psi_{\mathrm{avg}}(t)},\tag{S18}$$

where $B_{\mathrm{Loughlin}}(t)$ is the Loughlin instantaneous bandwidth and $\psi_{\mathrm{avg}}(t)$ is the weighted mean instantaneous frequency.

Ultimately, the original time series $T(t)$ is decomposed by TVFEMD to obtain J IMF components [$c^1(t)$, $c^2(t)$, … , $c^J(t)$], and satisfies the following relation:

$$T(t)=\sum_{i=1}^{J}c^i(t).\tag{S19}$$

where: $c^i(t)$ is the i-th IMF component and the last term is referred to as the residual term.

Section S3

Step 1: Define the algorithm parameters m and r, where m is the length of the comparison vector and r denotes the metric value for determining the similarity. IMF is constructed as an m-dimensional vector, i.e.

$$X(i)=[x(i)\ ,\ x(i+1)\ ,\ldots,x(i+m-1)],\tag{S20}$$

where $i = 1, 2, ..., N\text{-}m+1$ and N denotes the total number of samples.

Step 2: The distance between $X(i)$ and $X(j)$ is defined as $d[X(i), X(j)]$, which corresponds to the absolute value of the maximum difference in the elements, where i in Eq. (S21) is not equal to j.

$$d\left[X(i),X(j)\right]=\max_{k\in(0,m-1)}\left|x(i+k)-x(j+k)\right|,\tag{S21}$$

Step 3: Count the number of $d_m[X(i), X(j)] < r$ in all i values and write down A_i, then increase the number of dimensions to m+1, repeat the above steps and count the number of $d_{m+1}[X(i),X(j)] < r$ (m+1 dimensions) in all i values and write down B_i. The sample entropy result can then be calculated according to the following equation.

$$SE = -\ln\left[\frac{B_{m+1}(r)}{A_m(r)}\right], \tag{S22}$$

where $B_{m+1}(r)$ and $A_m(r)$ are each calculated as follows:

$$A_m(r) = \frac{1}{N-m+1}\sum_{i=1}^{N-m+1}\frac{1}{N-m}A_i, \tag{S23}$$

$$B_{m+1}(r) = \frac{1}{N-m}\sum_{i=1}^{N-m}\frac{1}{N-m-1}B_i. \tag{S24}$$

Section S4

After initialization, the weak learner needs to be trained iteratively, and for round t iteration, the objective function Eq. (S25) needs to be solved and the residual $r_i^{(t)}$ needs to be calculated.

$$O_{\text{obj}}^{(t)} = \sum_{i=1}^{N} l[y_i, \hat{y}_i^{(t-1)} + f_t(x_i)] + \Omega(f_t), \tag{S25}$$

$$r_i^{(t)} = y_i - \hat{y}_i^{(t-1)}. \tag{S26}$$

where $l[y_i, \hat{y}_i^{(t-1)}+f_t(x_i)]$ is a loss function that measures the predicted value of $\hat{y}_i^{(t-1)}$ and the fit of the current weak learner $f_t(x_i)$ to the true value of y_i after round t-1, $\Omega(h_t)$ is a regularization term that is used to limit the complexity of the weak learner and prevent over-fitting.

A decision tree is trained using a greedy algorithm. First, a feature and split point are selected, and the dataset is divided into left and right subtrees. The predicted value of each leaf node is calculated as the average of the residuals of all samples in that node. Then, the G_{Gain} is calculated, and a new tree is generated by selecting the feature and split point with the greatest gain.

$$G_{\text{Gain}} = \frac{1}{2}\left[\frac{\sum_{i\in I_L} r_i}{\sum_{i\in I_L} w_i + \lambda} + \frac{\sum_{i\in I_R} r_i}{\sum_{i\in I_R} w_i + \lambda} - \frac{(\sum_{i\in I} r_i)^2}{\sum_{i\in I} w_i + \lambda}\right] - \gamma. \tag{S27}$$

where I denotes the set of samples at the current node, r_i denotes the current residual, I_L and I_R denote the set of samples in the left and right subtrees respectively, w_i is the weight of sample I, and λ and γ are the regularization parameters to limit the complexity of the decision tree respectively.

After training the t-th weak learner, the predicted values of the model are updated as:

$$\hat{y}_i^{(t)} = \hat{y}_i^{(t-1)} + \eta f_t(x_i). \tag{S28}$$

where η is the learning rate and represents the degree of contribution of each weak learner.

The above steps are repeated until the tree reaches its maximum depth or no further gain is possible, and all the weak learners are combined into a powerful integrated model for making predictions. Specifically, we weight the predicted values of the M weak learners to sum up as the final prediction:

$$\hat{y}_i = \sum_{m=1}^{M} \eta f_m(x_i), \tag{S29}$$

The loss function is complex and cannot be directly optimized, so a Taylor expansion is used to approximate the objective function. This approximation allows the gradient and second-order derivative of the objective function to be calculated, which updates the weights of the decision tree.

$$O_{\text{obj}}{}^{(t)} \approx \sum_{i=1}^{n} [g_i f_{t-1}(x_i) + \frac{1}{2} h_i f_{t-1}^2(x_i)] + \gamma T + \frac{1}{2}\lambda \sum_{j=1}^{T} w_j^2, \tag{S30}$$

where: g_i and h_i denote the first and second order derivatives of the loss function at the sample x_i, respectively, $f_{t\text{-}1}(x_i)$ denotes the weighted sum of the predicted values of all trees for the i-th sample in the previous t-1 iterations, w_j denotes the weight of the j-th leaf node of the kth tree, γ and λ denote the hyperparameters of the regularization term, respectively.

By letting Eq. (S31), Eq. (S32) the final objective function is transformed into a quadratic function as shown in Eq. (S33):

$$G_j = \sum_{i \in I_j} g_i, \tag{S31}$$

$$H_j = \sum_{i \in I_j} h_i, \tag{S32}$$

$$O_{\text{obj}}{}^{(t)} = \sum_{j=1}^{T} \left[G_j \omega_j + \frac{1}{2}(H_j + \lambda)\omega_j^2 \right] + \gamma T \tag{S33}$$

Therefore, the best w_j and the best objective reduction O_{obj}* are derived as shown in Eqs. (S34) and (S35), respectively:

$$\omega_j^* = -\frac{G_j}{H_{j+\lambda}}, \tag{S34}$$

$$O_{\text{obj}}{}^* = -\frac{1}{2}\sum_{j=1}^{T} \frac{G_j^2}{H_j + \lambda} + \gamma T. \tag{S35}$$

Feature importance can be scored based on the number of splits in the decision tree and the split gain. Each split of the decision tree selects the best feature and calculates the split gain of that feature, which reflects its significance for the sample. We can then accumulate the split gain of each feature across all decision trees to evaluate the importance of the i-th feature.

$$s_i = \sum_{t=1}^{T} \omega_t \times 1_{\{f_t = i\}}. \tag{S36}$$

where T denotes the number of decision trees, ω_t is the weight of the t-th decision tree, and $1_{\{ft=i\}}$ is an indicator function where $1_{\{ft=i\}}=1$ if the t-th decision tree chooses feature i as the splitting feature and 0 otherwise.

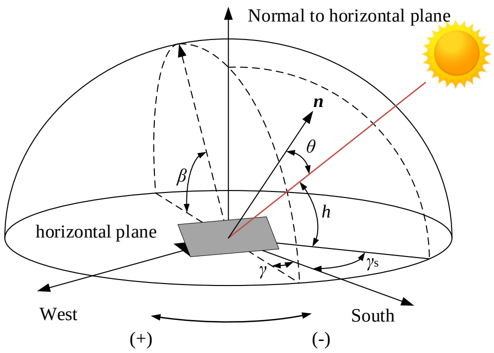

Fig. S1 Relative position of the sun to a horizontal plane on the ground.

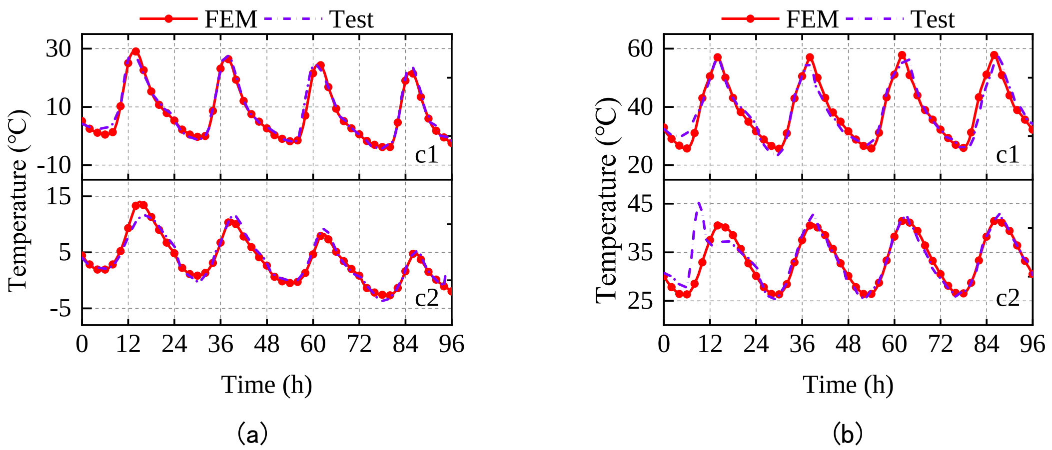

Fig. S2 FE model calculation results verification: (a) winter (2013.01.01~2013.01.04); (b) summer (2013.08.10~2013.8.13)

Table S1 Boundary conditions and calculation parameters

Category	Type of radiation	h_c (m/s)	ρ (kg/m³)	k (W/m/K)	c (J/kg/°C)	A_m	A_s
Asphalt concrete	$R_d + R_s$	14.33	2360	1.05	1168	0.95	0.9
Side surface of bridge deck	$R_d + R_s + R_r$	11.33	2549	1.70	920	0.9	0.7
Underside of bridge deck	R_r	9.83					
Side of steel beam	$(R_d) + R_s + R_r$	13.29					
Underside of steel beam	R_r	11.05	7850	53.2	460	0.6	0.6
Top surface of lower flange of steel beam	$(R_d) + (R_s)$	11.05					

Table S2 Hyperparameter settings for each ML model after optimization

Model	Hyperparameters	Original	Timelag-based	EMD-based	TVFEM-based
RF	(n_estimators, max_features min_samples_split, min_samples_leaf,)	(566, none, 5, 3)	(769, none, 4, 5)	(646, sqrt, 5, 2)	(54, none, 5, 2)
SVR	(kernel, C, gamma, epsilon)	(rbf, 142, 0.0008, 0.002)	(rbf, 83, 0.04, 0.04)	(rbf, 142, 0.008, 0.02)	(rbf, 142, 0.0008, 0.002)
MLP	(hidden_layer_sizes, alpha, max_iter, learning_rate, activation, batch_size, learning_rate_init, solver)	[100, 3, 500, invscaling, relu, 64, 0.01, adam]	[100, 2, 500, invscaling, relu, 64, 0.01, adam]	[(50, 50), 5.8, 1000, constant, relu, 256, 0.02, adam]	[100, 4.1, 1000, constant, relu, 256, 0.01, sgd]
GBR	(n_estimators, min_samples_split, min_samples_leaf, max_depth, learning_rate)	(150, 14, 3, 6, 0.1)	(150, 14, 3, 6, 0.1)	(150, 14, 3, 6, 0.1)	(150, 14, 3, 6, 0.1)
XGBoost	(colsample_bytree, gamma, learning_rate, max_depth, min_child_weight, n_estimators, reg_alpha, reg_lambda, subsample)	(0.9, 7.0, 0.4, 5, 4, 219, 0.0006, 3.7, 0.6)	(0.8, 8.8, 0.15, 10, 9, 72, 1.4, 3.5, 0.6)	(0.5, 9.2, 0.05, 5, 3, 459, 1.6, 7.5, 0.5)	(0.7, 9.4, 0.08, 3, 8, 379, 4.6, 5.3, 0.7)

Table S3 Comparison of single model prediction performance

Model	Original			Timelag-based			EMD-based			TVFEMD-based		
	R^2	MAE	RMSE	R^2	MAE	RMSE	R^2	MAE	RMSE	R^2	MAE	RMSE
RF	0.89	2.11	2.66	0.95	1.40	1.79	0.93	1.71	2.10	0.97	1.00	1.26
SVR	0.86	2.27	2.86	0.94	1.40	1.84	0.93	1.58	2.11	0.95	1.45	1.80
MLP	0.87	2.19	2.75	0.94	1.63	2.09	0.95	1.28	1.70	0.97	1.13	1.42
GBR	0.88	2.16	2.68	0.95	1.34	1.68	0.96	1.10	1.46	0.97	0.98	1.23
XGBoost	0.89	2.06	2.54	0.95	1.32	1.72	0.97	1.07	1.37	0.97	0.96	1.24

Table S4 Stacking integrated model prediction performance

Model	Original			Timelag-based			EMD-based			TVFEMD-based		
	R^2	MAE	RMSE	R^2	MAE	RMSE	R^2	MAE	RMSE	R^2	MAE	RMSE
Stacking	0.91	1.85	2.34	0.96	1.20	1.51	0.97	1.03	1.34	0.98	0.79	1.01

Table S5 Relative error calculation results

	EMD-based					TVFEMD-based				
	Q75%	Median	Q25%	Average	IQR	Q75%	Median	Q25%	Average	IQR
RF	20.23	6.73	0.21	15.37	20.02	5.74	0.48	-4.94	0.35	10.68
SVR	1.15	-4.3	-10.96	-3.68	12.11	-0.42	-5.70	-17.24	-12.37	16.82
MLP	7.96	0.86	-3.94	3.39	11.9	2.56	-2.17	-7.13	-2.12	9.69
GBR	0.56	-4.58	-10.57	-4.57	11.13	0.13	-4.36	-8.87	-3.43	9.00
XGBoost	1.99	-2.72	-9.22	-3.7	11.21	4.60	-0.72	-6.37	-0.51	10.97
Stacking	5.23	-0.46	-5.7	0.89	10.93	3.80	-0.58	-4.68	0.49	8.48

Table S6 Parameter estimation and goodness-of-fit test results for the temperature distribution model

	Actual value			EMD-Stacking			TVFEMD-Stacking		
i	w_i	μ_i	σ_i	w_i	μ_i	σ_i	w_i	μ_i	σ_i
1	0.55	15.27	15.39	0.18	27.69	17.64	0.63	15.54	18.31
2	0.25	27.06	21.05	0.24	8.15	6.42	0.19	27.80	13.76
3	0.20	7.38	6.38	0.58	15.99	15.64	0.18	7.33	4.74
Results	Adopted			Adopted			Adopted		
RMSE(PDF)	0.0024			0.0048			0.0038		
RMSE(CDF)	0.0043			0.0150			0.0086		