Applied Thermal Engineering 40 (2012) 56–63

Contents lists available at SciVerse ScienceDirect

Applied Thermal Engineering

journal homepage: www.elsevier.com/locate/apthermeng

A hybrid optimization algorithm for the thermal design of radiant paint cure ovens

A. Ashrafizadeh [a,*], R. Mehdipour [b], C. Aghanajafi [a]

[a] Department of Mechanical Engineering, K. N. Toosi University of Technology, No. 17 Pardis St., Mollasadra Ave., Vanak Square, Tehran 1999143344, Iran
[b] Department of Mechanical Engineering, Tafresh University of Technology, Tafresh, Iran

ARTICLE INFO

Article history:
Received 9 October 2010
Accepted 27 January 2012
Available online 2 February 2012

Keywords:
Dynamic optimization
Radiation heat transfer
Paint cure ovens
Hybrid optimization

ABSTRACT

Continuous paint cure ovens have many important industrial applications. In particular, convection ovens are extensively used in auto industries. Radiation paint cure ovens have attractive features as well and attempts have been made to design the oven and the radiation panels such that the moving loads experience desirable, nearly uniform, heating process. Due to the motion of the load and the variation of the radiation exchange factors during the curing process, the solution of this design problem corresponds to the solution of a dynamic optimization problem. This is computationally demanding in a realistic three-dimensional case and the computational cost needs to be minimized. Two-dimensional test problems provide opportunities for algorithm development and quick evaluation. This paper focuses on the convergence acceleration of this thermal optimization algorithm for a 2D test problem. By combining the features of an optimization algorithm with the capabilities of the neural network method, a hybrid design algorithm is obtained which is considerably faster than the original algorithm. It is shown that by employing a neural network trained by a simplified physical model, the computational cost can be reduced close to an order of magnitude without significant loss of accuracy.

1. Introduction

Cure ovens are extensively used in the industry to control the drying process of the coatings and paint layers on various body configurations. For example, the quality of the paint on an automobile body depends on the intensity, duration and mechanisms of heat transfer in a series of cure ovens which comprise the body paint shop. Useful information regarding the physical and chemical aspects of the paint curing processes is provided in a number of publications [1–3].

Continuous paint cure ovens can be generally categorized as convection, radiation and radiation–convection ovens. Traditionally, radiation–convection ovens have been used in auto industries. In this type of ovens radiation panels are used near the entrance of the oven and are responsible mainly for a relatively rapid heat-up process during which the paint layer is heated up and dried. Afterward, nozzles are used to blow in hot air. The convection section of the oven is mainly responsible for the holding period of the curing process. This latter part of the heat treatment process is highly energy consuming. Furthermore, there are safety concerns associated with the curing of chemical-based paint materials via convective heat transfer [4]. The other noticeable technical problem relevant to the convective ovens is the management of the air flow and temperature field such that a nearly uniform curing process occurs on all body parts. For complex geometries it is often necessary to block or change the caps of a number of nozzles to achieve a desirable velocity field around the body. Obviously, the objective is to control the heat transfer rate to the painted body by modifying the convection heat transfer coefficient.

A radiation paint cure oven provides an attractive alternative that overcomes some of the difficulties associated with the classical convection and radiation–convection ovens just described. In terms of the energy consumption, radiation ovens are considerably more efficient and in terms of the pollution and safety measures, they posses obvious preferences. However, in spite of the fact that the radiation panels can be more easily positioned toward hidden parts of the body, convective ovens still provide a better solution for the problem of uniformity of the heating process. Technological advances in radiation ovens may overcome this relative shortcoming in the near future.

Radiation exchange between the heat source panels and a stationary load is discussed in many text books [5,6]. Finite element-based radiation exchange models [7–9] and the classical

* Corresponding author. Tel.: +98 21 84063219; fax: +98 21 88677274.
E-mail addresses: ashrafizadeh@kntu.ac.ir, aliash03@yahoo.com (A. Ashrafizadeh), raminme56@yahoo.com (R. Mehdipour), aghanajafi@kntu.ac.ir (C. Aghanajafi).

1359-4311/$ – see front matter © 2012 Elsevier Ltd. All rights reserved.
doi:10.1016/j.applthermaleng.2012.01.062

Nomenclature

C_0	constant thermo–physical property in Eq. (4)
C_i	total heat capacity of the element i (J/K)
CDOA	classical dynamic optimization algorithm
$E^k_{b,i}$	emissivity power of element i at time level k (W/m^2)
EIT	equivalent isothermal time (s)
f	transfer function
F_{i-j}	radiation shape factor between elements i and j
F_{avg}	objective function
H	heaviside function
$\tilde{\boldsymbol{H}}$	The Hessian Matrix
l	number of neurons in the NN
MDOA	modified dynamic optimization algorithm
n	total number of elements
n_b	number, total number of points on the body
n_h	number of heaters
n_l	number of neural network layers
n_n	number of neurons
n_t	time intervals that the body is placed in the oven
n_x	the number of inputs of the neural network
n_y	the number of outputs of the neural network
N_A	number of active elements
NCP	Nominal Cure Point
NN	Neural Network
P_i	penalty term in Eq. (6)
$\boldsymbol{p}_r$	search direction at iteration number r
$Q_{i,\text{rad}}$	net radiation from the element i (W)
s_i	number of iterations to achieve the minimum point
s_t	number of iterations of matrix solution due to non-linearity
t_c	curing time (s)
T_i(t)	temperature of element i (K)
T_r	nominal curing temperature (K)
TH	temperature history
TTT	target transient temperature
x_{di}	inputs of the NN model
y_{ds}	anticipated outputs of the NN model
y_s	outputs of the NN model
v	number of design variables
w	weight functions

Greek symbols

α_r	step size
ε_i	emissivity of element i
θ_r	vector of design variables at the iteration r
Φ_{target}	NCP
Φ_i	equivalent isothermal time of element i

Subscripts

B	body
E	east boundary
i&j	element number
i,r	element i in iteration number r
H	heater
h	hidden layer of the NN model
N	north boundary
o	output layer of the NN model
r	counter of optimization iteration
S	south boundary
t	time
W	west boundary

Superscripts

k	time level

network method [10] are well-known and widely used radiation analysis techniques. This thermal analysis problem is mathematically modeled as a system of algebraic equations for the unknown panel temperatures or heat fluxes. The coefficient matrix in the algebraic set is dense; meaning that there are no, or very few, zero elements in the matrix.

Radiation exchange analysis in ovens with moving loads is computationally more demanding, because a sequence of fixed load radiation analysis problems needs to be solved [11]. Even for an oven with a non-participating medium, this radiation analysis problem is computationally expensive. Frequent calculation of the radiation view factors between different elements and also frequent inversion of a dense coefficient matrix at different time levels are the major computationally demanding tasks in solving a dynamic radiation exchange problem.

Thermal design of radiation panels in enclosures with stationary loads has also been discussed by many researchers. The formulation and solution of the design problem as an inverse problem is discussed in [7,12,13]. Iterative methods, which solve a sequence of analysis problems, have also been developed in [3].

Compared to the enclosures with stationary loads, there are fewer publications regarding the thermal design of continuous radiation ovens with moving loads. Such design problems are often formulated as dynamic optimization problems [11,14].

Three basic procedures in an optimization loop, i.e. procedures A, B and C, are shown in Fig. 1. The first procedure, i.e. procedure A, is basically a thermal analysis routine in which the field or state variables are updated. In procedure B the objective function is simply evaluated and checked against a convergence criterion. Finally, in procedure C the design variables are updated to ultimately nullify, up to a convergence criterion, the objective function.

In a Classical Dynamic Optimization Algorithm (CDOA) for the design of continuous radiation ovens, the procedure A requires the

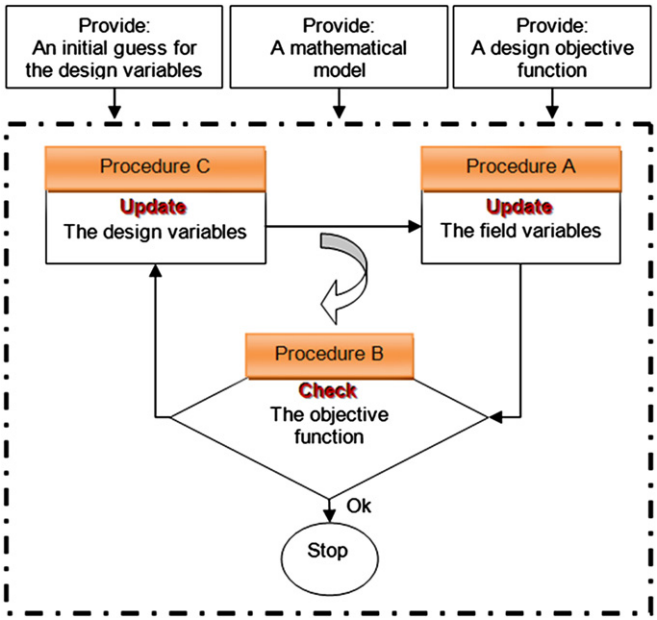

Fig. 1. Three basic procedures in an optimization loop.

solution of a dynamic radiation exchange problem in which during each time step the panel settings, i.e. the design variables, and the load are kept fixed and the surface temperature distribution of the load is calculated. This transient problem is usually one of the most expensive parts of the CDOA in which a radiation exchange problem in an enclosure has to be solved in each time step. Definition and use of the paint cure window, discussed in [15], is now an industry standard method for carrying out the procedure B. Finally, in procedure C the design variables are updated to reduce an objective function norm. Most commonly, the objective function gradient information with respect to the design variables is used to provide the search direction and the step size for the next move in the design space [2]. Procedure C can also be a computationally expensive procedure. To reduce the computational cost and time, improvements can be proposed in all three procedures of a typical optimization loop shown in Fig. 1. Such improvements relevant to the procedures B and C have been proposed by the authors in [2].

In this paper a simplified physical model is used to train a Neural Network (NN) and to achieve a Modified Dynamic Optimization Algorithm (MDOA). The objective of the MDOA is to reduce the computational cost of the procedure A in the design loop. While the energy exchange between all elements, used to define the discrete model of the load and the oven, is taken into the consideration, only the information relevant to a fraction of the elements is employed to train the corresponding neural network. The underlying idea is that only a small number of elements have sufficiently large shape factors so that they actually play important roles in the energy exchange. Therefore, it is reasonable to only use these active elements in the NN training process. Comparative studies reveal that the proposed MDOA is considerably less expensive than the CDOA in terms of the computational cost and time, while the computational results of both methods are sufficiently close to each other.

2. The active elements

Fig. 2 shows a two-dimensional section of an oven used to dry the paint layer on a cylindrical moving object. Full description of the problem and the model assumptions can be found in [2].

The cylindrical body is discretized using 40 elements and 520 elements are used to discretize the walls of the oven. There are 6 elements on each one of the 10 heaters shown in Fig. 2. For each element j on the body ($j = 1, \ldots, 40$) at an arbitrary location, the shape factors F_{ij} ($i = 1, \ldots, 560$) need to be calculated. Fig. 3 shows that for the element e_1, shown in Fig. 2, less than 100 wall elements have non-zero shape factors. If one defines a threshold for the F_{ij} to distinguish between the active elements and the passive ones, i.e. un-important elements from the energy exchange point of view, only a limited number of elements might be considered as active elements. Therefore, for each element j on the body at a given position along the oven, there are N_A active elements. The

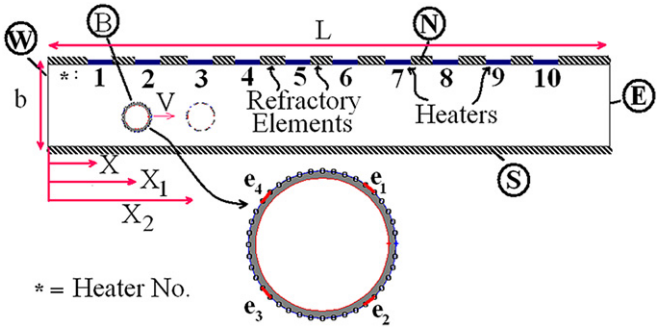

Fig. 2. A two-dimensional radiation oven.

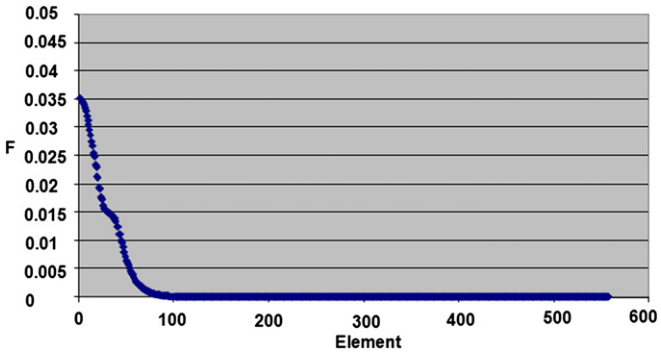

Fig. 3. View factors corresponding to the element e_1 in Fig. 2.

temperatures and view factors of the active elements are used in the training process of the corresponding neural network model as explained later.

3. The mathematical model of the curing process

The energy balance for an arbitrary element can be written as follows [2]:

$$C_i \frac{dT_i(t)}{dt} = Q_{i,g} - Q_{i,\mathrm{rad}} \quad (1)$$

$$Q_{i,\mathrm{rad}} = \sum_{j=1}^{N} \left[E_{b,i}A_i - Q_{i,\mathrm{rad}} \frac{1-\varepsilon_i}{\varepsilon_i} - E_{b,j}A_i + Q_{j,\mathrm{rad}} \frac{(1-\varepsilon_j)A_i}{\varepsilon_j A_j} \right] F_{ij} \quad (2)$$

where C_i is the total heat capacity of the element i (J/K), $T_i(t)$ is the temperature of the element i at time t, $Q_{i,g}$ is the net heat transfer rate into the element from an external thermal reservoir and $Q_{i,\mathrm{rad}}$ is the net radiation from the element into the enclosure. Parameters ε_i and A_i are emissivity and area of the element i respectively.

As fully explained in [2], energy balances for all elements result in a set of non-linear, coupled first order differential equations. By linearizing the equations and using an explicit time marching approach, a set of linear algebraic equations is obtained that can be solved for the elements' unknowns. This means that at the time step t_k all view factors need to be calculated and a full matrix needs to be inverted to obtain the temperatures or heat fluxes of the elements at t_k.

4. The optimization algorithm

Fig. 4 shows the classical optimization algorithm which employs the mathematical model, just described, to update the temperature field and the objective function [2]. In this algorithm a gradient-based optimization method is used in which the information regarding the gradient of the objective function is needed to calculate the search direction ($\vec{P}_r$) and the step size (α_r) at the rth design iteration. The objective function, F_{avg}, is calculated based on the definition of the equivalent isothermal time [2]. As shown in Fig. 5, a paint cure window and a Nominal Cure Point (NCP) are defined and the objective is to make sure that the curing times, i.e. the Φ values corresponding to different elements on the load shown in Fig. 5, are as close as possible to the NCP. Further information can be obtained from [2]. Looking at the design flow chart in Fig. 4, it is clear that the most expensive parts of the loop are the calculation of the unknowns (T_i^k and $Q_{i,\mathrm{rad}}^k$) and the sensitivity coefficients ($\partial E_{b,i}/\partial\theta_{i,r}$ and $\partial Q_{j,\mathrm{rad}}/\partial\theta_{i,r}$). Instead of using the actual mathematical model of the problem, described in the previous section, a neural network is trained to provide a non-linear function as a substitute for the mathematical model. To reduce the number of the inputs, only a number of

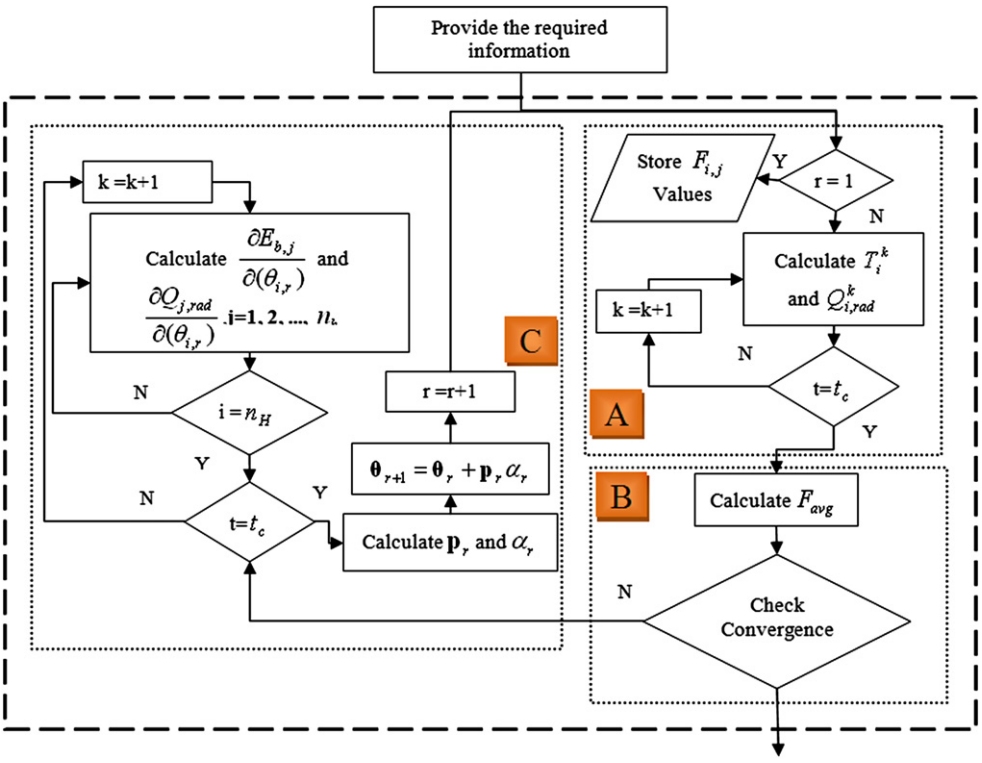

Fig. 4. The classical thermal optimization algorithm (CTOA).

selected active elements are used in the NN training process. Assuming that a trained NN is available and used for the estimation of the energy exchange in the oven, the computational time is drastically reduced. The training of the NN for the radiation oven design problem is discussed in the following section.

5. The Neural Network model

A single layer NN model, shown in Fig. 6, is used in this study. Using the back propagation method [16], the weight functions, i.e. w_{ji}^h and w_{kj}^o, are determined during the training process. The

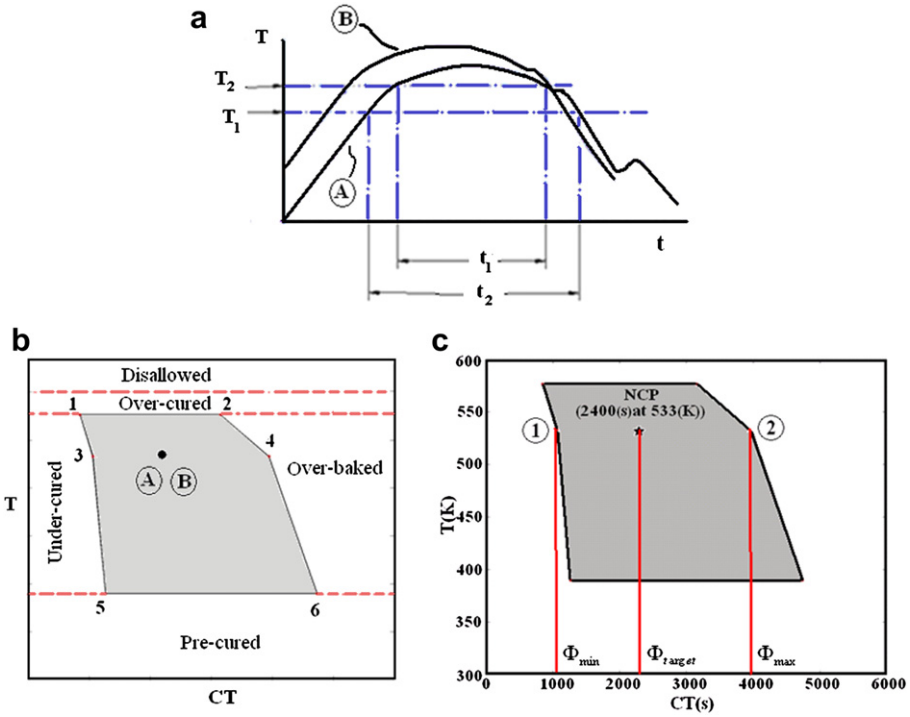

Fig. 5. Temperature history (TH) curves for arbitrary body points A and B (a), corresponding EITs (b), and A user-specified NCP in a factory-provided paint cure window (c).

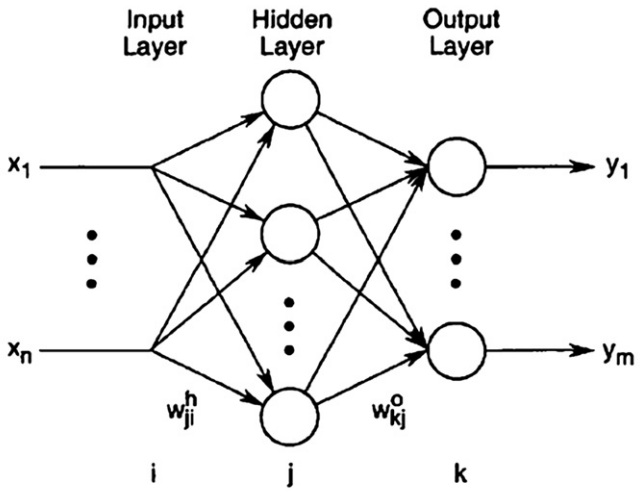

Fig. 6. The single layer nueral network used in this study ([16]).

weight functions are obtained so that the function $E(w)$ is minimized:

$$E(w) = \frac{1}{2}\sum_{s=1}^{m}(y_{ds} - y_s)^2 \tag{3}$$

In Eq. (3),y_{ds} represents the available data for the training and y_s is the following non-linear equation:

$$y_s = f_s^o\left[\sum_{j=1}^{l} w_{sj}^o f_j^h\left(\sum_{i=1}^{n} w_{ji}^h x_{di}\right)^2\right] \tag{4}$$

Parameter l represents the number of neurons, i.e. the number of nodes in the hidden layer, f_s^o and f_j^h are the transfer functions and x_{di} and y_s are the inputs and outputs of the NN model respectively. Other relevant information regarding the training of the NN model for the element j is as follows:

5.1. Inputs

(1) Emissive powers of 16 wall elements hotter than the jth element on the body, i.e. $E_{j,i} = F_{j-i}(\sigma T_j^4 - \sigma T_i^4)$, (16 parameters).
(2) Emissive powers of 8 wall elements colder than the jth element on the body, i.e. $E_{j,i} = F_{j-i}(\sigma T_j^4 - \sigma T_i^4)$, (8 parameters).
(3) Temperature of the jth element at the previous time step (1 parameter).

5.2. Output

(1) Temperature increment for the element j (1 parameter).

5.3. Transfer functions

(1) Heaviside step function is used for all transfer functions.

6. Training of the NN model

Training of the neural network is accomplished by employing the results of the thermal analysis of the moving load in a test oven via the finite element analysis method. The results of the thermal analysis are organized in a data matrix with 16,000 rows. Each row in the data matrix contains 26 parameters, i.e. the inputs and outputs used for the training of the element at a particular time. Among the data, 500 rows are randomly selected for the training. The weight functions are calculated so that the NN model generates the selected data with an average deviation of about 0.1%. The outcome of this procedure is a model capable of estimating the temperature variation of each user-specified element in the test oven. The applicability of the model was evaluated through the following three steps:

(1) The model was used to re-generate the original 16,000 set of data. The maximum deviation from the original data was about 5%.
(2) The model was then used to predict the temperature variation of an arbitrary element in an oven with arbitrary temperature settings (setting number 2). To examine the validity of the NN model, the temperature histories of two different elements, shown in Fig. 7, were calculated using both the NN model and the radiation analysis via the finite element method for the setting number 2. The model had the maximum deviation of about 9% in this case. Note that in this test case the NN model, trained for an element in the initially used oven (setting number 1), is employed in spite of the fact that the temperature settings and the orientation of the element have both been changed as compared to the initial model.
(3) Finally, the trained NN model was used for a different body shape; i.e. a rectangular cylinder. The maximum deviation of the model for an element on this geometry was about 16%. The panel settings were chosen similar to the case used for the training of the NN model.

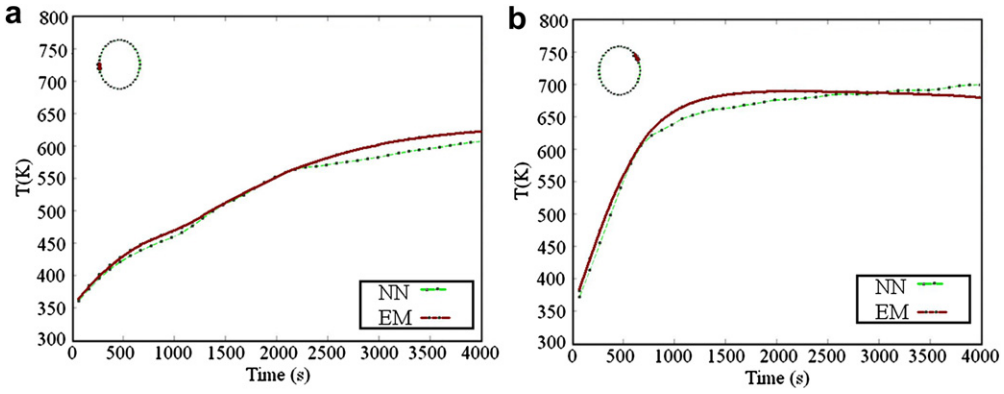

Fig. 7. Comparison between the results obtained from the NN model and the finite element method in an oven test case for two different elements.

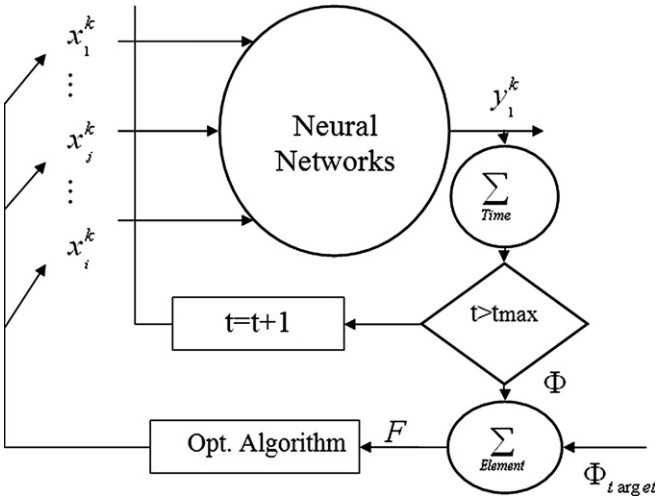

Fig. 8. The neural network model employed in the design procedure.

Now that an estimation of the errors associated with the NN model in different applications is available, it would be interesting to compare the computational times of the element-based radiation exchange analysis method and the neural network approach. The design example will be presented following some explanations regarding the optimization method.

7. The optimization algorithm

As mentioned before the optimization algorithm consists of three main procedures. We have already discussed the application of the NN model in the procedure A. Fig. 8 shows the data transfer process between the NN model and the optimization algorithm. Some details regarding the procedures B and C, as implemented in this study, are discussed in this section. Further details can be obtained in [2].

To define the objective function and to check the optimality condition, i.e. the Procedure B, the paint cure window and the Equivalent Isothermal Time (EIT) are used as suggested by [17]. The EIT at the reference temperature T_r for an arbitrary point i on the body, also shown by Φ_i, is defined as [17]:

$$\Phi_i = \mathrm{EIT}_i|_{\mathrm{Tr}} = \int_0^{\mathrm{tc}} \exp\left[C_0\left(\frac{T_i(t)-T_r}{T_i(t)T_r}\right)\right]\Delta t \tag{5}$$

In Eq. (5), C_0 is a constant thermo–physical property related to the paint material. Using the definition of the EIT, the objective function can now be defined as follows:

$$F_{\mathrm{avg}}(\theta) = \frac{1}{n_B}\sum_{i=1}^{n_B} a_1\left[\Phi_i(\theta)-\Phi_{\mathrm{target}}\right]^2 + P_i(\theta) \tag{6}$$

The last term in Eq. (6), $P_i(\theta)$, is a penalty term defined as follows:

$$\begin{aligned} P_i(\theta) \equiv\ & a_2[\Phi_{\max}-\Phi_i(\theta)]^2 H[\Phi_{\max}-\Phi_i(\theta)] \\ & + a_3[\Phi_i(\theta)-\Phi_{\min}]^2 H[\Phi_i(\theta)-\Phi_{\min}] \end{aligned} \tag{7}$$

In Eq. (7), H(•) stands for the Heaviside step function. Fig. 5c shows the cure window, the user-specified Φ_{target} as well as $\Phi_{\min}$ and $\Phi_{\max}$.

In the remaining part of the optimization loop a procedure is needed to update the design variables. Here a gradient-based optimization method is used to find the optimum values of design variables, θ^*, that minimizes the objective function $F_{\mathrm{avg}}(\theta)$. A correction vector c_r is needed to update the vector of design variables in the rth design iteration. There are many methods available to specify the search direction, $\boldsymbol{p_r}$, and the step size, α_r, in the following formula:

$$\theta_{r+1} = \theta_r + c_r = \theta_r + \alpha_r \boldsymbol{p}_r \tag{8}$$

The step size is set equal to a non-vanishing power series in this study [12]:

$$\alpha_r = \frac{\alpha_0}{r^{1.2}} \tag{9}$$

Gradient-based optimization methods use the objective function gradient with respect to the design variables, i.e. $g(\theta)$, to calculate the search direction vector $\boldsymbol{p}_r$. The steepest descent method simply takes $\boldsymbol{p}_r = -g(\theta_r)$. A better search direction is usually obtained using higher order derivatives, thereby reducing the required minimization steps, but the computational cost of calculating these terms could be prohibitive. An alternative approach is the quasi-Newton method, in which the second-order curvature information contained in the Hessian matrix is estimated based on how the gradient vector changes between successive iterations. For example, the BFGS[1]-implementation of the quasi-Newton method, used in the present study [16], calculates the search direction according to the following formula:

$$\boldsymbol{p}_r = -\tilde{\boldsymbol{H}}_r^{-1} g_r \tag{10}$$

where the Hessian is approximated by

$$\begin{aligned} \tilde{\boldsymbol{H}}_r &= \tilde{\boldsymbol{H}}_{r-1} + \tilde{M}_{r-1} + \tilde{N}_{r-1},\ r = 1,2,\ldots \\ \tilde{\boldsymbol{H}}_0 &= I \\ \tilde{M}_{r-1} &= \left(\frac{1 + y_{r-1}^T \tilde{\boldsymbol{H}}_{r-1} y_{r-1}}{y_{r-1}^T P_{r-1}}\right)\frac{p_{r-1}p_{r-1}^T}{p_{r-1}^T y_{r-1}} \\ \tilde{N}_{r-1} &= -\frac{p_{r-1}y_{r-1}^T \tilde{\boldsymbol{H}}_{r-1} + \tilde{\boldsymbol{H}}_{r-1} y_{r-1} p_{r-1}^T}{y_{r-1}^T p_{r-1}} \\ y_{r-1} &= g_r - g_{r-1} \end{aligned} \tag{11}$$

The elements of the gradient vector are the objective function sensitivities with respect to the design variables, i.e. the panel temperatures:

$$\begin{aligned} g_i(\theta_r) &= \left[\frac{\partial F(\Phi_i(\theta_r))}{\partial \theta_{1,r}} \frac{\partial F(\Phi_i(\theta_r))}{\partial \theta_{2,r}} \cdots \frac{\partial F(\Phi_i(\theta_r))}{\partial \theta_{v,r}}\right]^T \\ &= \left[g_{i1}(\theta_r) \quad g_{i2}(\theta_r) \cdots g_{iv}(\theta_r)\right]^T \end{aligned} \tag{12}$$

The sensitivity elements are found by differentiating the objective function:

$$\begin{aligned} g_{i1}(\theta_r) =& \frac{\partial F(\Phi_i(\theta_r))}{\partial \theta_{1,r}} = 2a_1(\Phi_i - \Phi_{target})\ \frac{\partial \Phi_i}{\partial \theta_{1,r}} \\ & + 2a_2 H(\Phi_1 - \Phi_i)\ (\Phi_i - \Phi_1)\ \frac{\partial \Phi_i}{\partial \theta_{1,r}} \\ & + a_2 \delta(\Phi_1 - \Phi_i)\ (\Phi_i - \Phi_1)^2 + 2a_3 H(\Phi_2 - \Phi_i) \\ & \times (\Phi_i - \Phi_2)\ \frac{\partial \Phi_i}{\partial \theta_{1,r}} + a_3 \delta(\Phi_2 - \Phi_i)(\Phi_i - \Phi_2)^2 \end{aligned} \tag{13}$$

[1] Broyden-Fletcher-Goldfarb-Shanno.

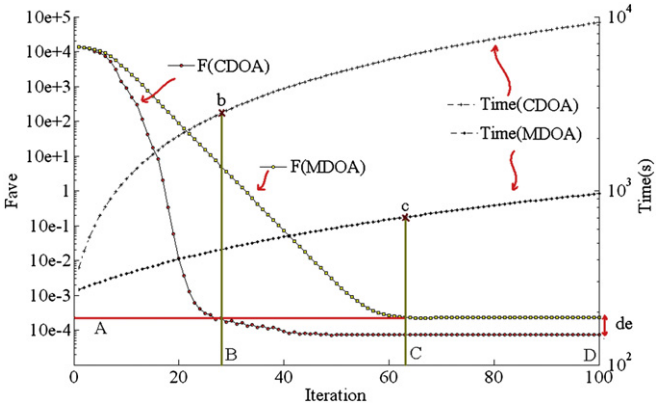

Fig. 9. Convergence of the optimization loop; comparison between the CDOA and MDOA.

Table 1
Comparison between MDOA and CDOA in terms of the computational time and accuracy for the test case shown in Fig. 2.

MDOA		CDOA			
Time(s)	F(ave)	Time(s)	F(ave)	Iteration	Point
460	5.157038	2790	0.00022	28	B
719	0.000225	6120	7.39E-05	65	C
964	0.000213	9270	7.39E-05	100	D

where δ is the Dirac delta function. The derivative terms at the right hand side of Eq. (13) are calculated as follows:

$$\frac{\partial \Phi_i}{\partial \theta_{1,r}} = \int_0^{tc} \left[\exp\left(C_0 \left(\frac{T_i(t) - T_r}{T_i(t) T_r} \right) \frac{C_0}{T_i^2} \left(\frac{\partial T_i(t)}{\partial (\theta_{l,r})} \right) \right) \right] dt \qquad (14)$$

Note that both $T_i(t)$ and $\partial T_i(t)/\partial \theta_{1,r}$ need to be calculated.

Similar sets of equations are solved to obtain $g_{i2}(\theta_r)$, $g_{i2}(\theta_r)$,..., $g_{iv}(\theta_r)$ in Eq. (12). After calculating the components as just described, $g(\theta)$ is obtained and used to update the design variables.

8. A design example

Fig. 9 shows the convergence histories for the CDOA and MDOA used to solve the thermal design problem in an oven similar to the one discussed in Section 2. The panel setting number 2 is used as the initial guess and the element-based NN model trained with the panel setting number 1 is employed in the MDOA calculations. Computational times for the two methods are also shown in Fig. 9.

Table 1 summarizes some of the relevant information regarding the results shown in Fig. 9 as well. It is seen that 28 iterations of the CDOA (point B in Fig. 9) corresponds to 65 iterations of the MDOA (point C in Fig. 9) to meet the convergence criterion shown by the solid horizontal line in Fig. 9. The corresponding times for the two methods are 2790 s for the CDOA (point b) versus 719 s for the MDOA (point c). Note that even though the number of iterations of the MDOA is higher than the CDOA, the computational time for the MDOA is considerably less than the CDOA. This is expected knowing that the execution time per iteration for the NN model is much smaller than the corresponding time for the full radiation analysis. A detailed analysis has shown that when all of the 40 elements around the circular cylinder are taken into the consideration, the finite element method needs about 90 s for a typical intermediate design iteration while this task takes only 7 s if the NN model is employed. This is close to one order of magnitude gain in the convergence rate. However, it is important to note that the NN model also allows the user to consider just a few selected elements. For example if 4 elements are used to evaluate the objective function, the NN model just needs about 1 s for each design iteration. This means that the actual gain in the computational speed can even be close to two orders of magnitude for this design example.

To take care of both computational efficiency and accuracy it is wise to use the NN model before the final iterations and switch to the full thermal analysis mode when the objective function is sufficiently small. This is the hybrid optimization method proposed in this paper. Fig. 10 shows the initial and final, i.e. optimum, temperatures of the 40 elements around the circular cylinder shown in Fig. 2. Note that the optimizer changes the temperatures of all elements considerably to comply with the given paint cure window and the NCP.

9. Remarks regarding the application of the NN model

There is a fixed initial computational cost associated with the training of the NN model which has not been discussed so far. As a general guideline, application of the NN model is advisable whenever a computationally expensive field solver is frequently called. Another noticeable application of NN models is in design problems in which the geometry and/or boundary conditions associated with a given physical phenomenon are changed during the design considerations. For example the proposed NN model in this study can be used in the radiation oven design problems regardless of the panel temperatures and the load geometry. However, in such cases the accuracy of the model needs to be considered to make sure that the computational results are practically acceptable. A remedy to the conflict between the computational speed and accuracy, suggested in this study, is to use the NN

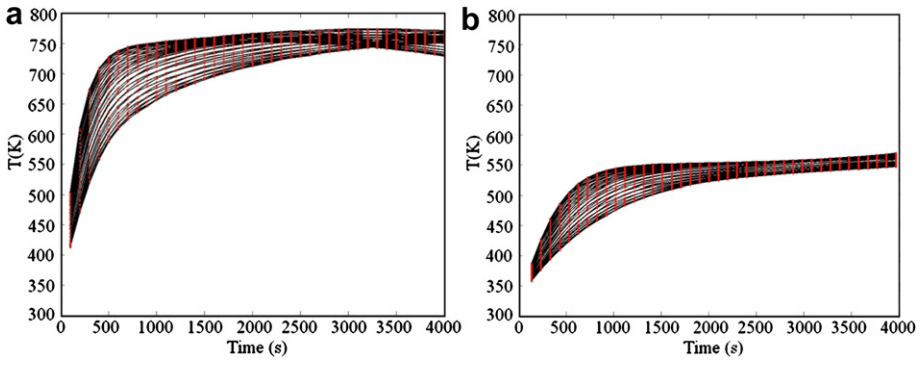

Fig. 10. Temperature history curves, (a) the first iteration, (b) the converged solution.

model in the intermediate iterations and switch to the actual mathematical model afterwards. It should also be reminded that experimentally obtained data can be used in NN models as well. This latter flexibility of the method is particularly important when the physical phenomena associated with the problem are very complex.

Finally, it is worthwhile to do an order of magnitude analysis to compare the computational costs associated with the element method, proposed in [2] and here called the CDOA, and the oven design method assisted with the NN model (the MDOA).

Further studies, not discussed here, show that the computational cost associated with the CDOA is proportional to the following quantity:

$$o\Big(s_i n_t\Big(s_t\Big((2n)^3+(2n)^2\Big)+n^2\Big)+s_i n_t n_h\Big(\Big((2n)^3+(2n)^2\Big)+s_i n_h n_t n_b\Big)\Big) \tag{15}$$

where, n_h, n, s_t, n_t, n_b and s_i represent the number of heaters, total number of elements, number of non-linear iterations for a matrix solution, number of time intervals, total number of elements on the body and the number of iterations required to obtain the optimum solution respectively. Considering the actual values of these numbers, Eq. (15) is roughly proportional to:

$$o(s_t+n_h)s_i n_t n^3 \tag{16}$$

In comparison, a similar analysis for the MDOA reveals that the computational cost in this case is proportional to the following quantities depending on the number of elements:

$$\begin{array}{l} o(n^2 n_t n_b) \text{ for a fine mesh} \\ o(n n_t n_b n_h s_i) \text{ for a coarse mesh} \end{array} \tag{17}$$

Comparing Eq. (17) with Eq. (16), reveals that the computational expense reduces about one or two order of magnitudes depending on the mesh coarseness when the MDOA is employed.

10. Conclusion

A hybrid optimization algorithm for the design of a radiation paint cure oven with a moving load was proposed in this paper. Instead of using a heat transfer analysis code in a gradient-based optimization algorithm, a trained neural network was used to update the temperature field and the objective function. Furthermore, instead of the real physical model, a simplified model was employed in the neural network training process. In the simplified model only the energy exchange of a fraction of the elements is taken into the consideration. A design example was presented which showed that the computational cost could be reduced up to two orders of magnitude, without sacrificing the accuracy in an unacceptable manner, when the proposed neural network model was employed.

References

[1] A. Ashrafizadeh, R. Mehdipour, M. Rezvani, Numerical Simulation of Convective Heat Transfer in a Continuous Paint Cure Oven, CANCAM, Halifax, Canada, 2009, pp. 25–26.
[2] R. Mehdipour, A. Ashrafizadeh, K.J. Daun, C. Aghanajafi, Dynamic optimization of a radiation paint cure oven using the nominal cure point criterion, Drying Technology 28 (2010) 1405–1415.
[3] J. Xiao, J. Li, Q. Xu, Y. Huang, H.H. Lou, ACS-based dynamic optimization for curing of polymeric coating, Journal of Wiley Inter Science 52 (4) (2006) 1410–1422.
[4] C. Geipel, P. Stephan, Experimental investigation of the drying process of automotive base paints, Journal of Applied Thermal Engineering 25 (2005) 2578–2590.
[5] R. Siegel, J. Howell, Thermal Radiation Heat Transfer, Taylor & Francis, New York, 2002.
[6] F.M. Modest, Radiative Heat Transfer, Academic Press, New York, 2003.
[7] K.J. Daun, J.R. Howell, D.P. Morton, Geometric optimization of radiative enclosures through nonlinear programming, Journal of Numerical Heat Transfer 43 (2003) 203–219.
[8] K.J. Daun, J.R. Howell, D.P. Morton, Design of radiant enclosures using inverse and non-linear, Journal of Inverse Problems in Engineering 11 (6) (2003) 541–560.
[9] H.R.F. Francis, J.R. Howell, Transient inverse design of radiative enclosures for thermal processing of materials, Inverse Problems, Design and Optimization Symposium Rio de Janeiro, Brazil, 2004.
[10] J. Zueco, A. Campo, Network model for the numerical simulation of transient radiation transfer process between the thick walls of enclosures, Journal of Applied Thermal Engineering 26 (2006) 673–679.
[11] A.G. Federov, K.H. Lee, R. Viskanta, Inverse optimal design of the radiant heating in materials processing and manufacturing, Journal of Material Engineering and Performance 7 (1998) 719–726.
[12] K.J. Daun, H. Erturk, J.R. Howell, Inverse design methods for high-temperature systems, Arabian Journal of Science Engineering 27 (2C) (2002) 3–49.
[13] K.J. Daun, F. França, M. Larsen, G. Leduc, J.R. Howell, Comparison of methods for inverse design of radiant enclosures, Journal of Heat Transfer 128 (2006) 269–282.
[14] H. Erturk, O.A. Ezekoye, J.R. Howell, Boundary condition design to heat a moving object at uniform transient temperature using inverse formulation, Journal of Manufacturing Science and Engineering 126 (2004) 619–626.
[15] E.A. Turi, Thermal Characterization of Polymeric Materials, Academic Press, New York, 1997.
[16] E.K.P. Chong, S.H. Zak, An Introduction to Optimization, John Wiley & Sons, New York, 2001.
[17] J. Xiao, J. Li, H.H. Lou, Y. Huang, Cure-window-based proactive quality control in topcoat curing, Journal of Industrial and Engineering Chemistry 46 (2006) 2351–2360.