OpenFOAM® Journal

Volume 1 [Full Papers], Pages 27–46
ISSN: 2753-8168

SEQUENTIAL INVERSE HEAT CONDUCTION PROBLEM IN OPENFOAM

J. BOHACEK[1,*], J. KOMINEK[1], A. VAKHRUSHEV[2], E. KARIMI-SIBAKI[2], AND T.-W. LEE[3]

[1]Heat Transfer and Fluid Flow Laboratory, Faculty of Mechanical Engineering, Brno University of Technology, Technicka 2896/2, 616 69 Brno, Czech Republic.
Email address: jan.bohacek@vut.cz

[2]Christian-Doppler Laboratory for Metallurgical Applications of Magnetohydrodynamics, Dept. of Metallurgy, Montanuniversitaet Leoben, Franz-Joseph Strasse 18, 8700 Leoben, Austria.

[3]Mechanical and Aerospace Engineering, SEMTE Arizona State University, Tempe, Arizona 85287, USA.

DOI: 10.51560/ofj.v1.33
Version(s): OpenFOAM® v7
Repo: –

Abstract. The solution of the inverse heat conduction problem (IHCP) is commonly found with the sequential algorithm known as the function specification method with explicit updating formulas and sensitivity coefficients of heat flux. This paper presents a different approach namely a direct mathematical optimization of minimizing the least squares norm between experimental data and simulation. A CFD open-source code OpenFOAM is used together with NLOPT and DLIB optimization libraries. To guarantee credibility of the simulation tool developed herein, real experimental data is used from spray cooling of a fast-moving hot steel plate. As the IHCP is inherently an ill-posed problem, the proposed sequential algorithm is stabilized using future time stepping and thereof the optimal number is explained. An assumption about the profile of thermal boundary condition during future steps must be made. It is shown that assuming a linear change of the heat transfer coefficient during each sequence of future time steps yields more accurate results than setting a constant value. For the problem size considered with less than 10k cells, the preconditioned conjugate gradient (FDIC) linear solver converges faster than the multigrid solver (GAMG). However, the latter performs better as the accuracy is concerned. Concerning the best choice of minimizer, the BOBYQA algorithm (quadratic approximation) is found superior to other methods. The proposed IHCP solver is compared with the well-established one.

1. Introduction

The inverse heat conduction problem (IHCP) comprises measurements of the internal or surface temperature of a heat conduction system and a subsequent numerical determination of the unknown thermal boundary conditions [1], initial temperature distribution [2], thermo-physical properties [3], boundary shape [4], internal heat sources [5] or the thermal contact conductance (TCC). Nowadays, the IHCP is useful in many engineering applications such as metallurgy [6], material processing [7], non-destructive testing [8], geometry optimization [9], nuclear physics [10], power engineering [11], manufacturing engineering [12], chemical engineering [13], nanotechnology [14], bioengineering [15], aerospace engineering [16], etc. The IHCP methodology frequently remains interchangeable between different application fields regardless of the unknown parameters. The IHCP technique is often the only, semi-experimental tool to recover the heat conduction system parameters.

The present paper focuses on the reconstruction of thermal boundary conditions during the spray cooling or quenching [17][18] during various metallurgical processes such as the hot rolling [19], the continuous casting of steel [20], etc. The direct measurement of the surface heat flux and surface temperatures is hindered by an opaque ambience during the industrial process due to the dust clouds and water mist as well as by high surface temperatures and velocities. The contact measurement devices cannot be used as the heat transfer would be significantly influenced. Therefore, solving the IHCP is the only available option [21].

* Corresponding author
Received: 28 January 2021, Accepted: 16 August 2021, Published: 21 October 2021

Various strategies exist to solve the IHCP [22]. An overview of solution techniques can be found in [23]. Exact solutions are presented in the literature mostly for the steady state and very rarely for the transient inverse problems exclusively with simple geometries [24]. The Laplace transform is effective for the 2D inverse problems [25], however, is inapplicable for the practical cases due to the severe limitations such as a semi-infinite domain dimension, isotropic and temperature independent thermal properties. On the other hand, these methods represent an excellent fast tool to verify numerical computational methods.

The full-domain (or the global time-space) inverse methods combine both the space and the time dimension. Application of the corresponding discretization technique typically reduces the problem to a single system of algebraic equations. In the well-known Trefftz method [26] the IHCP solution is approximated by a linear combination of functions, so-called T-functions, satisfying the governing equations. The coefficients of the objective functional are found with the least squares method (LSM) [27]. Regularization methods are necessary to cope with the ill-posedness of the system, such as singular value decomposition [28] or Tikhonov regularization method [29][30]. Finite Element Methods with T-functions (FEMT) are sustainable to very noisy measurement data. Energetic regularizations can be employed to improve the accuracy [31]. Generally, unlike in the classical FEM, less elements (or subdomains) are required to yield a very accurate solution since the T-functions as the shape functions perfectly fit the governing equation system. The FEMT still can be used for the problems with the temperature dependent properties [32] by performing the Kirchhoff transform [33]. However, to resolve the strong dependence of the properties and the steep temperature gradients the number of the elements dramatically grows.

The sequential (time-marching) inverse methods are alternative to the full-domain methods. The sequential methods can cope with the temperature dependent properties, composites solids, combined mechanisms of heat transfer, moving boundaries, etc. The function specification method is perhaps the most popular method to determine the surface heat flux based on the internal temperature history of the solid bodthe authory [34]. The temperature field is expanded in a Taylor series around arbitrary but known values of the heat flux; the first order derivatives are only kept referred to as sensitivity coefficients [35]. Then, the unknown heat flux is expressed by an explicit formula. Typically, the summation over several forward time steps is performed to smear the measured noise to ensure the convergence. Suitable for linear systems, the method can be also applied to the non-linear IHCP; however, iterative procedure is necessary [36][37]. The method has been successfully used with the multidimensional problems [38][39][40]. A similar sequential method is available known as the conjugate gradient method with the adjoint equation (CGMAE). Unlike the function specification method, the thermal boundary condition is not prescribed by any functional form and it is particularly useful for non-linear, multidimensional and online problems [41][42][43].

In the present paper the thermal boundary condition is directly determined according to the sequential inverse method by using an efficient optimization technique. The optimization methods are commonly classified into the stochastic, deterministic, and hybrid methods. The genetic algorithm (GA) and the Particle Swarm Optimization (PSO) are the stochastic methods [44] with the ability to find the global extremum of non-convex objective functions. Nevertheless, the local deterministic methods are often sufficient for solving the IHCP and provide significantly higher convergence rates. The least-squares method [45], the Levenberg-Marquardt method [46], the Nelder-Mead method [47][48] and the subplex [49] can be often encountered in the research papers about the IHCP.

This paper presents an original IHCP solver developed in the open-source code OpenFOAM® [50] and is organized into ten separate sections. Introduction 1 is followed by the physical background and the solver description in section 2. The model setup, the finite volume mesh and the input experimental data for the IHCP tests are presented in section 3. The subsequent sections are aimed at choosing the optimal number of forward time steps to ensure a stable solution (section 4), selecting the appropriate profile of the thermal boundary condition during the forward time marching (section 5), selecting the most efficient minimizer (section 6), comparing performance of linear solvers (section 7), comparing the proposed IHCP solver with the in-house alternative (section 8), conducting the grid-size sensitivity analysis (section 9), and finally summarizing the main findings in Conclusions 10.

2. Algorithm for Inverse Heat Conduction Problem (IHCP)

A simulated solid body is specified in the space domain Ω enclosed by surfaces Γ. The time-dependent temperature T_P is defined from the experiment at the tip of the thermocouple (point P). The thermophysical properties such as the thermal conductivity k, specific heat c_p and density ρ are constant and isotropic. The thermal boundary conditions are known at all surfaces but the surface Γ_1. The IHCP should be solved to recover the heat flux condition q_t at Γ_1. In this work, a constant profile of q_t along Γ_1 is assumed. Thus, a measurement from a single point is sufficient. The layout is shown in Fig. 1.

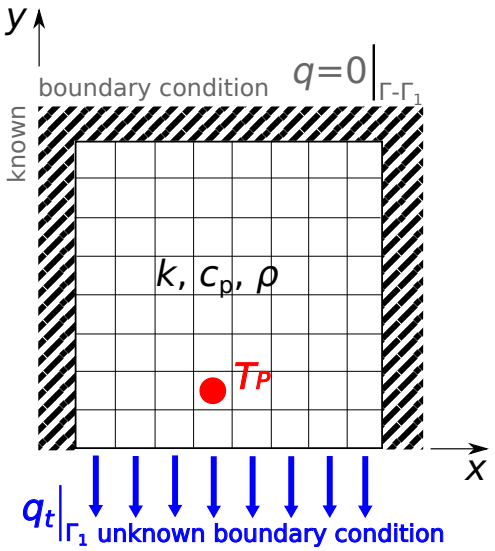

FIGURE 1. The geometrical layout of the 2D heat conduction system with the unknown heat flux q_t at Γ_1, all other surfaces being insulated, and the temperature T_P known at the point P (the red circle).

The unsteady heat conduction equation [51], the boundary and initial conditions can be written as:

$$\frac{\partial}{\partial t}(\rho c_p T) - \nabla\cdot(k\nabla T) = 0 \quad (\in \Omega, t > t_0) \tag{1}$$

$$-k\frac{\partial T}{\partial n} = 0 \quad (\in \Gamma - \Gamma_1, t > t_0)$$

$$-k\frac{\partial T}{\partial n} = q_t \quad (\in \Gamma_1, t > t_0) \tag{2}$$

$$T = T_0 \quad (\in \Omega, t = t_0) \tag{3}$$

The cooling curve is measured at the point P in the experiment:

$$T = T_{Pt} \quad (\in \Omega, t > t_0) \tag{4}$$

The unknown heat flux q_t is considered in the convective boundary condition form defined as:

$$q_t = HTC_t(T_{St} - T_{\infty t}), \tag{5}$$

in which HTC_t, T_{St} and $T_{\infty t}$ denote respectively the heat transfer coefficient, the surface temperature of Γ_1 and the ambient temperature. In order to find $q_t \in \Omega, t > t_0$, the difference between the experimental temperature (4) and the simulated temperature T' calculated with (1) must be minimized. The minimization problem can be formulated as:

$$\forall t \text{ find } q_t \text{ so that } F = \min(T_{Pt} - T'_t)^2, \tag{6}$$

with q_t and F denoting the parameter and the objective function, respectively. Typically, the temperature T_{Pt} in (4) is known in discrete times t incremented by Δt; it is also subjected to the data acquisition errors. The thermal boundary condition (2) is extremely sensitive to the small random variations in measurements. Furthermore, the existence of the solution to a physical problem can be proven only for a few cases [52]. Therefore, the IHCP is an ill-posed problem requiring a stabilization by expanding (6) by the additional forward marching in time. The minimization becomes:

$$\forall t \text{ find } q_t \text{ (or } HTC_t) \text{ so that } F = \min \sum_{i=0}^{N-1} (T_{Pt+i} - T'_{t+i})^2, \tag{7}$$

The sequential algorithm for the IHCP consists of solving (1)-(7). Equation (1) is solved using the laplacianFoam [53] in the finite volume framework of OpenFOAM® by reformulating it into:

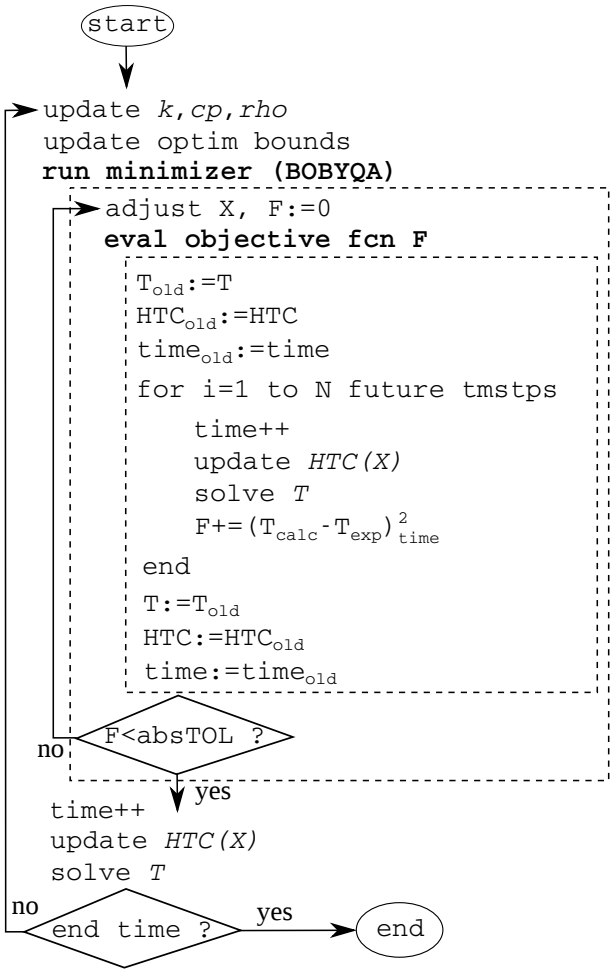

FIGURE 2. The sequential algorithm of the IHCP as was used in the present study.

$$\frac{\partial}{\partial t}(T) - \nabla\bullet(\alpha\nabla T) = 0 \quad (\in \Omega, t > t_0), \tag{8}$$

with the thermal diffusivity $\alpha = k/(\rho c_p)$.The minimization task (7) is solved by the corresponding methods from the open-source libraries NLopt [54] and DLIB [55]. The NLopt library includes the efficient local derivative-free bound-constrained minimizers. The DLIB library employs the local BFGS quasi-Newton method with the built-in finite difference derivative computation, which is not available in the Nlopt. Denoting the optimization parameter HTC_t as X, the individual steps of the IHCP algorithm are shown in Fig. 2.

The thermal properties are updated in each time step using the linear interpolation and lookup tables with the temperature dependent data. The lower and upper bounds are set for the parameter X to accelerate the convergence of (7). Next, the optimization of the objective function is performed employing the solution of (8) for the N forward time continuous updates of the heat transfer coefficient HTC_t. The absolute tolerance $absXTol$ of the parameter X defines the convergence criterion to proceed to the next time step.

3. Model setup with experimental data

The temperature history (4) was measured by a thermocouple during cooling of a 20 mm thick steel plate. The plate was initially preheated up to 1000 °C. The surface $\Gamma - \Gamma_1$ was insulated and the surface Γ_1 was cooled by a spraying nozzle. The K-type thermocouple was installed 1 mm below the cooled surface, the data was collected by a datalogger at 320 Hz. During 8 minutes of the experiment $ns = 155$k samples were collected. The cooling curve (4) is shown in Fig. 3. The periodically repeating temperature drops with subsequent recalescences are caused by the movement of the steel plate under the spraying nozzle. The relative motion between the steel and the spraying water has an important effect on the

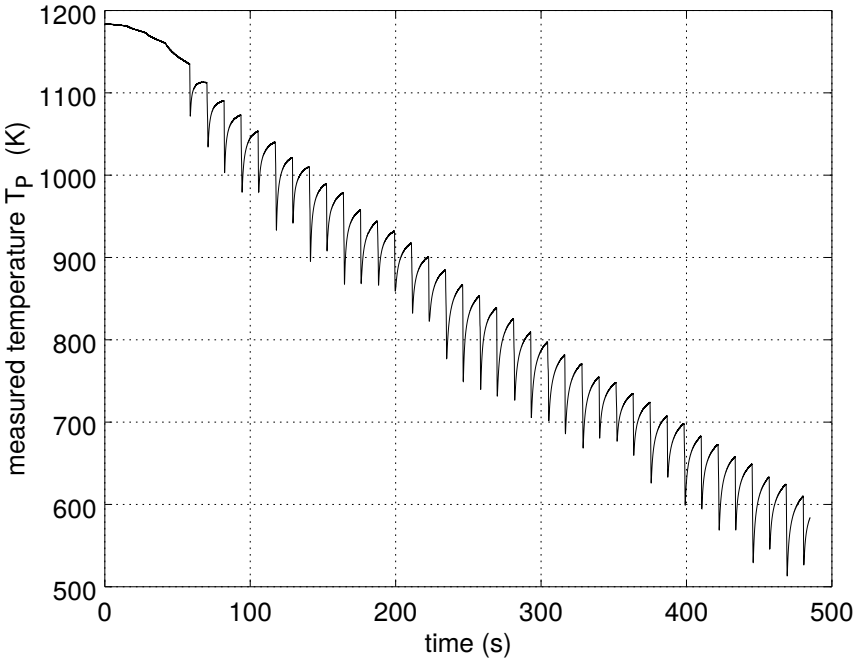

FIGURE 3. The cooling curve obtained from the K-type thermocouple at the point P, obtained from the experiment with the nozzle spraying onto the steel plate moving back and forth underneath.

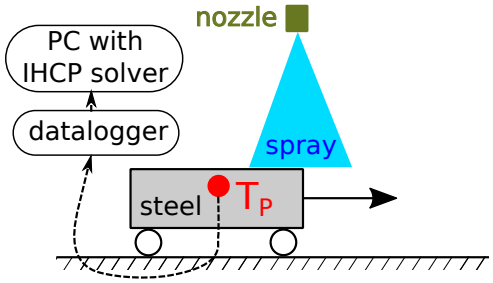

FIGURE 4. A schematic illustration of the experimental setup used in the present study.

cooling intensity. HTC becomes a function of the surface temperature T_S and the spatial coordinate in the direction of the plate motion. The experiment configuration is schematically shown in Fig. 4.

The heat conduction is dominantly normal to the cooled surface Γ_1 when the relative speed is high or when a uniform cooling is exhibited in the footprint of the nozzle, fully covering Γ_1. The lateral fluxes are significantly smaller and can be neglected in the numerical model. This assumption is found valid for the steel rolling where the relative speed is about 1 m/s. This assumption violates for the continuous casting with a rather small relative speed in meters per minute. The former allows considering only a small axisymmetric cutout from the steel plate, all the way through it and with the measuring point P of the thermocouple located on the axis of symmetry. Furthermore, the thermal boundary condition q_t (or HTC_t) can be considered to be constant over the surface Γ_1. The axisymmetric model is depicted in Fig. 5. The internal structure of the thermocouple can be seen. The 0.5 mm grounded thermocouple is inserted into the hole having a slightly larger bore. A precise welding is applied at the cooled surface to connect the thermocouple with the steel. The roughness of the cooled surface is homogenized afterwards.

A structured grid with 4k volume elements was used to properly resolve the heat transfer in a multi-material thermocouple and through the steel bulk. The most important parameters of the numerical model are summarized in Tab. 1. The size of the time step Δt is in correspondence with the recording frequency of the datalogger (320 Hz).

The heat transfer coefficient HTC_t and the corresponding surface temperature T_{St} were calculated at each discrete time t (see Fig. 6). The peaks of HTC_t curve correspond to the sudden temperature drops when the steel plate enters the spraying region (Fig. 4).

The L^1 *error norm* and the *max error* are used to evaluate the simulation results:

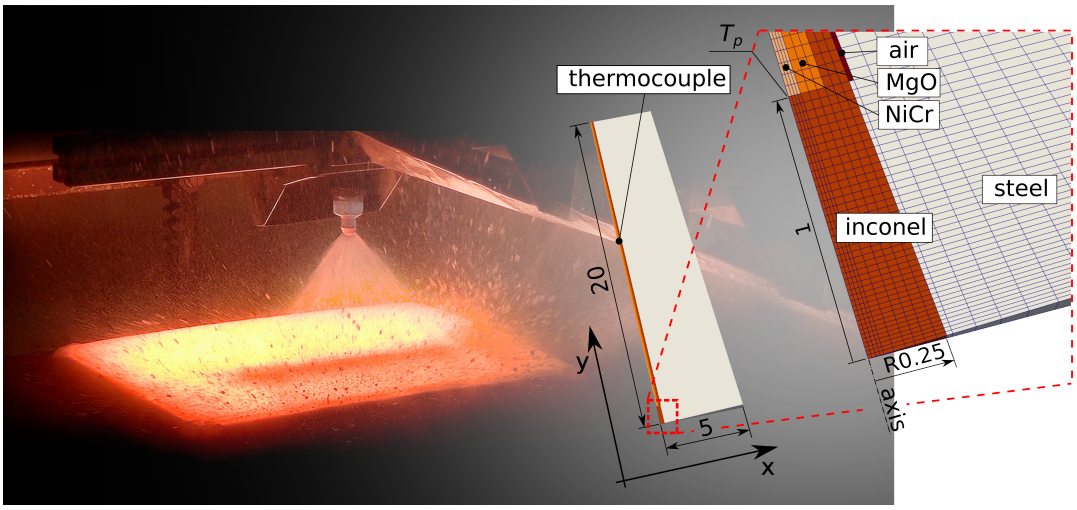

FIGURE 5. Cooling of a hot steel sample by a spraying nozzle (left) and the corresponding axisymmetric model for the IHCP with the detailed view on the K-type thermocouple embedded (right).

TABLE 1. Parameters of the IHCP calculation.

parameter name	value
Δt (s)	1/320
ns (−)	155k
N (−)	10
ddtScheme	Euler
grad(T)	linear
laplacian(DT,T)	Gauss linear Corrected
interpolation	linear
snGradSchemes	orthogonal
underrelaxation	none
linear solver	PCG (or GAMG)
minimizer	BOBYQA (bounded quadratic)
absXTol ($\mathrm{Wm^{-2}K^{-1}}$)	1
lower bound ($\mathrm{Wm^{-2}K^{-1}}$)	0
upper bound ($\mathrm{Wm^{-2}K^{-1}}$)	100000
parallel solve	no

$$L^1 = \frac{1}{ns}\sum_{i=1}^{ns} |T_{Pi} - T_i| \tag{9}$$

$$max\ error = \max_{i=1,ns} |T_{Pi} - T_i|, \tag{10}$$

with ns and T_i denoting respectively the number of the temperature T_P samples and the temperature from the IHCP calculation at the point P. In addition to (9) and (10), the results are further compared qualitatively by plotting HTC_t and calculated T at some arbitrary time window, as highlighted by the dashed rectangle in Fig. 6.

4. FORWARD TIME MARCHING FOR IHCP STABILITY

A sudden change of the thermal conditions at the surface Γ_1 results in a damped and lagged response of the internal temperature T_P due to the thermal resistance of the material. The response further delays and decreases for the deeper immersion of the thermocouple under the surface Γ_1. The measurement noise can overwhelm the true change of T_P. Furthermore, the smaller the time step Δt is, the more significant role of the noise is. The IHCP becomes ill-posed and requires a stabilization technique. In the present study a forward time marching, as presented in Fig. 2, is applied. The stability and convergence

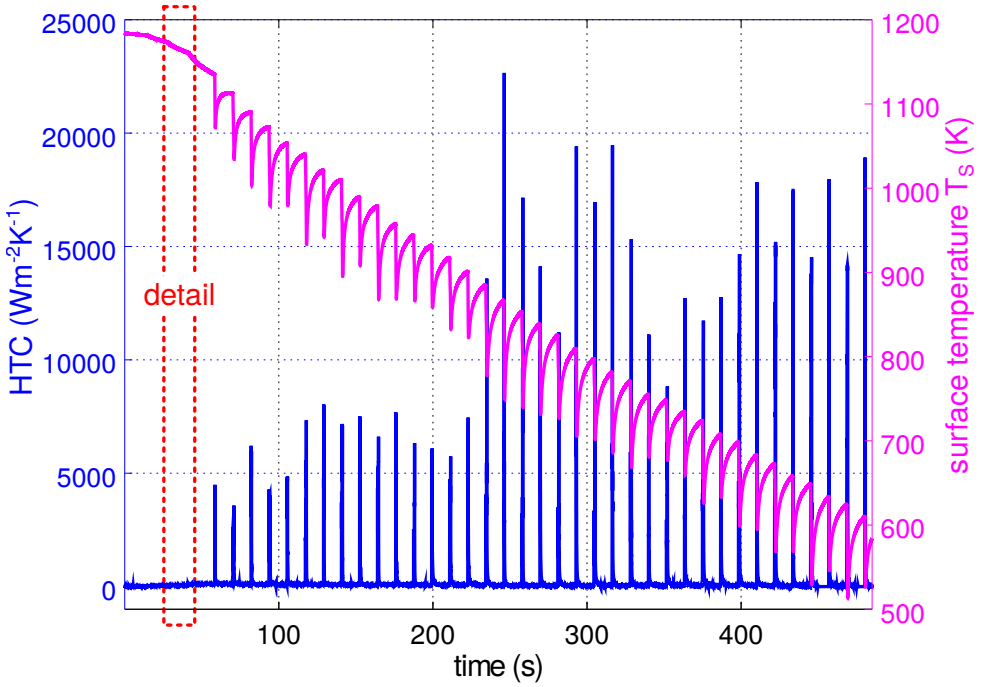

FIGURE 6. Results of the IHCP: the heat transfer coefficient HTC (in blue) and the surface temperature T_S (in magenta); The detail (the red dashed rectangle) was used to compare HTC of the IHCP solver with different settings.

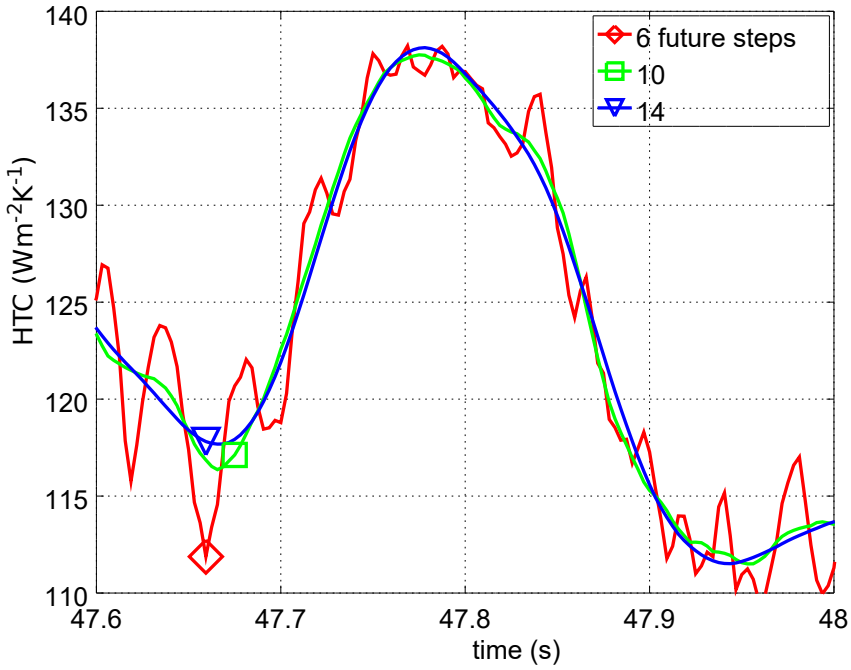

FIGURE 7. The effect of the number N of forward time steps on HTC_t; $N = 6, 10 \text{and} 14$; The optimal $N = 10$ was determined using the steepest descent of the 1st temporal derivative of the temperature at the point P (see Fig. 8).

of the IHCP solution depend on the number of the forward time steps N. The calculated HTC_t suffers from the numerical dispersion with the amplified local oscillations at the low range (see $N = 6$, Fig. 7). The algorithm becomes instable and finally diverges. The smoothing of the HTC_t is observed in case of 14 forward time steps and the local peaks are smeared out (see Fig. 7). The algorithm is stabilized; however, a numerical diffusion is introduced.

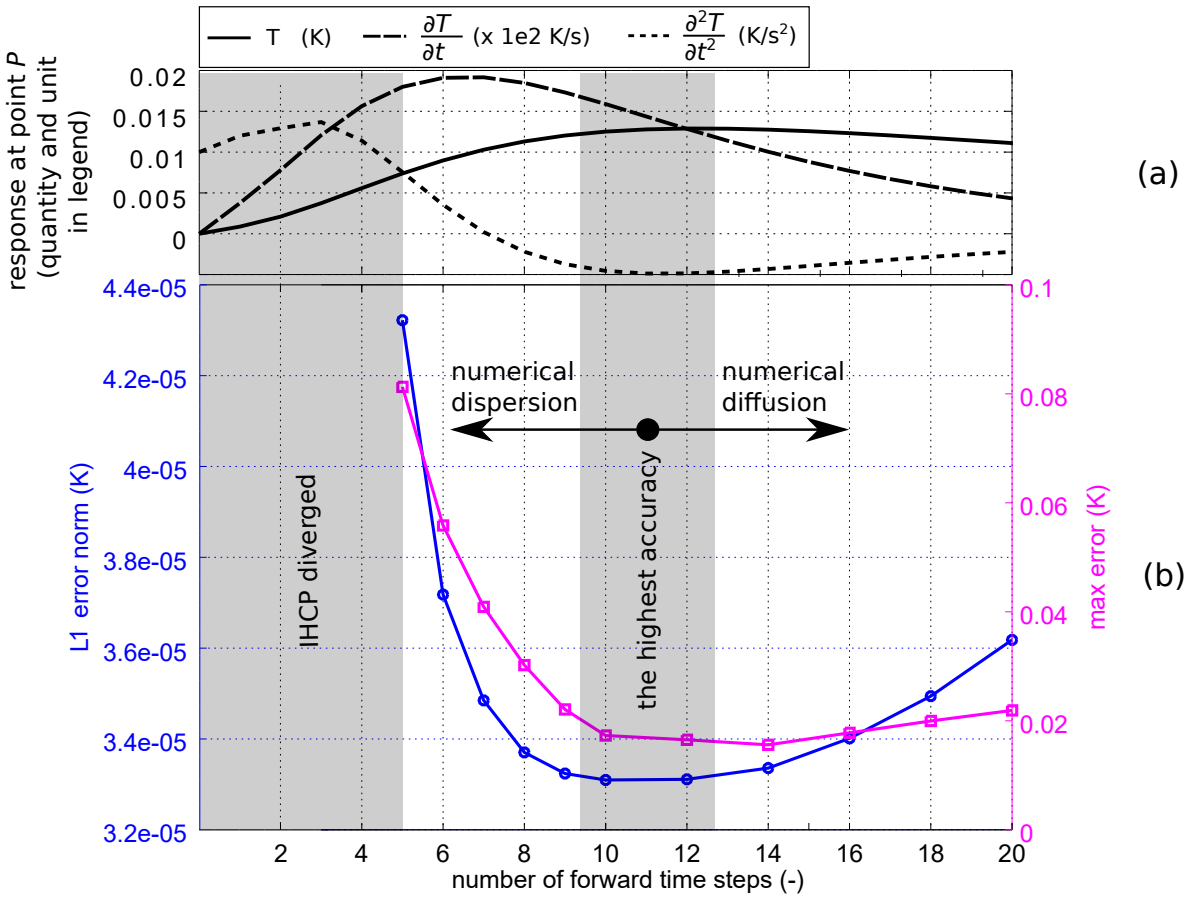

FIGURE 8. The IHCP results analysis: (a) forward time steps vs. the response of the temperature (solid curve) and its temporal derivatives (dashed and dotted curves) at the point P to the impulse of the unit heat flux at Γ_1 at time corresponding to $N = 10$; (b) the optimal N resides at the minimum of L^1 error norms (in blue) and *max errors* (in magenta).

In this study the optimal number N of forward time steps was selected based on the response of the temperature at the point P to the impulse of the unit heat flux q_t applied at Γ_1 at time corresponding to $N = 0$. Figure 8a shows the temperature (the solid curve), and the corresponding 1st (the dashed curve) and 2nd (the dotted curve) temporal derivatives. The optimal $N = 10$ coincides with the location of the steepest descent of the 1st temporal derivative of the temperature. The optimal $N = 10$ naturally coincides also with the minimum of the 2nd temporal derivative. In Fig. 8b, the corresponding L^1 error norms and *max errors* are plotted as a function of the number N of forward time steps. The smallest errors are found with $N = 10$. As the N decreases, dispersive errors become dominant until the divergence finally occurs. As the N increases, diffusive errors grow; however, the stability of the IHCP solver is improved. The optimal N can be alternatively determined according to [56].

5. Linear vs. constant HTC during forward time marching

A functional form of the heat transfer coefficient HTC_t should be defined for the forward time marching procedure. In the present work, the following relationship is applied for the parameter X of the minimization (7).

$$HTC_{t+i} = HTC_{t-1} + X \quad \text{with} \quad i = 0, ..., N-1, \tag{11}$$

in which the parameter X stands for a HTC_t increment. Alternatively, the HTC_t can be assumed to change linearly [57], as follows:

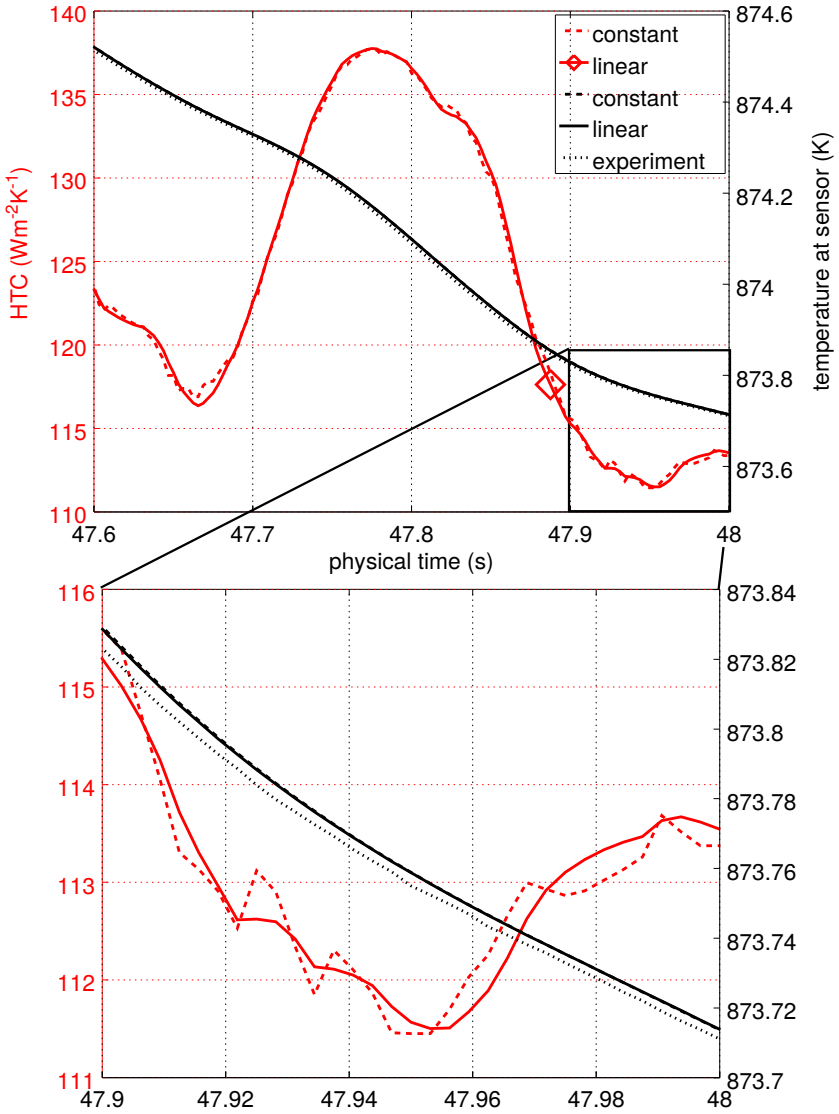

FIGURE 9. The choice of the parameter X: constant (dashed lines) and linear (solid lines) profile of HTC_t assumed during future time steps.

$$HTC_{t+i} = HTC_{t-1} + \sum_{i=0}^{N-1} X\Delta t, \tag{12}$$

in which X becomes a temporal slope of HTC_t. Results (HTC_t and T) of these two approaches are compared in Fig. 9. While not much of difference can be seen between the temperatures T, the values for the heat transfer coefficient HTC_t deviate noticeably from each other. The equation (11) produces dispersion between two subsequent time steps. The HTC_t obtained with (12) is significantly smoother. Since the error level as well as the calculation time are nearly identical (see Fig. 10), the approach (12) is preferable. Please note, the scale in Fig. 10 are purposely adjusted the same as those in Fig. 12 in the following section 6, in which different minimizers are compared.

6. Optimal minimizer for IHCP

Several iterations are typically required for the iterative minimizer to solve (7). In each iteration the minimizer must wait until the heat conduction equation (8) is computed N times. The minimizer with the fastest convergence rate is thus preferred. The proposed IHCP solver is based on the *laplacianFoam*

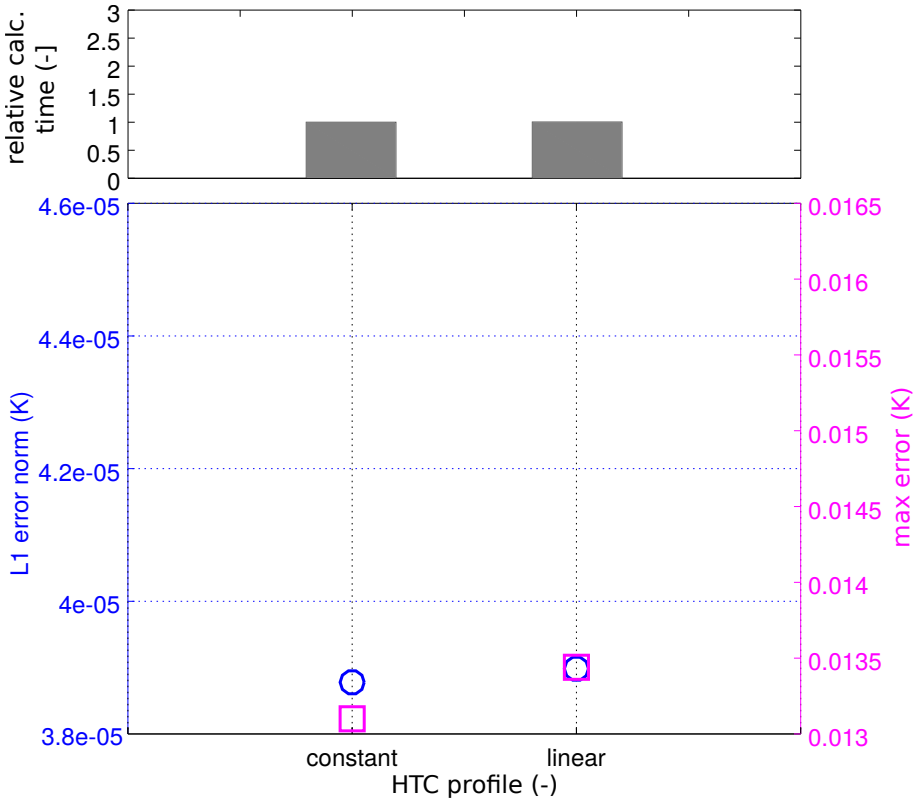

FIGURE 10. A negligible effect of the constant and linear HTC_t during future time steps on errors and relative calculation costs.

of the CFD open-source code OpenFOAM® , which was extended by incorporating several minimizers. The open-source optimization libraries NLopt and DLIB were employed. A list of tested minimizers follows with a brief description.

BOBYQA [58] is an acronym for a bound optimization by quadratic approximation. The method concentrates on minimizing damage from computer rounding errors, e.g. implementation of the RESCUE feature etc. COBYLA [59] is an acronym for a constrained optimization by linear approximation. Unlike BOBYQA, linear interpolations of the variables in the trust region typically results in a slower convergence. NELDER-MEAD method [60] is a direct search method. In n dimensions, a simplex is constructed from $n + 1$ vertices. The method consists of five steps: ordering, reflection, expansion, contraction and shrink. SBPLX (Subplex) [61] is a variant of the Nelder-Mead method and is claimed to perform more efficient and robust. BFGS [62] is an acronym for Broyden-Fletcher-Goldfarb-Shanno algorithm. It is a quasi-Newton method that uses an approximate Hessian matrix. It is similar to the conjugate gradient method except it is said to converge in fewer iterations. The method is particularly suitable for large problems, e.g. for the multi-parameter optimization. In the present paper it is the only method examined from the DLIB library. It is also the only method which requires gradient evaluations. Previously mentioned minimizers are derivative-free methods.

HTC_t curves calculated with the above-mentioned minimizers are shown in Fig. 11. Note that the convergence criterion and the search interval were identical for all minimizers. Large, unphysical dispersion is evident with COBYLA; dispersion can be also seen with NELDER-MEAD. A relatively smooth HTC_t is produced by SBPLX and BFGS methods. The smoothest profile of HTC_t is obtained with BOBYQA.

The minimizers performance was examined by comparing the calculation time and the error level (9), (10) in Fig. 12. The most favorable calculation time was attained with BOBYQA. Furthermore, BOBYQA is burdened with only small errors when compared to other minimizers. COBYLA also offers short calculation times; however, the errors and dispersion are excessive. BOBYQA and SBPLX show similar level of errors; however, SBPLX is 2.5times slower. Note that to compute HTC_t as shown in Fig. 6, a serial calculation required about three hours (10 future steps, BOBYQA, Intel Core i5-3570K CPU @ 3.4 GHz). SBPLX is the slowest minimizer herein. BOBYQA clearly outperforms other minimizers in all three aspects: the calculation time, error and smoothness.

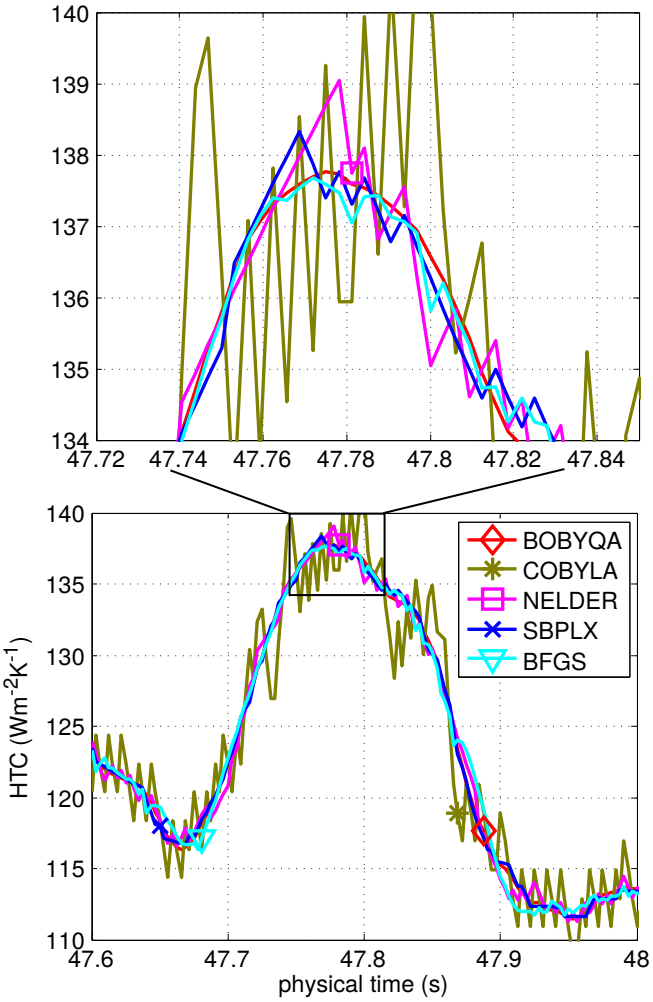

FIGURE 11. The effect of minimizer on HTC_t with a detailed view at the top; BOBYQA gives the smoothest results whilst COBYLA produces significant numerical dispersion.

In Fig. 13, the iteration process of each minimizer is depicted for an arbitrarily selected time t, proving fast convergence of BOBYQA. A similar behavior would be seen at any other time t.

7. Performance of PCG and GAMG solvers

Additional improvement of the IHCP solver performance is possible by tuning a linear equation solver for the heat conduction equation (8), which is a parabolic-type PDE. Since the size of the problem is small (4k volume elements), solvers with the low per-iteration costs are promoted such as the preconditioned conjugate gradient (PCG). The geometric agglomerated algebraic multigrid (GAMG) is also considered.

The conjugate gradient (CG) method requires preconditioning to improve the condition number of the solved system and the rate of convergence. In OpenFOAM® so-called faster diagonal incomplete Cholesky (FDIC) preconditioner is known to converge better than DIC [63]. Other preconditioners are typically less effective or not suitable for sequential CPU architectures at all, e.g. polynomial preconditioners [64].

The multigrid (GAMG) is suitable for large and ill-posed problems in OpenFOAM. The GAMG needs less iterations than PCG; however, extra costs arise due to the mesh refinement and mapping of the field data.

Firstly, the convergence criteria of the linear solver should be discussed. The relative tolerance is not recommended for the transient calculations. Therefore, the absolute tolerance [65] was used to decide whether the solution of (8) is converged. The IHCP, as described in section 3, was calculated with the absolute tolerance in range from 1e-12 to 1e-2. The PCG solver requires the tolerance and the preconditioner as an input (Tab. 2). For the numerous GAMG solver parameters the default settings were used (Tab. 2). The performance was compared in sense of the calculation time and the error level defined by (9), (10) which are shown in Fig. 14 and Fig. 15 respectively. In both figures, regions highlighted in

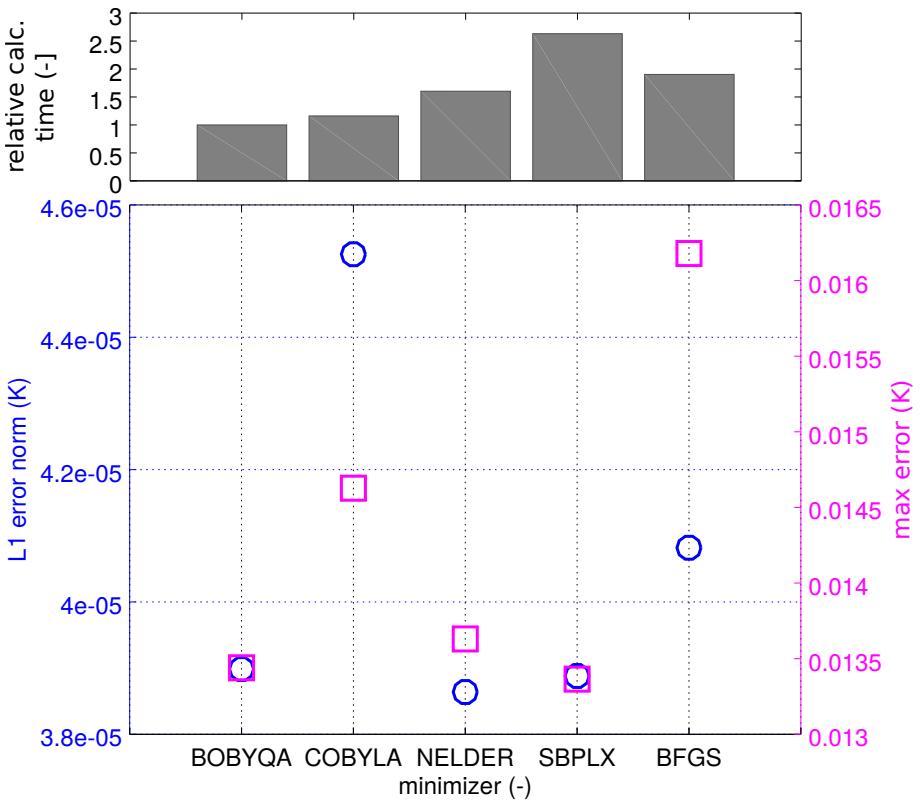

FIGURE 12. HTC_t errors and relative calculation costs of different minimizers – BOBYQA is superior to others.

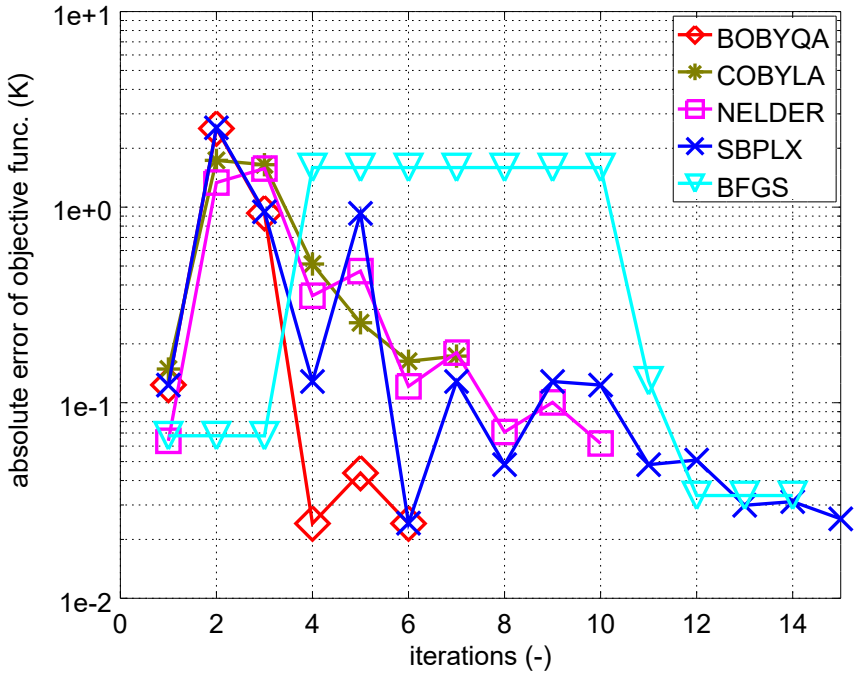

FIGURE 13. An example of the iteration process of the minimizers considered in the present study.

TABLE 2. Settings of linear solvers.

linear solver	GAMG	PCG
preconditioner	-	FDIC
tolerance	1e-4	1e-8
relTol	-	-
minIter	3	-
maxIter	100	-
smoother	DIC	-
nPreSweeps	0	-
nPostSweeps	2	-
nFinestSweeps	2	-
scaleCorrection	TRUE	-
directSolveCoarsestLevel	FALSE	-
cacheAgglomeration	on	-
nCellsInCoarsestLevel	500	-
agglomerator	faceAreaPair	-
mergeLevels	1	-

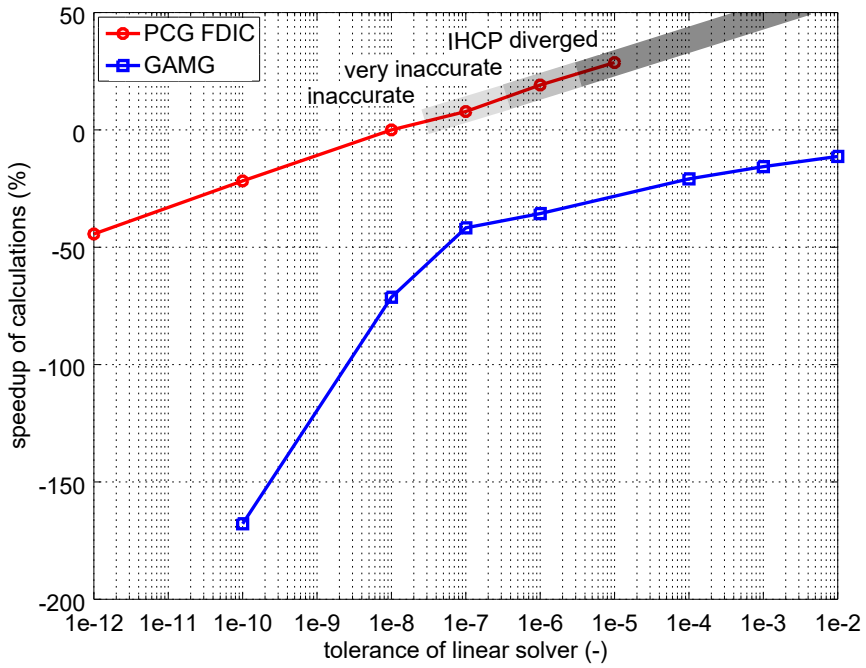

FIGURE 14. Performance comparison of the preconditioned (FDIC) conjugate gradient (PCG) method and the multigrid (GAMG) solver from the OpenFOAM® CFD package for the IHCP with 4k finite volume elements.

shades of grey show that PCG can produce inaccurate results or even lead to the solution divergence, which does not happen with the GAMG solver in the entire range of the absolute tolerances considered. The GAMG solver beneficially exhibits the plateauing of error levels that are in correspondence with a fixed number of GAMG iterations. Unlike the PCG [66][67], the GAMG converges fast because of the effective removal of low frequency errors from the solution during sub-cycling.

Nevertheless, the PCG offers a fair enough accuracy with the absolute tolerance of 1e-8. Moreover, the GAMG is nearly by 70% slower than the PCG at this level of accuracy. For the IHCP with a relatively small size (<10k volume elements), the PCG is clearly a better choice than the GAMG; however, the accuracy concern must be addressed.

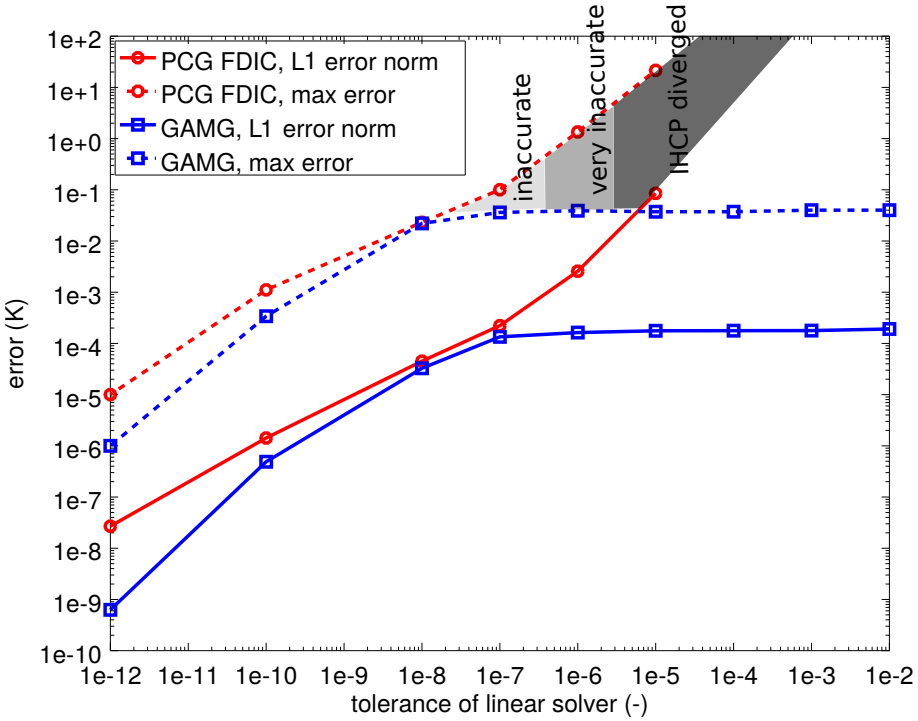

FIGURE 15. Convergence of the preconditioned (FDIC) conjugate gradient (PCG) method and the multigrid (GAMG) from the OpenFOAM® CFD package from the perspective of L^1 (solid curves) and *max errors* (dashed curves) for the IHCP with 4k finite volume elements.

8. COMPARISON WITH THE LINE-BY-LINE IHCP SOLVER

The IHCP has been the central tool of research in the Heat Transfer and Fluid Flow Laboratory at Brno University of Technology since decades. It has been continuously developed improving both the accuracy and the speed of calculations. First IHCP solvers were based on the original Beck's approach. Soon it was realized that non-linear problems are better solved with a direct minimization. Henceforth, a hybrid IHCP solver was used, combining the downhill simplex method applied at steep changes of the temperature curve (4) and the Beck′s approach applied elsewhere [68]. Meanwhile, two artificial intelligence mechanisms were considered: the neural network and the genetic algorithm [69]. It was pointed out that only a combination of both may become a competitive tool to the Beck′s approach. Nevertheless, since then it has been neither further developed nor successfully and repeatedly utilized in any commercial projects on reconstruction of thermal boundary conditions during spray cooling. Later, full-domain methods were examined, namely three variants: the full-domain method, the full-domain method with regularization and sub-domain methods [70]. Only with the sub-domain method it was possible to get results as fast as with the sequential algorithm. Although the accuracy was improved significantly by one order of magnitude, the IHCP solver was limited to constant thermal properties. Thus, the sub-domain methods still remain to be a questionable tool to tackle practical inverse heat conduction problems. At present, the sequential IHCP algorithm is commonly used with a direct minimization (the quadratic minimization) in the Heat Transfer and Fluid Flow Laboratory [71][72]. The direct part of the in-house IHCP solver is computed using the line-by-line algorithm [73], hereafter referred to as the line-by-line IHCP solver. The proposed IHCP solver in OpenFOAM is compared to the line-by-line solver in the following text.

In Fig. 16, the heat transfer coefficients HTC_t time lines are shown. The red and black curves represent results of the proposed and the line-by-line IHCP solver respectively. Despite considering identical stopping criterion for the minimization (7) in both IHCP solvers, different stopping criteria were used with the linear solvers. This fact explains the wiggles observed with the black curve. In other words, the convergence tolerance was stricter in OpenFOAM® , which is clearly confirmed by comparing error levels of both IHCP solvers in Fig. 17. Although the linear solver tolerance was more restrictive with OpenFOAM® , resulting in by one order of magnitude lower error levels than those obtained with the line-by-line IHCP solver, the proposed IHCP solver was twice as fast on the same machine (Intel Core i5-3570K CPU @ 3.4 GHz), as can be seen at the top of Fig. 17.

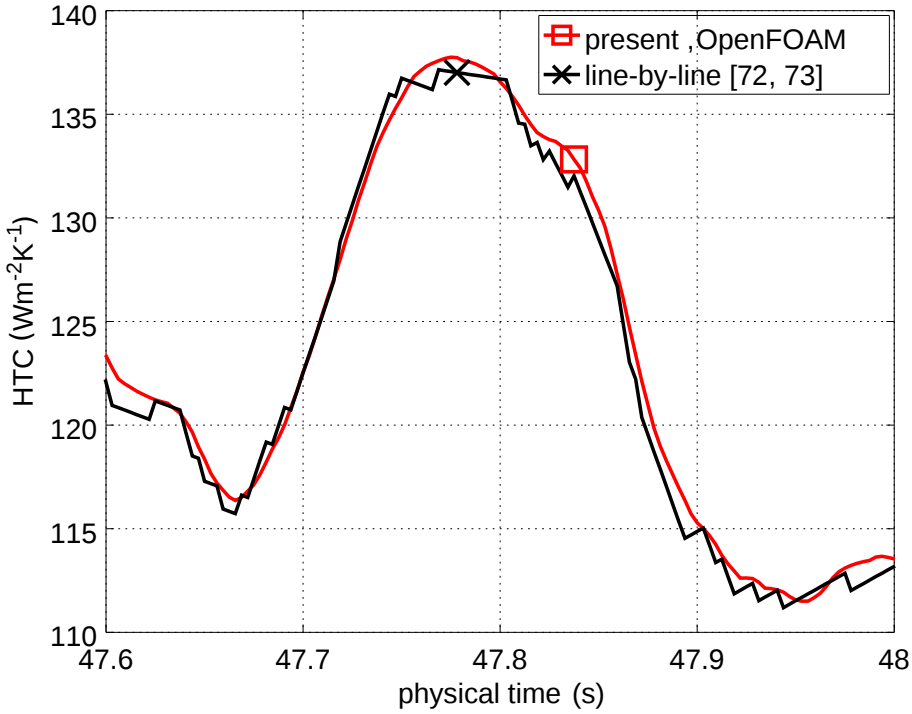

FIGURE 16. Results (HTC_t) of the proposed (OpenFOAM®) and the line-by-line IHCP solver.

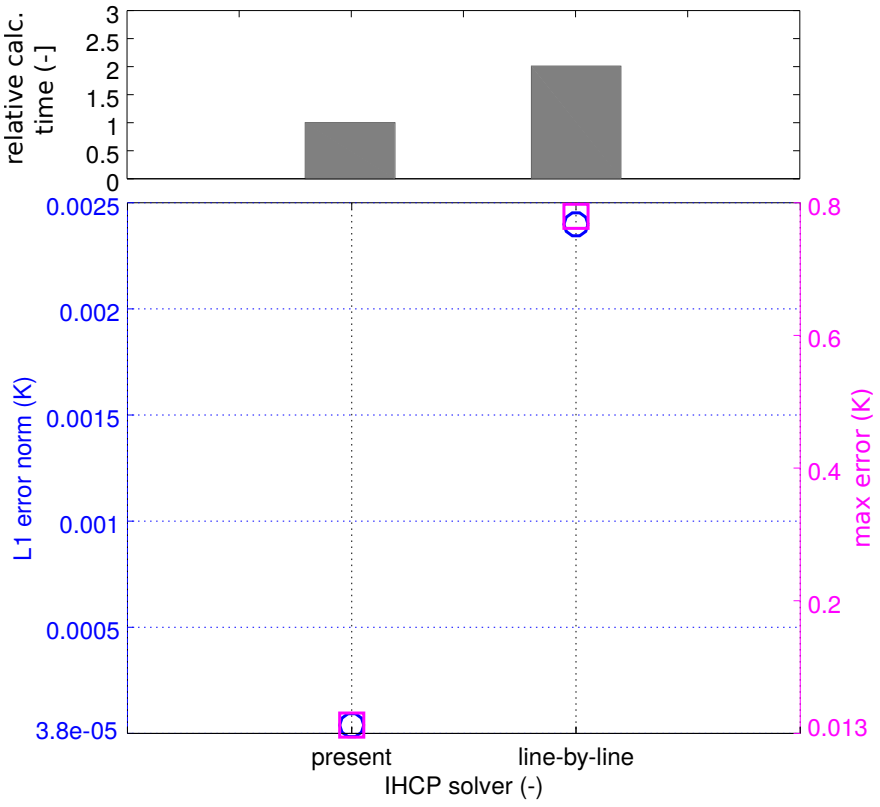

FIGURE 17. HTC_t errors and relative calculation costs of the proposed and the line-by-line IHCP solver.

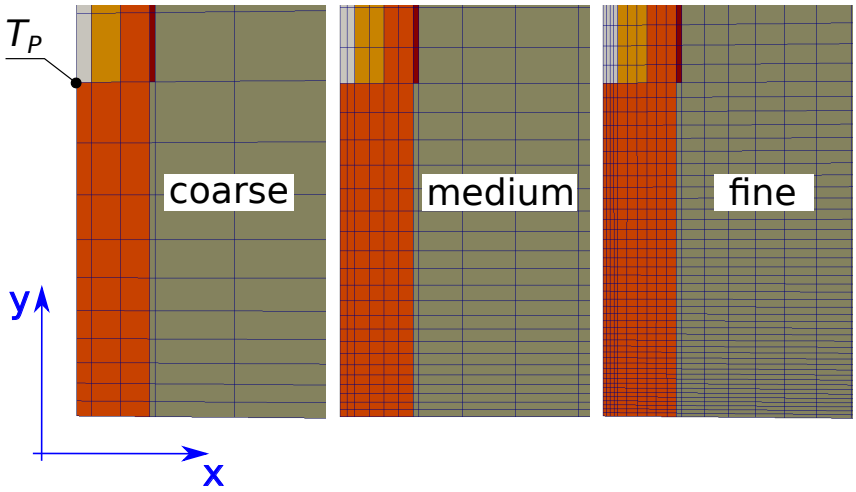

FIGURE 18. A grid resolution in the vicinity of the thermocouple (the temperature T_P measured at the point P); three grid sizes (coarse, medium and fine) considered for the grid sensitivity analysis.

9. Sensitivity analysis on grid size

Three grid types (Fig. 18) were used to assess the sensitivity of the IHCP solution to the refinement. The coarse, medium and fine grids had about 250, 1k and 4k volume elements respectively. The FDIC PCG linear solver was used with a tight tolerance of 1e-12. Other settings were adopted from Tab. 1.

In Fig. 19, the time series for the heat transfer coefficient HTC_t are displayed. The HTC_t curves evidently become grid-independent with the grid refinement. This statement is confirmed by identifying similar error levels obtained with the medium and the fine grid, as shown in Fig. 20. In the same figure, the relative calculation times are shown. The calculation with the fine grid took 5.7 times longer than with the coarse grid. The fine grid (4k volume elements) was used in all the above-mentioned studies to guarantee grid-independent results.

10. Conclusions

The sequential IHCP solver was developed in the finite volume framework of the open-source CFD code OpenFOAM® . The distributed *laplacianFoam* solver was used as a basis, supplemented with the open-source functionality of the optimization libraries NLopt and DLIB. The main goal was to arrive at a fast and accurate IHCP solver. Performance of the IHCP solver was analyzed using real experimental data recorded at 320 Hz (totaling 155k samples) with one K-type thermocouple embedded in a steel plate. The most important findings to be highlighted are:

- Accurate thermo-physical properties, typically temperature dependent, are necessary to yield realistic reconstruction of thermal boundary conditions such as HTC_t,
- The depth of the thermocouple, i.e. the position of the measuring point, must be precisely known to eliminate bias of HTC_t,
- Internal geometry of the thermocouple should be carefully considered in the numerical model,
- The number of future time steps should be determined prior to running the IHCP to stabilize the solution. For that purpose, a response of the unit heat flux can be utilized. Not enough future steps produce dispersive errors and may eventually lead to divergence. Too many future time steps produce diffusive errors and slow down the calculation process.
- A constant and linear profile of HTC_t were assumed in future time stepping. Although both strategies reveal identical behaviour with regard to the calculation times and error level, the linear profile should be preferred as it suppresses dispersive wiggles seen with HTC_t.
- From several minimizers, BOBYQA minimizer offered a superior convergence rate without sacrifice accuracy. Furthermore, this minimizer produced the most continuous thermal boundary condition as a function of time. In this IHCP study, one thermocouple was present. Minimizer other than BOBYQA might perform better when more thermocouples are involved.
- The preconditioned conjugate gradient (PCG FDIC) is preferred over the multigrid (GAMG) solver on the problem size considered (4k volume elements). Unlike the multigrid, it may however suffer from poor convergence when a large absolute tolerance of the solver is used. Hence, the

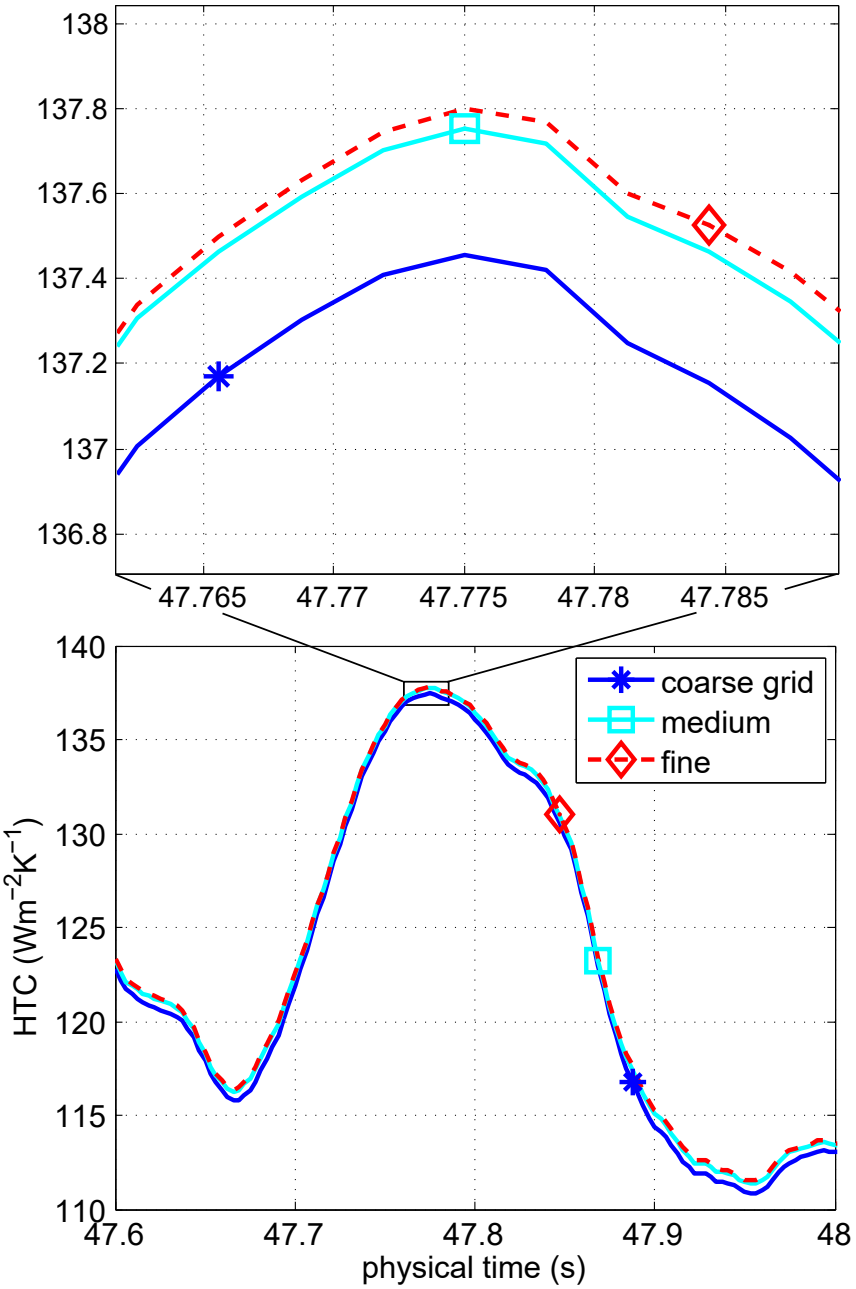

FIGURE 19. Results (HTC_t) obtained with the coarse (solid blue), medium (solid cyan) and fine (dashed red) grid having respectively 250, 1000 and 4000 volume elements.

multigrid is a better choice when the appropriate level of tolerance is unknown, e.g. when starting a new case.

- The IHCP solver developed in OpenFOAM® can be easily extended to problems with multiple thermocouples. Moreover, large problems can be solved in parallel on more processors. The GAMG efficiency is expected to grow and ultimately outperform that of the PCG. Unstructured grids are supported in OpenFOAM® ; hence, complex curved geometries embedded with many thermocouples can be easily handled.
- The OpenFOAM® code of the present IHCP solver is simple, intuitive and thus easy to read.

ACKNOWLEDGEMENTS

Authors gratefully acknowledge financial support from the Ministry of Education under the programme INTER-EXCELLENCE, within the project LTAUSA19053. Computational resources were supplied by the project "e-Infrastruktura CZ" (e-INFRA LM2018140) provided within the program Projects of Large

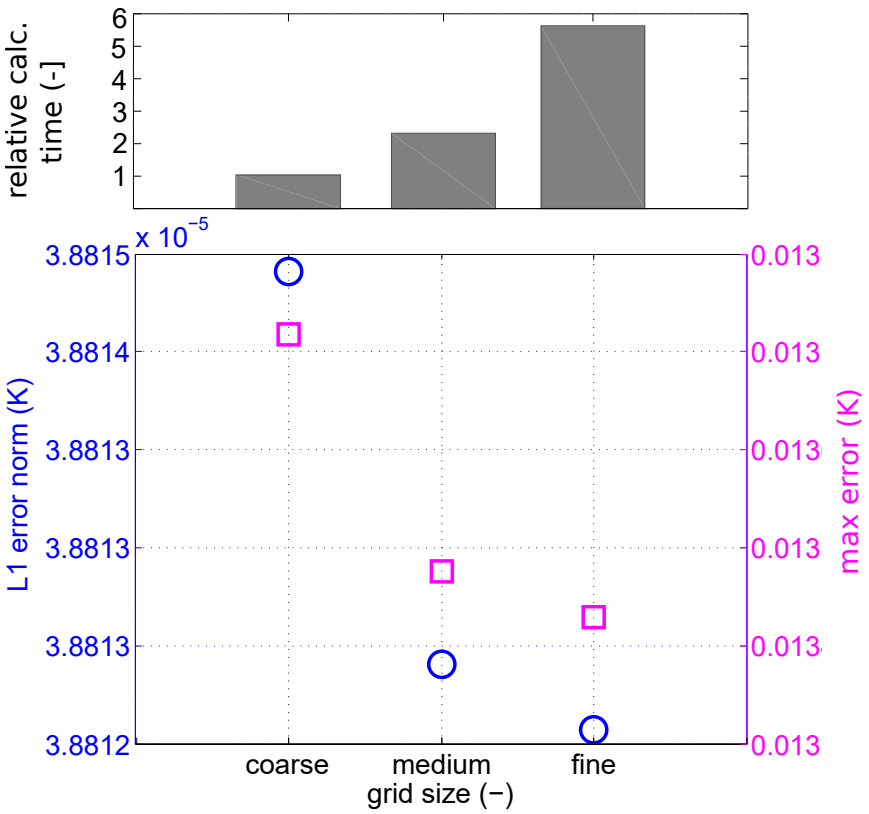

FIGURE 20. HTC_t errors (bottom) and relative calculation costs (top) obtained with the coarse, medium and fine grid; Convergence to grid-independent results is noticeable.

Research, Development and Innovations Infrastructures. Computational resources were provided by the ELIXIR-CZ project (LM2018131), part of the international ELIXIR infrastructure.

Author Contributions: Conceptualisation, J.B., A.V. and T.-W. L.; methodology, J.B., A.V. and T.-W. L.; software, J.B. and A.V.; validation, J.B. and J.K.; formal analysis, J.B. and J.K.; data curation, J.K.; writing—original draft preparation, J.B.; writing—review and editing, J.B., A.V., J.K. and E. K.-S.; visualisation, J.B.; supervision, E. K.-S. and J.K.; All authors have read and agreed to the published version of the manuscript.

REFERENCES

[1] H.-J. Reinhardt, D. N. Hao, J. Frohne, and F.-T. Suttmeier, "Numerical solution of inverse heat conduction problems in two spatial dimensions," *Journal of Inverse and Ill-posed Problems*, vol. 15, pp. 181–198, 2007.

[2] S. S. Pereverzyev, R. Pinnau, and N. Siedow, "Initial temperature reconstruction for nonlinear heat equation: application to a coupled radiative-conductive heat transfer problem," *Inverse Problems in Science and Engineering*, vol. 16, pp. 55–67, 2005.

[3] K. Grysa, L. Hozejowski, W. Marczewski, and E. Sendek-Matysiak, "Thermal diffusivity estimation from temperature measurements with a use of a thermal probe," in *In Proceedings of Experimental Fluid Mechanics 2009*, Liberec, Czech Republic, 2009.

[4] H.-S. Ren, "Application of the heat-balance integral to an inverse stefan problem," *International Journal of Thermal Sciences*, vol. 46, pp. 118–127, 2007.

[5] M. Ikehata, "An inverse source problem for the heat equation and the enclosure method," *Inverse Problems*, vol. 23, pp. 183–202, 2007.

[6] L. Huiping and G. Zhao, "Inverse heat conduction analysis of quenching process using finite-element and optimization method," *Finite Elements in Analysis and Design*, vol. 42, pp. 1087–1096, 2006.

[7] A. Bendada and K. T. Nguyen, "Inverse heat conduction of time and space dependent temperature during material," in *AIP Conference Proceedings*, vol. 557, 2001, p. 1562.

[8] F. Lopez and V. d. P. Nicolau, "Inverse heat transfer approach for ir image reconstruction: Application to thermal non-destructive evaluation," *Applied Thermal Engineering*, vol. 33-34, pp. 109–118, 2012.

[9] L. Yang, X. Sun, and Y. Chu, "Boundary shape inversion of two-dimensional steady-state heat transfer system based on finite volume method and decentralized fuzzy adaptive pid control," *Applied Sciences*, vol. 10, p. 153, 2020.

[10] J. Xiong, Z. Wang, P. Xiong, T. Lu, and Y. Yang, "Experimental investigation on transient boiling heat transfer during quenching of fuel of fuel cladding surfaces," *International Journal of Heat and Mass Transfer*, vol. 148, p. 119131, 2020.

[11] M. Pilarczyk, B. Wegloswski, and L. O. Nord, "A comprehensive thermal and structural transient analysis of a boiler's steam outlet header by means of a dedicated algorithm and fem simulation," *Energies*, vol. 13, p. 111, 2020.

[12] C. Guo and S. Malkin, "Inverse heat transfer analysis of grinding, part1: methods," *Journal of Engineering for Industry*, vol. 118, pp. 137–142, 1996.
[13] L. Mesquita, P. Piloto, M. A. P. Vaz, and T. M. G. Pinto, "Decomposition of intumescent coatings: Comparison between a numerical method and experimental results," in *In Proceedings of Applications of Structural Fire Engineering 2009*, Prague, Czech Republic, 2009.
[14] S. K. Kim and I. M. Daniel, "Solution to inverse heat conduction problem in nanoscale using sequential method," *Numerical Heat Transfer: Part B: Fundamentals*, vol. 44, pp. 439–456, 2003.
[15] T. Loulou and E. P. Scott, "An inverse heat conduction problem with heat flux measurements," *International Journal of Numerical Methods in Engineering*, vol. 67, pp. 1587–1616, 2006.
[16] H. Najafi, O. Uyanna, and J. Zhang, "Application of artificial neural network as a near-real time technique for solving non-linear inverse heat conduction problems in a one-dimensional medium with moving boundary," in *In Proceedings of ASME 2020 Heat Transfer Conference, Virtual, Online*, 2020, p. V001T02A013.
[17] J. Lee, "Role of surface roughness in water spray cooling heat transfer of hot steel plate," *ISIJ Interntional*, vol. 49, pp. 1920–1925, 2009.
[18] M. Chabicovsky, P. Kotrbacek, H. Bellerova, J. Kominek, and M. Raudensky, "Spray cooling heat transfer above leidenfrost temperature," *Metals*, vol. 10, p. 1270, 2020.
[19] B. Sunden, C. A. Brebia, and D. Poljak, *Advanced Computational Methods and Experiments in Heat Transfer XII*. Croatia: WIT Transactions on Engineering Sciences, 2012.
[20] M. Raudensky and J. Horsky, "Secondary cooling in continuous casting and leidenfrost temperature effects," *Ironmaking&Steelmaking*, vol. 32, pp. 159–164, 2005.
[21] X. Zhou, B. G. Thomas, C. A. H. B., A. H. C. E., and F. A. A. G., "Measuring heat transfer during spray cooling using controlled induction-heating experiments and computational models," *Applied Mathematical Modelling*, vol. 37, p. 3192, 2013.
[22] J. V. Beck and K. A. Woodbury, "Inverse heat conduction problem: Sensitivity coefficient insights, filter coefficients, and intrinsic verification," *International Journal of Heat and Mass Transfer*, vol. 97, pp. 578–588, 2016.
[23] M. N. Ozisik and H. R. B. Orlande, *Inverse Heat Transfer:Fundamentals and applications*. New York: Taylor&Francis, 2000.
[24] O. R. Burggraf, "An exact solution of the inverse problem in heat conduction theory and application," *Journal of Heat Transfer*, vol. 86, pp. 373–382, 1964.
[25] M. Monde, H. Arima, W. Liu, Y. Mitutake, and J. A. Hammad, "An analytical solution for two-dimensional inverse heat conduction problems using laplace transform," *International Journal of Heat and Mass Transfer*, vol. 46, pp. 2135–2148, 2003.
[26] E. Trefftz, "Ein gegenstuek zum ritzschen verfahren," in *In Proceedings of the 2nd International Congress of Applied Mechanics*, Orell Fussli Verlag, Zurich, 1926, pp. 131–137.
[27] I. Frank, "An application of least squares method to the solution of the inverse problem of heat conduction," *Journal of Heat Transfer*, vol. 85C, pp. 378–379, 1963.
[28] P. C. Hansen, *Computational Inverse Problems in Electrocardiology*. WIT Press, 2001, ch. The L-curve and its use in the numerical treatment of inverse problems.
[29] A. N. Tikhonov and V. Y. Arsenin, *On the solution of ill-posed problems*. New York, USA: John Wiley and Sons, 1977.
[30] H. Najafi, "Real-time heat flux estimation using filter based solutions for inverse heat conduction problems," Ph.D. dissertation, The University of Alabama, Tuscaloosa, Alabama, 2015.
[31] K. Grysa and A. Maciag, *Hetnarski R.B. (eds) Encyclopedia of Thermal Stresses*. Springer, Dordrecht, 2004, ch. Trefftz method in solving inverse heat conduction problems.
[32] M. J. Cialkowski and K. Grysa, "Trefftz method in solving the inverse problems," *Journal of Inverse and Ill-Posed problems*, vol. 18, pp. 595–616, 2010.
[33] B. A. Peavy, "A heat transfer note on temperature dependent thermal conductivity," *J. Thermal Insul. and Bldg. Envs.*, vol. 20, pp. 76–90, 1996.
[34] J. V. Beck, "Calculation of surface heat flux from an internal temperature history," *ASME Paper 62-HT-46*, 1962.
[35] J. V. Beck, B. Blackwell, and J. C. R. St. Clair, *Inverse Heat Conduction: Ill-Posed Problems*. New York, USA: John Wiley and Sons, Inc., 1985.
[36] D. G. Cuadrado, A. Marconnet, and G. Paniagua, "Non-linear non-iterative transient inverse conjugate heat transfer method applied to microelectronics," *International Journal of Heat and Mass Transfer*, vol. 152, p. 119503, 2020.
[37] H.-J. Reinhardt and D. N. Hao, "Sequential approximation to nonlinear inverse heat conduction problems," *Math. Comput. Modelling*, vol. 20, pp. 189–200, 1994.
[38] G. Blanc, M. Raynaud, and T. H. Chau, "A guide for the use of the function specification method for 2d inverse heat conduction problems," *Rev Gen Therm*, vol. 37, pp. 17–30, 1998.
[39] J. Taler and W. Zima, "Solution of inverse heat conduction problems using control volume approach," *International Journal of Heat and Mass Transfer*, vol. 42, pp. 1123–1140, 1999.
[40] S. Wang, Y. Deng, and X. Sun, "Solving of two-dimensional unsteady inverse heat conduction problems based on boundary element method and sequential specification method," *Complexity*, vol. 2018, p. 6741632, 2018.
[41] K. J. Dowding and J. V. Beck, "A sequential gradient method for the inverse heat conduction problem (ihcp)," *Transactions of the ASME*, vol. 121, pp. 300–306, 1999.
[42] S. A. Aghayan, D. Sardari, S. R. M. Mahdavi, and M. H. Zahmatkesh, "An inverse problem of temperature optimization in hyperthermia by controlling the overall heat transfer coefficient," *Journal of Applied Mathematics*, vol. 2013, p. 734020, 2013.
[43] N. Gnanasekaran and S. Balaji, "Inverse approach for estimating boundary properties in a transient fin problem," *Sadhana*, vol. 43, p. 108, 2018.
[44] S. Szenasi and I. Felde, "Modififed particle swarm optimization method to solve one-dimensional ihcp," in *In Proceedings of 16th IEE International Symposium on Computing Intelligence and Informatics*, Hungary, 2015, pp. 85–88.

[45] S.-M. Lin, "A sequential algorithm and error sensitivity analysis for the inverse heat conduction problems with multiple heat sources," *Applied Mathematical Modelling*, vol. 35, pp. 2607–2617, 2011.
[46] P. Ranut, "Optimization and inverse problems in heat transfer," Ph.D. dissertation, Universita degli studi di Udine, Italy, 2012.
[47] R. Das, "A simplex search method for a conductive-convective fin with variable conductivity," *International Journal of Heat and Mass Transfer*, vol. 54, pp. 5001–5009, 2011.
[48] R. Brociek and D. Slota, "Reconstruction of the boundary condition for the heat conduction equation of fractional order," *Thermal Science*, vol. 19, pp. S35–S42, 2015.
[49] P. Salomonsson and M. Oldenburg, "Investigation of heat transfer in the press hardening process," in *In Proceedings of 2nd International Conference on Hot Sheet Metal Forming of High-performance Steel*, Luleå, Sweden, 2009, pp. 239–246.
[50] H. G. Weller, G. Tabor, H. Jasak, and C. Fureby, "A tensorial approach to computational continuum mechanics using object orientated techniques," *Computers in Physics*, vol. 12, pp. 620–631, 1998.
[51] T. Holzmann, *Mathematics, Numerics, Derivations and OpenFOAM®* . Munich, Germany: www.holzmann-cfd.de, 2019.
[52] L. Chiwiakowsky and H. D. C. Velho, "Different approaches for the solution of a backward heat conduction problem," *Inverse Problems in Engineering*, vol. 11, pp. 471–494, 2003.
[53] OpenFOAM.com, "laplacianFoam documentation." [Online]. Available: https://www.openfoam.com/documentation/guides/latest/doc/guide-applications-solvers-basic-laplacianFoam.html
[54] S. G. Johnson, "The nlopt nonlinear-optimization package." [Online]. Available: http://github.com/stevengj/nlopt
[55] D. E. King, "Dlib-ml: A machine learning toolkit," vol. 10, pp. 1755–1758, 2009.
[56] J. Kominek and M. Pohanka, "Estimation of the number of forward time steps for the sequential beck approach used for solving inverse heat conduction problems," in *In Proceedings of 22nd International conference on materials and technology*, Ljubljana, Slovenia, 2014, p. 122.
[57] M. Pohanka, H. Votavova, M. Raudensky, J. Y. Hwang, J. W. You, and S. H. Lee, "The effect of water temperature on cooling during high pressure water descaling," *Thermal Science*, vol. 6, pp. 2965–2971, 2018.
[58] M. J. D. Powell, "The bobyqa algorithm for bound constrained optimization without derivatives," Department of Applied Mathematics and Theoretical Physics, Cambridge, England, Tech. Rep. NA2009/06, 2009.
[59] ——, "Direct search algorithms for optimization calculations," *Acta Numerica*, vol. 7, pp. 287–336, 1998.
[60] M. J. Box, "A new method of constrained optimization and a comparison with other methods," *Computer J.*, vol. 8, pp. 42–52, 1965.
[61] T. Rowan, "Functional stability analysis of numerical algorithms," Ph.D. dissertation, Department of Computer Sciences, University of Texas at Austin, USA, 1990.
[62] R. H. Byrd, P. Lu, J. Nocedal, and C. Zhu, "A limited memory algorithm for bound constrained optimization," *SIAM Journal on Scientific Computing*, vol. 16, pp. 1190–1208, 1995.
[63] W. J. Demmel, *Applied Numerical Linear Algebra*. Society for Industrial and Applied Mathematics, 1997.
[64] J. Bohacek, A. Kharicha, A. Ludwig, M. Wu, T. Holzmann, and E. Karimi-Sibaki, "A gpu solver for symmetric positive-definite matrices vs. traditional codes," *Computers & Mathematics with Applications*, vol. 78, pp. 2933–2943, 2019.
[65] F. Moukalled, L. Mangani, and M. Darwish, *The Finite Volume Method in Computational Fluid Dynamics: An Advanced Introduction with OpenFOAM and Matlab*. Switzerland: Springer Publishing Company, 2015.
[66] J. N. Shoosmith, *Numerical Analysis, Encyclopedia of Physical Science and Technology*. Academic Press, 2003.
[67] J. Nieplocha and W. C. Schreiber, "The use of conjugate gradients methods with a segregated finite volume procedure for solving transient incompressible navier-stokes equations," *Communications in numerical methods in engineering*, vol. 11, pp. 849–856, 1995.
[68] M. Raudensky, M. Pohanka, and J. Horsky, "Combined inverse heat conduction method for highly transient processes," *WIT Transactions on Engineering Sciences*, vol. 35, pp. 35–42, 2002.
[69] M. Raudensky, J. Horsky, J. Krejsa, and L. Slama, "Usage of artificial intelligence methods in inverse problems for estimation of material parameters," *International Journal of Numerical Methods for Heat Transfer and Fluid Flow*, vol. 6, pp. 19–29, 1996.
[70] V. Hribova, "Development of inverse sub-domain method for boundary conditions computation of heat conduction," Master's thesis, Brno University of Technology, Czech Republic, 2015.
[71] M. Pohanka and K. A. Woodbury, "A downhill simplex method for computation of interfacial heat transfer coefficients in alloy casting," *Inverse Problems in Engineering*, vol. 11, pp. 409–424, 2003.
[72] M. Pohanka and J. Horsky, "Inverse algorithms for time dependent boundary reconstruction of multidimensional heat conduction model," in *In Proceedings of Thermophysics 2007*, Bratislava, 2007, pp. 14–23.
[73] M. Pohanka and J. Ondrouskova, "Implicit numerical multidimensional heat-conduction algorithm parallelization and acceleration on a graphics card," *Materials and technology*, vol. 50, pp. 183–187, 2016.