UNIVERSITY OF MISKOLC
FACULTY OF MECHANICAL ENGINEERING AND INFORMATICS

DESIGNING AND IMPLEMENTING NUMERICAL METHODS TO SOLVE NONLINEAR HEAT TRANSFER PROBLEMS TO IMPROVE BUILDINGS' ENERGY EFFICIENCY

BOOKLET OF PHD THESIS

Prepared by

Humam Kareem Jalghaf Al-Janabi
Mechanical Engineering (BSc),
Mechanical Engineering (MSc)

ISTVÁN SÁLYI DOCTORAL SCHOOL OF MECHANICAL ENGINEERING SCIENCES

TOPIC FIELD OF BASIC ENGINEERING SCIENCES
TOPIC GROUP OF TRANSPORT PROCESSES AND MACHINES

Head of Doctoral School

Dr. Gabriella Bognár
DSc, Full Professor

Head of Topic Group

Dr. László Baranyi
Full Professor

Scientific Supervisor

Dr. Endre Kovács

Dr. Betti Bolló

Miskolc
2025

JUDGING COMMITTEE

Chair:

Secretary:

Members:

OFFICIAL REVIEWERS

1. INTRODUCTION

Energy efficiency in buildings is essential for addressing climate change and fostering sustainability. Buildings are major energy consumers, and optimizing their thermal performance can significantly reduce energy consumption and greenhouse gas emissions. Enhancing energy efficiency requires managing heat transfer within building components, including walls, roofs, and floors. Understanding this process involves heat transfer equations that account for material properties (e.g., thermal conductivity, density, and specific heat capacity) and boundary conditions (e.g., temperature, humidity, and airflow). Accurate thermal analysis of building walls requires numerical methods, and research has focused on transient heat transfer in multilayered media. Various mathematical models and computational methods, such as MATLAB simulations and computational fluid dynamics (CFD), have been used for this purpose.

The diffusion equation, a key mathematical model for heat transfer, has been extensively studied. Early analytical solutions assumed constant parameters, but space-dependent properties necessitate numerical approaches. Zoppou and Knight [1] provided analytical solutions for the advection-diffusion equation with spatially variable coefficients, though most cases require numerical solutions. Spatially discretized partial differential equations (PDEs) are converted into ordinary differential equations (ODEs), which can be challenging to solve, especially in three-dimensional cases. Explicit methods like Runge-Kutta are conditionally stable and computationally expensive, while implicit methods allow for larger time steps but require solving complex algebraic systems. Advances in implicit methods have addressed stability issues in stiff problems, such as those involving rapid temperature changes and high thermal diffusivity materials [2]. A novel approach in this research involves developing explicit, unconditionally stable numerical methods that are easily parallelizable. One such method is the two-stage odd-even hopscotch (OEH) algorithm, introduced by Gordon [3] and refined by Gourlay [4-6]. The OEH method has been applied to the Navier-Stokes equations [7], reaction-diffusion equations [8,9], and nonlinear Dirac equations [10]. Goede and Boonkkamp [11] improved OEH through vectorization for solving two-dimensional Burgers' equations. Recently, Maritim et al. [12,13] developed hybrid algorithms combining OEH, Crank-Nicolson, and Du Fort-Frankel methods, finding their implicit algorithms stable and accurate. Further research [14-16] identified limitations in OEH for stiff systems, leading to new algorithmic modifications to enhance stability and accuracy.

Heat transfer management is crucial for achieving energy efficiency in buildings. The thermal performance of walls plays a significant role in maintaining indoor comfort. Research has explored integrating phase change materials (PCMs) and thermal insulation into building envelopes. PCMs store and release latent heat, mitigating temperature fluctuations and reducing reliance on mechanical heating and cooling. Concurrently, thermal insulation minimizes heat transfer, further improving energy efficiency. Numerical simulations have been key in understanding the thermal behavior of PCM-integrated walls. Geng et al. [17] investigated optimal PCM and thermal insulation material (TIM) placements in multilayered walls, demonstrating energy savings of 46.69–64.73%. Liu et al. [18] studied PCM performance in lightweight building walls, finding significant heat flux reductions of up to 66.4% for east- and west-facing walls. Other studies highlight the effectiveness of combining PCM with conventional insulation. Tunçbilek et al. [19] found that integrating PCMs in building walls reduced energy consumption by up to 38.2%. Cascone et al. [20] optimized PCM use in office building retrofits for Mediterranean climates, aligning with the EU's 2020 sustainability goals. Jam et al. [21] conducted an economic analysis of PCM integration in educational buildings, determining an optimal 3 cm PCM thickness with a 50-month payback period. Abden et al. [22] evaluated PCM and insulation combinations in Australian homes, revealing cost savings of AU\$167.0/m² and energy efficiency improvements of up to 4.3 stars in various climate zones. Iffa et al. [23] designed a hybrid wall system integrating thermal energy storage, achieving peak heat flux reductions of 81.92W/m². Arumugam et al. [24] optimized PCM-insulation placement in Indian office buildings, achieving cooling load reductions of 57–64% across different climates.

This research focuses on developing efficient numerical methods for solving heat transfer equations in Cartesian, cylindrical, and spherical coordinate systems. It builds on existing numerical methods, such as Explicit Euler, Crank-Nicolson, Runge-Kutta, and the hopscotch method, to enhance their efficiency and stability. Novel methods, including the Shifted-Hopscotch, Leapfrog-Hopscotch, and Pseudo-Implicit methods, represent advancements in numerical modeling for heat transfer problems. These methods have been implemented in MATLAB, validated against analytical and experimental data, and applied to real-life engineering scenarios. The study examines heat transfer in different building walls and heated cylinders, focusing on materials like insulators, PCMs, and concrete. The goal is to optimize heat transfer management, promoting energy-efficient building designs. By improving numerical methods and their application in thermal analysis, this research contributes to sustainable building practices and energy conservation efforts.

In this study, I aim to calculate heat transfer in different geometries, beginning with deriving the general heat energy equation (conduction, convection, and radiation) based on energy balance in Cartesian, cylindrical, and spherical coordinates.

For Cartesian coordinates, energy balance on a small rectangular element during a small-time interval is expressed by the heat transfer equation, incorporating Fourier's law of heat conduction, Newton's law of heat convection, and Stefan-Boltzmann law for radiation. The governing equation includes conductive, convective, and radiative heat terms, with heat generation represented by the incoming heat from external sources like radiation and convection. The equation for temperature, including these heat transfer mechanisms, is derived.

$$\alpha\left(\frac{\partial^2 u}{\partial x^2}+\frac{\partial^2 u}{\partial y^2}+\frac{\partial^2 u}{\partial z^2}\right)+\frac{q^*}{\rho c\Delta x}+\frac{h}{\rho c\Delta x}u_a+\frac{\sigma^*}{c\rho\Delta x}u_i^4-\frac{h}{\rho c\Delta x}u-\frac{\sigma^*}{c\rho\Delta x}u^4=\frac{\partial u}{\partial t} \tag{1.1}$$

After simplification, I have

$$\frac{\partial u}{\partial t}=\frac{1}{\rho c}\nabla(k\nabla u)+q-Ku-\sigma u^4 \tag{1.2}$$

where u k, c, and ρ are the temperature, heat conductivity, specific heat and the density, respectively, $q=\frac{q^*}{c\rho\Delta x}+\frac{h}{c\rho\Delta x}\cdot u_{\mathrm{a}}+\frac{\sigma^*}{c\rho\Delta x}u_i^4$ is the heat generation or heat source coming from the outside of the wall structure, $K=K(\vec{r})=\frac{h}{c\rho\Delta x}$ is the heat transfer convection term, and $\sigma=\sigma(\vec{r})=\frac{\sigma^*}{c\rho\Delta x}$ is the radiation heat transfer term. The terms *q*, *Ku* and σu^4 are nonnegative and still in $[\mathrm{K/s}]$.

The standard central difference formula in two space dimensions is applied for the second-order derivative ($\nabla^2 u$). The space steps are Δx and Δz a

$$\frac{du_i}{dt}=\alpha\frac{u_{i\text{-}1}-2u_i+u_{i+1}}{\Delta x^2}+\alpha\frac{u_{i-Nx}-2u_i+u_{i+Nx}}{\Delta z^2}+q-Ku_i-\sigma u_i{}^4 \tag{1.3}$$

I now change from node to cell variables, which means that u_i ,c_i, and ρ_i will be the temperature, specific heat, and density of cell *i*, respectively. Furthermore, since the material boundaries will always coincide with the cell borders, I write the average $\frac{k_i+k_{i+1}}{2}$ instead of $k\left(x_i+\frac{\Delta x}{2}\right)$.

The heat capacity of the cell can be calculated as $C_i=c_i\rho_i V$. I calculate the horizontal and vertical thermal resistances between the neighbouring cells, as $R_{i,i+1}\approx\frac{\Delta x}{2k_i\Delta z}+\frac{\Delta x}{2k_{i+1}\Delta z}$, and $R_{i,i+N_x}\approx\frac{\Delta z}{2k_i\Delta x}+\frac{\Delta z}{2k_{i+N_x}\Delta x}$ respectively, where *i* and *j* represented the cells index in *x-axis* and *z-axis*. Semi-discretized form of Equation (1.3) can be expressed as below:

$$\frac{du_i}{dt}=\frac{u_{i-1}-u_i}{R_{i-1,i}C_i}+\frac{u_{i+1}-u_i}{R_{i+1,i}C_i}+\frac{u_{i-N_x}-u_i}{R_{i-N_x,i}C_i}+\frac{u_{i+N_x}-u_i}{R_{i+N_x,i}C_i}+q-Ku_i-\sigma u_i{}^4 \tag{1.4}$$

The time is discretized uniformly with time-step size Δt and represents the temperature of cell *i* at the time $n\Delta t$, $n=0,1,...,T$. Now the formulae of the used methods are presented for the general discretization (1.2) only. For the simpler formula, I need to define the following quantities:

$$mr_i=\Delta t\sum_{j\neq i}\frac{1}{C_iR_{ij}}\ ,\quad A_i=\Delta t\sum_{j\neq i}\frac{u_j^n}{C_iR_{ij}}+\Delta t\cdot q_i \text{ and } A_i^{\mathrm{new}}=\Delta t\sum_{j\neq i}\frac{u_j^{\mathrm{pred}}}{C_iR_{ij}}+\Delta t\cdot q_i$$

Where mr_i is the general mesh-ratio, while A_i reflects the state and the effect of the neighbors of cell i.We prefer to use the ODE system for a general grid, which gives the time derivative of each temperature independently of any coordinate system

$$\frac{du_i}{dt} = \sum_{i\neq j} \frac{u_j - u_i}{R_{i,j}C_i} + q_i - Ku_i - \sigma u_i{}^4 \tag{1.5}$$

Which can be written in matrix form

$$\frac{d\vec{u}}{dt} = M\vec{u} + Q\,, \tag{1.6}$$

where $Q = \vec{q}_i - K\vec{u}_i - \sigma\vec{u}_i{}^4$, and the diagonal element of matrix M can be written as follows $m_{ii} = \sum\limits_{j\in neighbour} \frac{-1}{R_{i,j}C_i}$. The off-diagonal $m_{ii} = 1/R_{i,j}C_i$ element of the M matrix can be nonzero only if the cells i and j are neighbours.

In a similar way, the heat transfer equation in cylindrical coordinates can be obtained from an energy balance on a volume element in cylindrical coordinates, considering a small 3D cylindrical element The heat convection, radiation and the change in energy of an element over a specific time interval are the same in Cartesian coordinate except the element volume being $\Delta V = \Delta\phi(r + \Delta r/2)\Delta r \times \Delta z$. In the case of full cylindrical symmetry, it is better to choose a full ring-shaped element, which yields $\Delta V = 2\pi(r + \Delta r/2)\Delta r \times \Delta z = \pi\left((r+\Delta r)^2 - r^2\right)\Delta z$.

From these equations, one can derive the heat-transport equation in a 3D cylindrical coordinate system, which can be written as:

$$\frac{1}{r}\frac{\partial}{\partial r}\left(k\,r\frac{\partial u}{\partial r}\right) + \frac{1}{r^2}\frac{\partial}{\partial\phi}\left(k\,r\frac{\partial u}{\partial\phi}\right) + \frac{\partial}{\partial z}\left(k\frac{\partial u}{\partial z}\right) + \frac{Q_{gen}}{\Delta V} - \frac{hSu}{\Delta V} - \frac{\sigma^* Su^4}{\Delta V} = \rho c\frac{\partial u}{\partial t} \tag{1.7}$$

In the case of spherical coordinates, a small 3D spherical element, the heat-transport equation for this case can be expressed as follows:

$$\frac{1}{r^2}\frac{\partial}{\partial r}\left(k\,r^2\frac{\partial u}{\partial r}\right) + \frac{1}{r^2\sin^2\theta}\frac{\partial}{\partial\phi}\left(k\,r\frac{\partial u}{\partial\phi}\right) + \frac{1}{r^2\sin\theta}\frac{\partial}{\partial\theta}\left(k\sin\theta\frac{\partial u}{\partial\theta}\right) + \frac{Q_{gen}}{\Delta V} - \frac{hSu}{\Delta V} - \frac{\sigma^* Su^4}{\Delta V} = \rho c\frac{\partial u}{\partial t} \tag{1.8}$$

The cell's heat capacity in the cylindrical and in the spherical case is approximated as $C_i = c_i \rho_i \pi \left(r_{i+1}^{*2} - r_i^{*2} \right) \Delta z$ and $C_i = c_i \rho_i \, {}^4\!/_3 \, \pi \left(r_{i+1}^{*3} - r_i^{*3} \right)$, respectively.

Let us denote the area of the cylindrical cell-surface perpendicular to *r* with S_r, which can be given as $S_r = 2\pi \, r \, \Delta z$. Now, for the thermal resistance in the *r*-direction, the approximate formula

$$R_{i,i+1} \approx \int_{r_i}^{r_{i+1}} \frac{dr}{k_{i,i+1} S_r} = \int_{r_i}^{r_{i+1}} \frac{dr}{k_{i,i+1} 2\pi r \Delta z} = \frac{\ln r_{i+1} - \ln r_i}{2\pi k_{i,i+1} \Delta z} \tag{1.9}$$

is used. For the thermal resistance in the *z*-direction, the approximate formula $R_{i,i+N_x} \approx \frac{\left(z_{i+N_r} - z_i \right)}{k_i \pi (r_{i+1}^2 - r_i^2)}$ is used, where the cell i + N_r is below the cell i.

In the spherical case, S_r can be given as $S_r = 4\pi \, r^2$. Using this, the thermal resistance is calculated similarly as that in the cylindrical case, but now the integration yields $R_{i,i+1} \approx \frac{1}{4\pi k_{i,i+1}} \frac{r_{i+1} - r_i}{r_i \, r_{i+1}}$. From Equations (1.7) and (1.8) it is easy to obtain the ODE system

$$\frac{du_i}{dt} = \sum_{j \neq i} \frac{u_j - u_i}{R_{i,j} C_i} + \frac{Q_{gen}}{C_i} - \frac{hSu_i}{C_i} - \frac{\sigma^* S u_i^4}{C_i} \tag{1.10}$$

to determine the time-evolution of the cell temperatures. Here, *S* is the area of the surface on which the convection and radiation occur, which will be the outer surface of the cylinder. If one neglects the higher powers of Δr, one can easily derive that $C_i / S = c_i \rho_i \Delta r$ in both cases. Inserting these into (1.10), It can write in a simpler form:

$$\frac{du_i}{dt} = \sum_{j \neq i} \frac{u_j - u_i}{R_{i,j} C_i} + q_i - K u_i - \sigma u_i^4 \, , \tag{1.11}$$

which will be solved numerically.

1. METHODOLOGY OF STUDY

My research focuses on designing and analyzing numerical methods for solving the heat equation, divided into three key directions. First, I developed and tested novel odd-even hopscotch-type numerical schemes. Second, I proposed adaptive time-step controllers of I-type and PI-type. Third, I designed additional adaptive controllers for cases where the diffusion coefficient varies with time and space.

2.1. Existing Numerical methods

Several well-known methods exist for solving the heat equation:

2.1.1. Explicit Euler (FTCS) [25]: Simple but conditionally stable.

$$u_i^{n+1} = \left(1 - mr_i\right) u_i^n + A_i - \Delta t \cdot K_i \cdot u_i^n - \Delta t \cdot \sigma_i \cdot (u_i^n)^4 \quad . \qquad (2.1)$$

2.1.2. Crank-Nicolson [11]: A second-order, unconditionally stable implicit method.

$$u_i^{\mathrm{n}+1} = \frac{\left(1 - {}^{mr_i}\!/_2\right) u_i^{\mathrm{n}} + A_i + 1 - \Delta t \cdot K_i \cdot u_i^n - \Delta t \cdot \sigma_i \cdot (u_i^n)^4}{1 + mr_i\left(1 - {}^1\!/_2\right)} \quad . \qquad (2.2)$$

2.1.3. UPFD [26]: Designed for diffusion-advection-reaction equations, adapted for heat transfer.

$$u_i^{n+1} = \frac{u_i^n + A_i}{1 + mr_i + \Delta t \cdot K_i + \Delta t \cdot \sigma_i \cdot (u_i^{\mathrm{n}})^3} \, . \qquad (2.3)$$

2.1.4. Dufort-Frankel (DF) [27]: A two-step explicit method, unconditionally stable but requiring a self-starting scheme.

$$u_i^{\mathrm{n}+1} = \frac{\left(1 - mr_{ii}\right) u_i^{n-1} + 2A_i - 2 \cdot \Delta t \cdot K \cdot u_i^{\mathrm{n}} - 2 \cdot \Delta t \cdot \sigma \cdot (u_i^{\mathrm{n}})^4}{1 + mr_{ii}} \qquad (2.4)$$

2.1.5. Rational Runge-Kutta [28]: Uses predictor-corrector steps to improve stability.

$$g_i^1 = mr\left(u_{i-1}^{\mathrm{n}} - 2u_i^{\mathrm{n}} + u_{i+1}^{\mathrm{n}}\right) + \Delta t \cdot q - \Delta t \cdot K \cdot u_i^{\mathrm{n}} - \Delta t \cdot \sigma \cdot (u_i^{\mathrm{n}})^4 \quad ,$$

and

$$g_i^1 = -mru_i^n + A_i - \Delta t \cdot K \cdot u_i^{\mathrm{n}} - \Delta t \cdot \sigma \cdot (u_i^{\mathrm{n}})^4 \, .$$

Using these g_i^1 values, the predictor values can be obtained for all grid types as

$$u_i^{\text{pred}} = u_i^n + g_i^1 .$$

After this, using the predictor values obtained above, the increment of a second Euler step is calculated:

$$g_i^2 = mr\left(u_{i-1}^{\text{pred}} - 2u_i^{\text{pred}} + u_{i+1}^{\text{pred}}\right) + \Delta t \cdot q - \Delta t \cdot K \cdot u_i^{\text{pred}} - \Delta t \cdot \sigma \cdot (u_i^{\text{pred}})^4 ,$$

$$g_i^2 = -mr_i u_i^{\text{pred}} + A_i^{\text{new}} - \Delta t \cdot K \cdot u_i^{\text{pred}} - \Delta t \cdot \sigma \cdot (u_i^{\text{pred}})^4 .$$

Now one needs to calculate the following scalar products

$$p_1 = \left(\vec{g}^1, \vec{g}^1\right) = \sum_{i=1}^{N} g_i^1 g_i^1 ,\;\; p_{12} = \left(\vec{g}^1, \vec{g}^2\right) = \sum_{i=1}^{N} g_i^1 g_i^2 ,\;\; p_2 = \left(\vec{g}^2, \vec{g}^2\right) = \sum_{i=1}^{N} g_i^2 g_i^2 ,$$

and with them one obtains the final expression for the new values of the variable:

$$u_i^{n+1} = u_i^n + \frac{2p_1 g_i^1 - 2p_{12} g_i^1 + p_1 g_i^2}{4p_1 - 4p_{12} + p_2} \tag{2.5}$$

2.1.6. Heun’s Method [29]: A second-order explicit scheme that refines the Euler approach.

$$u_i^{\text{pred}} = \left(1 - mr_i\right) u_i^n + A_i - \Delta t \cdot K \cdot u_i^{\text{n}} - \Delta t \cdot \sigma \cdot (u_i^{\text{n}})^4 .$$

$$u_i^{n+1} = u_i^n - mr_i \frac{u_i^n + u_i^{\text{pred}}}{2} + \frac{\Delta t}{2}\left(A_i + A_i^{\text{new}} - K \cdot \left(u_i^n + u_i^{\text{pred}}\right) - \sigma \cdot \left(u_i^n + u_i^{\text{pred}}\right)^4 \right) \tag{2.6}$$

2.1.7. Original Odd-Even Hopscotch (OOEH) [30]: The classic hopscotch method using explicit and implicit Euler schemes, with conditional corrections to prevent instability.

Explicit Euler: $$u_i^{n+1} = \left(1 - mr_i\right) u_i^n + A_i - \Delta t \cdot K_i \cdot u_i^{\text{n}} - \Delta t \cdot \sigma_i \cdot (u_i^{\text{n}})^4 \tag{2.7}$$

Implicit Euler: $$u_i^{n+1} = \frac{u_i^n + A_i^{\text{new}}}{1 + mr_i + \Delta t \cdot K_i + \Delta t \cdot \sigma_i \cdot (u_i^{\text{n}})^3} , \tag{2.8}$$

2.1.8. Reversed Odd-Even Hopscotch (ROEH): Swaps the order of explicit and implicit stages while ensuring stability using UPFD-like corrections.

2.2. Newly Developed Methods

Building on existing schemes, I introduced novel methods to improve efficiency, stability, and accuracy.

2.2.1. Constant Neighbor (CNe) Method [31]: A modified theta-scheme using known neighbor values, making it fully explicit.

$$u_i^{\text{n}+1} = \frac{\left(1 - 2mr\theta\right) u_i^n + mr\left(u_{i-1}^{n+1} + u_{i+1}^{n+1}\right) + \Delta t \cdot q - \Delta t \cdot K \cdot u_i^{\text{n}} - \Delta t \cdot \sigma \cdot (u_i^{\text{n}})^4}{1 + 2mr\left(1 - \theta\right)} \tag{2.9}$$

2.2.2. Two-Stage and Three-Stage Linear-Neighbor (LNe2, LNe3) Methods [32]: Refinements of CNe with additional correction steps for increased accuracy.

$$s_i = \frac{mr}{\Delta t^2}\left(u_{i-1}^{\text{pred}} + u_{i+1}^{\text{pred}} - u_{i-1}^{\text{n}} - u_{i+1}^{\text{n}}\right)$$

and then the corrector values for the two-stage LNe method:

$$u_i^{n+1} = u_i^n e^{-2mr} + \frac{u_{i-1}^n + u_{i+1}^n}{2}\left(1-e^{-2mr}\right) + s_i \frac{\Delta t^2}{2mr}\left(1-\frac{1-e^{-2mr}}{2mr}\right) + \Delta t\cdot q - \Delta t\cdot K\cdot u_i^{\text{n}} - \Delta t\cdot\sigma\cdot(u_i^{\text{n}})^4$$

For the general case, I can make the corrector step as follows:

$$u_{\text{i}}^{\text{n+1}} = u_{\text{i}}^{\text{n}} e^{-mr_{\text{i}}} + \left(A_{\text{i}} - \frac{A_{\text{i}}^{\text{new}} - A_{\text{i}}}{mr_{\text{i}}}\right)\frac{1-e^{-mr_{\text{i}}}}{mr_{\text{i}}} + \frac{A_{\text{i}}^{\text{new}} - A_{\text{i}}}{mr_{\text{i}}} - \Delta t\cdot K\cdot u_i^{\text{n}} - \Delta t\cdot\sigma\cdot(u_i^{\text{n}})^4 \,. \quad (2.10)$$

2.2.3. CpC Algorithm [33]: A two-stage variant of CNe with fractional time steps for improved precision.

$$u_{\text{i}}^{\text{pred}} = u_i^n e^{-mr_{\text{i}}/2} + \frac{A_i}{mr_i}\left(1-e^{-mr_{\text{i}}/2}\right) - \Delta t\cdot K\cdot u_i^{\text{n}} - \Delta t\cdot\sigma\cdot(u_i^{\text{n}})^4 \quad (2.11)$$

In the second stage, I can use $A_i^{\text{new}} = \Delta t\sum_{j\neq i}\frac{u_j^{\text{pred}}}{C_i R_{ij}}$ with Δt_1 and take a full-time step size corrector step using the CNe formula again. Thus, the final values at the end of the time step are

$$u_i^{n+1} = u_i^n\cdot e^{-mr_{\text{i}}} + \frac{A_i^{\text{new}}}{mr_{\text{i}}}\left(1-e^{-mr_{\text{i}}}\right) - \Delta t\cdot K\cdot u_i^{\text{n}} - \Delta t\cdot\sigma\cdot(u_i^{\text{n}})^4 \quad (2.12)$$

2.3. Invented Hopscotch Variants

2.3.1 Shifted-Hopscotch (SH): it is a new method invented by me, that has the repeating block consisting of five stages, two of them are half and three of them are full-time steps. The first half-sized time step is taken for the odd cells with the following general formula:

$$u_i^{n+1/2} = \frac{u_i^n + A_i + \Delta t\cdot q}{1+2mr+\Delta t\cdot K+\Delta t\cdot\sigma\cdot\left(u_i^n\right)^3} \quad (2.13)$$

Then, full-time steps are taken strictly alternately with the following formula:

$$u_i^{\mu+1} = \frac{\left(1-mr_i/2\right)u_i^{\mu} + A_i^{\mu+1/2} + \Delta t\cdot q}{1+mr+\Delta t\cdot K+\Delta t\cdot\sigma\cdot\left(u_i^{\mu}\right)^3} \quad (2.14)$$

The upper index μ is n for the even nodes and n+1 for the odd nodes. for the even, the odd, and the even cells follow again. Finally, a half-length time step for the odd cells closes the calculation with the formula

$$u_i^{n+2} = \frac{\left(1-mr_i\right)u_i^{n+1} + A_i^{n+3/2} + \Delta t\cdot q}{1+\Delta t\cdot K+\Delta t\cdot\sigma\cdot\left(u_i^{n+1}\right)^3} \quad (2.15)$$

2.3.2 Leapfrog-Hopscotch (LH): it is a new method invented by our team that has a structure consisting of two half and several full-time steps. In the first stage, the general formulas

(2.13) are used, Then, for the even and odd nodes, full-time steps are taken strictly alternately with the formulas (2.14). Always the latest available values of the neighbors must be used (for example in $A_i^{\mu+1/2}$) when the new values of *u* are calculated, regardless of the size of the time step. This alternation goes on until the end of the last timestep, where (2.14) is used again, but with a halved time step size, to reach the same final time point as the even nodes.

2.3.3 Asymmetric Hopscotch (ASH): it is also a new method that is very similar to the SH method, but contains less integer stages. The calculation starts with a half-time step size for the odd cell with (2.13). Then a full-time step is coming for the even cell with formula (2.14), and finally a half-time step size with (2.15).

2.3.4 Pseudo-Implicit (PI) Method: it is a new explicit method with advanced handling of convection and radiation terms.

$$\textit{Stage 1}:\ u_i^{\text{pred}} = \frac{u_i^n + A_i/2}{1 + mr_i + \Delta t \cdot K + \Delta t \cdot \sigma \cdot (u_i^{\text{n}})^3} \tag{2.16}$$

$$\textit{Stage 2}:\ u_i^{n+1} = \frac{(1 - mr_i)u_i^n + A_i^{\text{new}}}{1 + mr_i + \Delta t \cdot K + \Delta t \cdot \sigma \cdot (u_i^{\text{pred}})^2 \cdot u_i^{\text{n}}} \tag{2.17}$$

2.4. Optimizing Some of The Invented Methods

The new algorithms are fully explicit time-integrators obtained by a half-time step and applied different formulas in different stages. All of the algorithms consist of five stages, but they are one-step methods in the sense that when the new values of the unknown function u are calculated, only the most recently calculated u values are used, thus the methods can be implemented such that only one array of storage is required for the u variable, which means that the memory requirement is very low. I applied the conventional theta-method with 9 different values of theta and the non-conventional CNe method to construct 10^5 combinations in the case of small systems with random parameters. I examined the competitiveness of the best algorithms by testing them in case of large systems against popular solvers.

I recall that the following general time-discretization

$$\frac{u_i^{\text{n+1}} - u_i^{\text{n}}}{\Delta t} = \frac{\alpha}{\Delta x^2}\left[\theta\left(u_{i-1}^{\text{n}} - 2u_i^{\text{n}} + u_{i+1}^{\text{n}}\right) + (1-\theta)\left(u_{i-1}^{\text{n+1}} - 2u_i^{\text{n+1}} + u_{i+1}^{\text{n+1}}\right)\right],$$

leads to the so-called theta-method:

$$u_i^{\text{n+1}} = u_i^{\text{n}} + mr\left[\theta\left(u_{i-1}^{\text{n}} - 2u_i^{\text{n}} + u_{i+1}^{\text{n}}\right) + (1-\theta)\left(u_{i-1}^{\text{n+1}} - 2u_i^{\text{n+1}} + u_{i+1}^{\text{n+1}}\right)\right], \tag{2.18}$$

where $mr = \frac{\alpha \Delta t}{\Delta x^2} = -\frac{m_{ii}\Delta t}{2} > 0,\ 0 < i < N-1$ is the usual mesh ratio and $\theta \in [0,1]$.

I examine 2-dimensional rectangle-structured lattices with $N = N_{\text{x}} \times N_{\text{z}}$ cells similar to what can be seen in Fig. 1. I solve Eq. (1.4) subjected to randomly generated initial conditions $u_{\text{i}}(0) = rand$, where rand is a (pseudo)random number with a uniform distribution in the interval (0, 1), generated by the MATLAB for each cell. I also generate different random values for the heat capacities and thermal resistances but with a log-uniform distribution as follow:

$$C_{\mathrm{i}} = 10^{(\alpha_C - \beta_C \times rand)},\ R_{\mathrm{x,i}} = 10^{(\alpha_{Rx} - \beta_{Rx} \times rand)},\ R_{\mathrm{z,i}} = 10^{(\alpha_{Rz} - \beta_{Rz} \times rand)}$$

where the coefficients $\alpha_C ,...,\beta_{Rz}$ in the exponents will be concretized later.

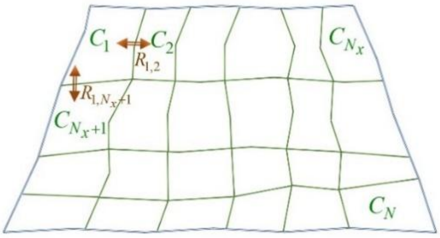

Figure 1. Arrangement of the generalized variables

I calculate the numerical error by comparing our numerical solutions $u_{\mathrm{j}}^{\mathrm{num}}$ with the reference solution $u_{\mathrm{j}}^{\mathrm{ref}}$ at final time t_{fin} . the reference solution will be an analytical solution, otherwise it is a very accurate numerical solution which has been calculated by the ode15s built-in solver of MATLAB with very strict error tolerance. I use the following three types of (global) error. The first one is the maximum of the absolute differences:

$$\mathrm{Error}(L_\infty) = \max_{0 \le \mathrm{j} \le N} \left| u_{\mathrm{j}}^{\mathrm{ref}}(t_{\mathrm{fin}}) - u_{\mathrm{j}}^{\mathrm{num}}(t_{\mathrm{fin}}) \right| . \tag{2.19}$$

The second one gives the error in terms of energy in case of the heat equation. It takes into account that an error of the solution in a cell with a large volume or heat capacity has more significance in practice than in a very small cell

$$\mathrm{Error}(Energy) = \frac{1}{N} \sum_{0 \le \mathrm{j} \le N} C_{\mathrm{j}} \left| u_{\mathrm{j}}^{\mathrm{ref}}(t_{\mathrm{fin}}) - u_{\mathrm{j}}^{\mathrm{num}}(t_{\mathrm{fin}}) \right| \tag{2.20}$$

I examine a grid with isolated boundary the sizes were fixed to $N_x = 100$ and $N_z = 100$, thus the total cell number was 10000, while the final time was $t_{fin} = 0.1$.

$$\alpha_C = 2,\ \beta_{\mathrm{C}} = 4,\ \alpha_{Rx} = \alpha_{Rz} = 1,\ \beta_{Rx} = \beta_{Rz} = 2, \tag{2.21}$$

The exponents introduced above have been set to the following values which means that log-uniformly distributed values between 0.01 and 100 have been given to the capacities. The generated system can be characterized by its stiffness ratio and $h_{\mathrm{MAX}}^{\mathrm{FTCS}}$ values, which are 3.1×10^7 and 7.3×10^{-4}, respectively. The performance of new algorithms was compared with the following widely used MATLAB solvers:

- ode15s, a first to fifth order (implicit) numerical differentiation formulas with variable-step and variable order (VSVO), developed for solving stiff problems;
- ode23s, a second order modified (implicit) Rosenbrock formula;
- ode23t, applies (implicit) trapezoidal rule with using free interpolant;

- ode23tb, combines backward differentiation formula and trapezoidal rule;
- ode45, a fourth/fifth order explicit Runge-Kutta-Dormand-Prince formula;
- ode23, second/third order explicit Runge-Kutta-Bogacki-Shampine method;
- ode113, 1 to 13 order VSVO Adams-Bashforth-Moulton numerical solver.

For all used MATLAB solvers, tolerances have been changed over many orders of magnitude, from the maximum value $\text{'AbsTol'} = \text{'RelTol'} \doteq \text{'Tol'} = 10^{3}$ to the minimum value $\text{'AbsTol'} = \text{'RelTol'} \doteq \text{'Tol'} = 10^{-5}$. I have plotted the L_∞ errors and energy errors as a function of the effective time step size Δt_{EFF}, and based on this, I selected the following top 5 combinations from those listed in (2.18) and after that:

S1 (C, C, C, C, C),

S2 (¼, ½, C, ½, ¾),

S3 (¼, ½, ½, ½, ¾),

S4 (0, ½, ½, ½, 1),

S5 (0, ½, ½, C, 1)

In Fig. 2, I present the error and energy error functions only for these top 5 combinations. Furthermore, Table 1 presents some results that were obtained by our numerical schemes and the “ode” routines of MATLAB. Notably, the results demonstrate that the best combination of the shifted-hopscotch method achieved a maximum error of 10^{-8}, an energy error of 10^{-6}, and a running time can reach 10^{-2} which represents approximately four orders of magnitude better performance compared to the ordinary and MATLAB routines.

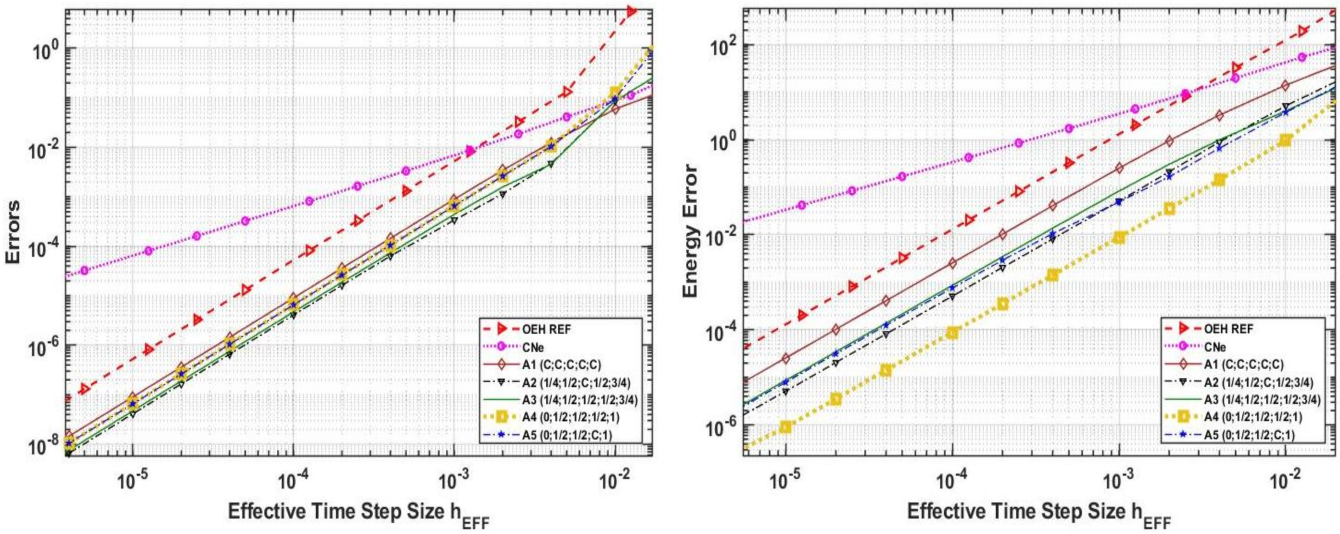

Figure 2. L_∞ errors (Left), and Energy errors (Right) as a function of the effective time step size for the first (moderately stiff) system.

Table 1. Comparison of different shifted hopscotch algorithms and MATLAB routines for the moderately stiff system of ten thousand cells.

Numerical Method	Running Time (sec)	Error(L_∞)	Error(L_1)	Energy Error
ode15s, $\text{Tol}=10^{3}$	3.97×10^{2}	1.3×10^{-2}	1.1×10^{-3}	5.62×10^{1}
ode23s, $\text{Tol}=10^{3}$	4.346×10^{3}	4.2×10^{-4}	3.0×10^{-5}	1.5×10^{-1}
ode23t, $\text{Tol}=10^{-8}$	8.49×10^{2}	2.9×10^{-7}	2.0×10^{-8}	1.0×10^{-4}
ode23tb, $\text{Tol}=10^{2}$	4.28×10^{2}	4.1×10^{-4}	2.9×10^{-5}	1.4×10^{-4}
ode45, $\text{Tol}=10^{-1}$	2.1×10^{1}	3.3×10^{-3}	6.5×10^{-5}	2.7×10^{-3}
ode23, $\text{Tol}=10^{-6}$	2.7×10^{1}	3.7×10^{-7}	9.6×10^{-9}	4.8×10^{-5}
ode113, $\text{Tol}=10^{-6}$	1.91×10^{1}	6.7×10^{-7}	4.2×10^{-10}	1.9×10^{-6}
A1, $\Delta t=1.25\times10^{-4}$	1.97×10^{-1}	9.06×10^{-6}	2.63×10^{-7}	2.56×10^{-3}
A2, $\Delta t=1.25\times10^{-3}$	2.02×10^{-2}	3.39×10^{-4}	6.93×10^{-6}	5.08×10^{-2}
A3, $\Delta t=2.5\times10^{-4}$	1.01×10^{-1}	1.88×10^{-5}	3.64×10^{-7}	3.44×10^{-3}
A4, $\Delta t=5\times10^{-4}$	5.03×10^{-2}	1.06×10^{-4}	1.07×10^{-6}	1.42×10^{-3}
A5, $\Delta t=2.5\times10^{-5}$	9.75×10^{-1}	2.62×10^{-7}	4.44×10^{-9}	3.15×10^{-5}

In a manner similar to the Shifted-Hopscotch method, the hopscotch space structure was combined with leapfrog time integration. Using the theta method with nine different values of θ, along with the recently invented CNe method. 10^5 combinations were constructed. Via subsequent numerical experiments, this huge number was decreased by excluding the combinations that underperformed and, finally, only the top five of these were retained. two-dimensional stiff systems containing 10,000 cells with completely discontinuous random parameters and initial conditions, so the results presented just for these five algorithms.

The best algorithms were compared with other methods for a large, moderately stiff system, and for a large, very stiff system. for the same system size and final time. The following top 5 combinations are chosen based on the best performance of the maximum and energy error.

L1 (C, C, C, C, C),

L2 (0, ½, ½, ½, ½),

L3 (⅕, ½, ½, ½, ½),

L4 (¼, ½, C, ½, ½),

L5 (⅕, ½, C, ½, ½).

In Fig. 3. the L_∞ errors as a function of the effective time step size are presented for the top 5 algorithms and a first-order “reference curve” for the original CNe method. I note that very similar curves have been obtained for the u_1 solution, as well as for other space and time intervals

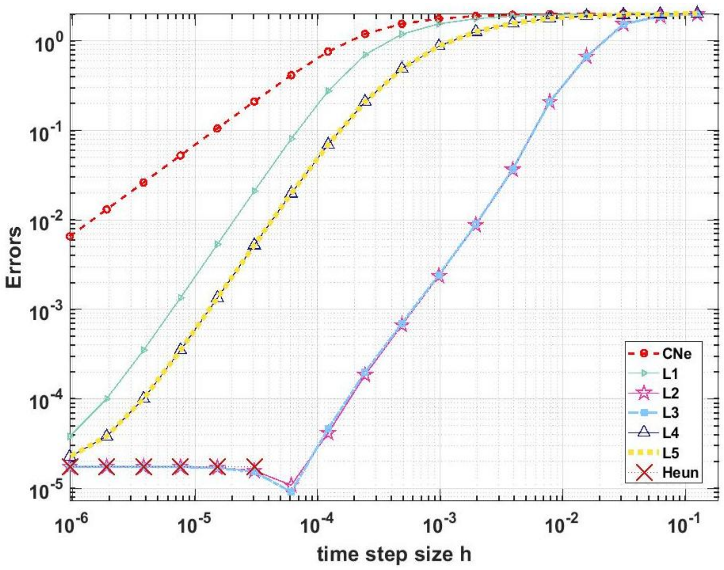

Figure 3. The L_∞ errors as a function of time step size for the space-dependent mesh

2.5. Using Efficient Methods to Solve Real-Life Heat Transfer Problems

2.5.1. Calculate The Heat Transfer in an Insulated Wall with Thermal Bridging

I examined 14 numerical methods (ExpE, NS-ExpE, Heun, UPFD, DF, NS-DF, RRK, PI, OOEH, NS-OEH, ROEH, LH, SH, and ASH) to solve the heat equation (1.4) inside building walls As one can see in Fig. 4. I considered heat conduction, convection, and radiation, in addition to heat generation. Five of the used methods are recently invented explicit algorithms that are unconditionally stable for conducting problems.

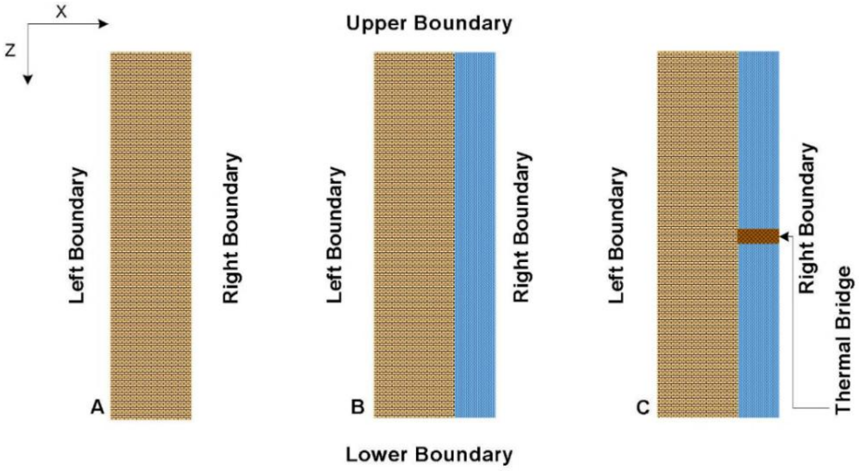

Figure 4. (**A**) One-layer wall, (**B**) wall with insulator, and (**C**) wall with insulator and thermal bridge.

I generate the mesh of the current geometries. In the cross-section case, the left 50% of the cells are always brick and the right 50% are insulator for programming simplicity. It implies that the volume of the brick and the insulator is the same in the equidistant case. However, if I have a gradual change in the *x*-direction, the thickness of the insulator is smaller (0.269m). The thermal bridge has the same thickness as the insulator in the *x* direction, thus the horizontal position of the bridge is from *x*=0.5m to *x*=1m for equidistant and from *x*=0.731m to *x*=1m for the non-equidistant mesh. The height of the bridge is one cell (1cm) in the *z* direction, i.e., 0.01m, while it is positioned in row number 50 from *z*=0.49m to *z*=0.50m.

In the present study, real material properties are taken into account. As they are listed in Table 2.

Table 2. The properties of the materials used [34].

	$\rho\left(\mathrm{kg}\cdot\mathrm{m}^{-3}\right)$	$k\ \left(\mathrm{W}\cdot\mathrm{m}^{-1}\cdot\mathrm{K}^{-1}\right)$	$c\ \left(\mathrm{J}\cdot\mathrm{kg}^{-1}\cdot\mathrm{K}^{-1}\right)$
Brick	1600	0.73	800
Glass wool	200	0.03	800
Steel structure	7800	16.2	840

I use zero Neumann boundary conditions in all cases for all boundaries, which forbids conductive heat transfer at the boundaries:

$$\frac{\partial u}{\partial x}(x,z=0,t)=\frac{\partial u}{\partial x}(x,z=1,t)=\frac{\partial u}{\partial z}(x,z=0,t)=\frac{\partial u}{\partial z}(x,z=1,t)=0\ .$$

This is implemented by setting zero for the matrix elements describing heat conduction through the boundary via the setting of the appropriate resistances to infinity.

The interior elements cannot gain or lose heat by the heat source, heat convection, or radiation. Elements on the right and left sides can transfer heat by radiation and convection to the *x* direction with the values shown in Table 3.

Table 3. The heat source, convection, and radiation parameters on both sides of wall elements in case of cross-sectional area.

	$h_c\ \left(\frac{\mathrm{W}}{\mathrm{m}^2\cdot\mathrm{K}}\right)$	$\sigma^*\ \left(\frac{\mathrm{W}}{\mathrm{m}^2\cdot\mathrm{K}^4}\times10^{-8}\right)$	$q^*\ (\mathrm{W})$
Right Elements	2	5	500
Left Elements	4	4	500

I supposed that the right elements and left elements have the following heat source convection and radiation as follows:

- For the left elements (interior side): $q=\frac{1}{c\rho}\times500\frac{W}{m^2}+\frac{h_c}{c\rho\cdot\Delta x}\times293\mathrm{K}$
- For the right elements (external side): $q=\frac{1}{c\rho}\times500\frac{W}{m^2}+\frac{h_c}{c\rho\cdot\Delta x}\times303\mathrm{K}$

The initial condition is again a linear function of the *z* variable:

$$u\left(x,z,t=0\right)=303-288z\ .$$

The maximum errors are plotted for equidistant and non-equidistant meshes in Fig.5. while in Fig. 6, the temperature contour is presented for the initial and the final time moments, for the equidistant mesh.

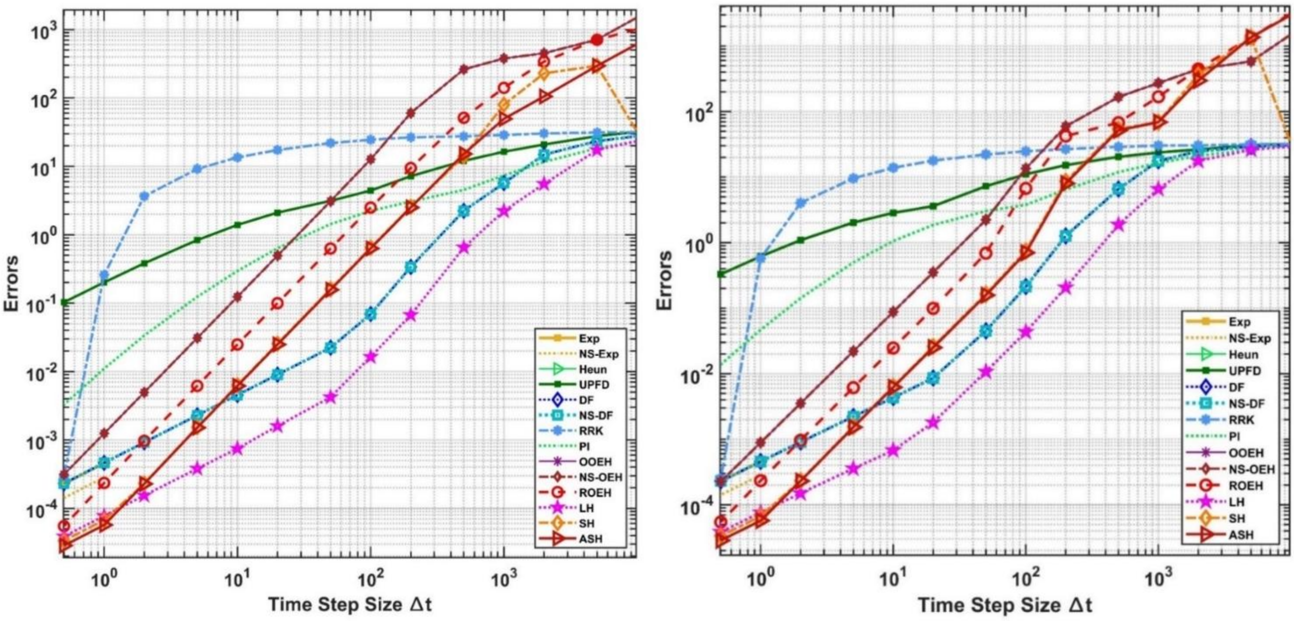

Figure 5. The maximum errors as a function of the time step size for the equidistant (Left), and non-equidistance (Right) mesh.

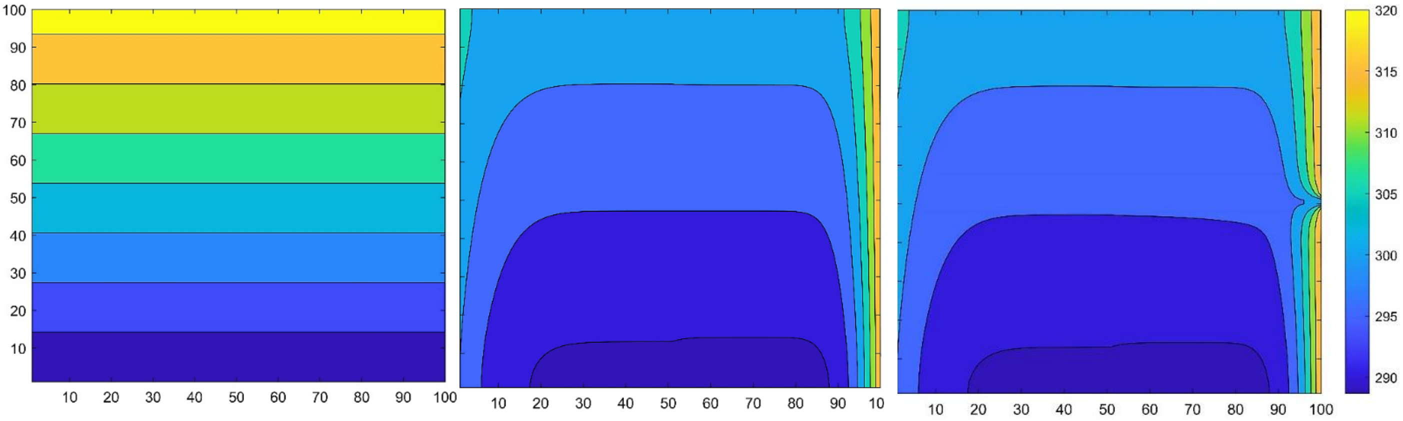

Figure 6. The temperature contour at the initial time (left), the final time for the insulator wall (middle), and the final time with insulator and thermal bridging(right)

2.5.2. Calculate The Heat Transfer in Cylindrical and Spherical Shaped Bodies

I am going to reproduce the experimental results of Cabezas et al. [35], where heat transfer was studied in a steel C45 cylinder of 168 mm total height with properties shown in Table 4. below.

Table 4. The properties of the steel used [35].

Material	$\rho\left(\mathrm{kg}\cdot\mathrm{m}^{-3}\right)$	$k\ \left(\mathrm{W}\cdot\mathrm{m}^{-1}\cdot\mathrm{K}^{-1}\right)$	$c\ \left(\mathrm{J}\cdot\mathrm{kg}^{-1}\cdot\mathrm{K}^{-1}\right)$
Steel C45	7800	40	480

The bottom of the cylinder was heated for 30 s at the beginning of the experiment with P = 1500W power. However, in the original work [35], the position of the lowest thermocouple was 50mm higher than the heated surface. The top 118 mm and not the bottom 50 mm of the cylinder was examined either experimentally or numerically, and I followed this in my work. This means that the simulated volume of the cylinder segment is $V = 1.0087\times10^{-4}\ \mathrm{m}^3$, while $(r,z)\in[0, 0.0165\,\mathrm{m}]\times[0, 0.118\,\mathrm{m}]$. In my approximation, physical quantities did not change in the ϕ-direction, thus, that 3D was irrelevant and, computationally, I dealt with a two-dimensional problem. The number of the cells along the *r* axis and *z* axis were set to N_r = 15 and N_z = 100; thus, the total number of the cells in the system was $N = N_r N_z = 1500$.

I used a constant initial condition in all cases.

$$u\left(r, z, t=0\right) = 30.7\ {}^\circ\mathrm{C}$$

I used different boundary conditions on different sides. On the left side, the center of the cylinder, I applied Neumann boundary conditions in all cases, which do not allow conductive heat transfer at the boundary

$$u_r(r=0,z,t) = u_r(r=L_r,z,t) = u_z(r,z=L_z,t) = 0\ .$$

On the right (external) and upper boundaries, there was a heat exchange with the environment via convection and radiation, considering the heat convection coefficient $h = 4.5\ \left(\mathrm{W}\cdot\mathrm{m}^{-2}\cdot\mathrm{K}^{-1}\right)$[38] and the emissivity constant as 0.85 to obtain realistic values for σ^*. The convective and radiative energy transfer was perpendicular to the surface. The interior elements cannot gain or lose heat by the heat source, heat convection, or radiation. On the lower boundary, I applied changing Dirichlet boundary conditions based on the temperature measurement results taken from the experimental report.

The methods verified in compare with the analytical solution, while I take the height of the cylinder as well as Δz unity. It means that, computationally, there is one space dimension only in both the cylindrical and the spherical case. The solution parameters are:

$$N_r = 500,\, N_z = 1,\, N = N_r \times N_z = 500,\ r_0 = 0.0003,\ r_{max} = 0.999,\ \Delta r = 0.002,\, \alpha = 1,$$
$$a \in \{1, 1.2,\ 2\},\, t^0 = 0.1,\ t^{\mathrm{fin}} = t^0 + 0.1.$$

Here, *N* represents the total number of cells, *a* self-similar exponent, while r_0 and r_{max} are the radial coordinates of the center of the first and last cells. The obtained maximum errors are displayed as a function of the time-step size in Fig.7. for the cylindrical and spherical coordinates. The fact that we obtained very small errors in all cases verifies not only the numerical algorithms but the equivalence of the two mathematical treatments of the physical problem.

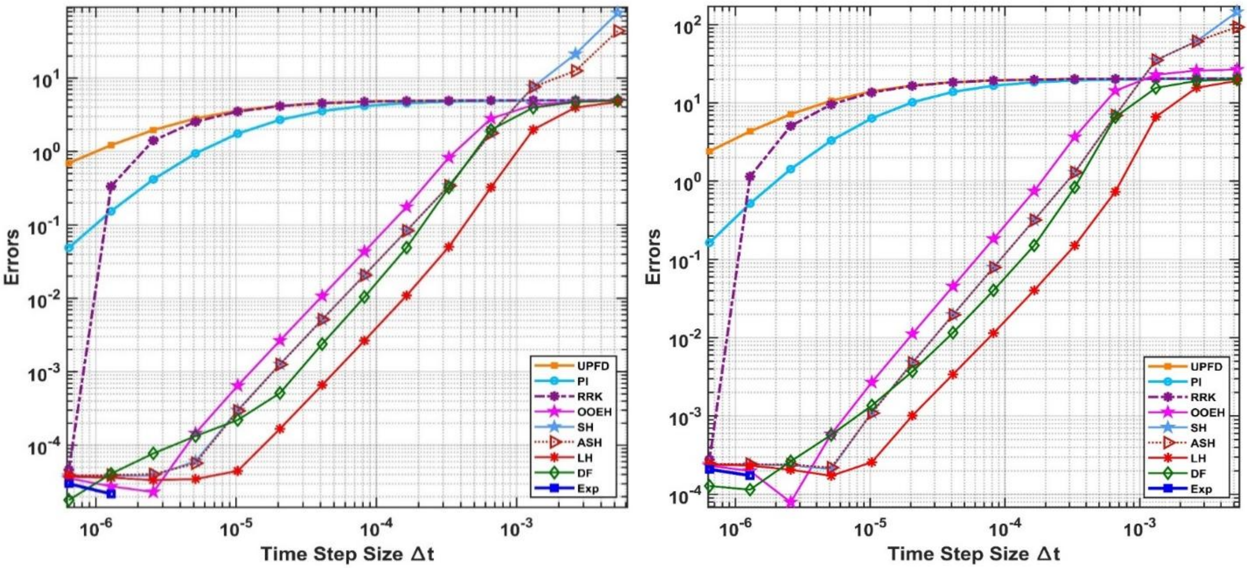

Figure 7. The maximum errors as a function of the time step size for the 9 numerical methods in case of cylindrical coordinates with *a=1* (Left), and in case of spherical coordinates with *a* = 1.2.(Right)

I present the results at the end of the examined time interval, which is defined as $t_{\text{fin}} = 1200, 1440 \text{ and } 1800 \text{s}$. I chose the top five algorithms, namely DF, OOEH, LH, SH, and ASH. The simulation of a steel C45 cylinder was conducted using these selected algorithms considering different boundary conditions, as previously mentioned. Among these algorithms, the shifted-hopscotch method was chosen to visualize the temperature contour due to its high accuracy at small time-step size. Fig. 8 displays the final temperature distribution obtained from this method.

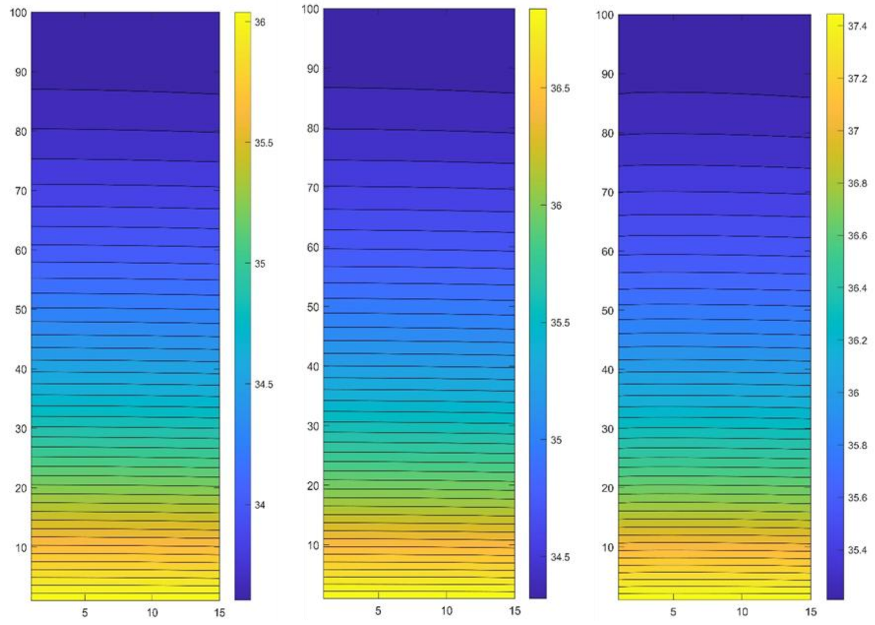

Figure 8. The temperature distribution contour for different time values (*t* = 20, 24, and 30 min) is presented by the SH method.

2. THESES – NEW SCIENTIFIC RESULTS

T1. I constructed and tested the Shifted-Hopscotch algorithms, which were fully explicit time-integrators obtained by applying half-time steps and full-time steps in the odd-even hopscotch structure. I applied the conventional theta method with 9 different values, and the non-conventional CNe method to construct 10^5 combinations and I chose the top five of them via numerical experiments. These experiments suggest that the proposed methods are, indeed, competitive, as they can give fairly accurate results orders of magnitude faster than the well-optimized MATLAB routines or the Crank–Nicolson method, and they are also significantly more accurate for stiff systems than the UPFD, the Dufort–Frankel, or the original odd-even hopscotch method. If high accuracy is required, the S4 (0, ½, ½, ½, 1) combination can be proposed; however, when preserving positivity is crucial, the S1 (C, C, C, C, C) algorithm should be used .

T2. To demonstrate the practical utility of these advanced numerical techniques, I investigated 13 algorithms to solve the problem of linear heat conduction in building walls. These included eight explicit, unconditionally stable algorithms invented by our research group, such as the Shifted-Hopscotch (SH) scheme. The validation process, where numerical results were compared against analytical solutions using both uniform and non-uniform spatial discretizations, was carried out as a team. Then, I applied carefully designed nontrivial boundary conditions: spatially varying temperatures on the brick side and time-dependent temperatures on the outer surface of the insulation. I found that the classic Odd-Even Hopscotch (OEH) method delivers superior accuracy for homogeneous scenarios, while the Leapfrog-Hopscotch (LH) algorithm performs best in non-uniform configurations. Nevertheless, the Shifted-Hopscotch (SH) method also exhibited strong competitiveness across all test cases.

T3. I also examined 11 of the new methods to solve heat conduction, convection, radiation, and heat generation inside building walls' elements. These methods were tested on real-life applications involving surface area (one-layer brick) and cross-sectional area (two-layer brick and insulator) walls, with and without thermal bridging, to determine accuracy dependence on material properties, mesh type, and time step size. Neumann boundary conditions were applied to all boundaries, for surface area cases, the heat source, convection, and radiation inside all elements were considered, while for cross-sectional area cases only the right and left boundary elements containing heat source, convection, and radiation. The results indicate that the Original Odd-Even Hopscotch method is usually the best for uniform cases, while the Leapfrog-Hopscotch algorithm performs best for non-uniform cases.

T4. In addition to Cartesian coordinates, I developed 9 of the new methods to solve heat transfer problems in cylindrical and spherical geometries. I reproduced novel and nontrivial analytical solutions for the heat-conduction PDE with high accuracy. Furthermore, I verified the numerical methods in cylindrical and spherical coordinates, incorporating convection and radiation terms, by reproducing real experimental data of a heated cylinder and comparing it with Finite Element Methods (FEM) ANSYS workbench. Convection and nonlinear radiation were considered on the boundary of the cylinder. Verification results demonstrated the high accuracy of the numerical methods in dealing with cylindrical and spherical bodies. Additionally, temperature comparisons across all approaches revealed that explicit methods are more accurate than finite element software in all cases, with the Leapfrog-Hopscotch algorithm typically being the most accurate among the studied methods.

T5. I investigated the heat transfer through building walls, considering different wall geometries and heat load scenarios, encompassing both cooling and heating. My objective was to analyze how heat transfer depends on the wall materials and evaluate algorithm performance in cases involving heat transfer between solid surfaces and fluid (convection) on the outdoor surface, particularly across an air gap between the insulation and Photovoltaic Cells (PVC). The results of the study reveal that insulation prevents heat from entering the building, maintaining a comfortable indoor environment. Forced convection significantly enhances heat dissipation, especially during cooling operations to protect PVC with limited working temperature. Furthermore, the simulations highlight the air gap’s efficiency in cooling PVC and reducing maximum temperatures on the insulation’s outer surface, especially under forced convection conditions. The test results show that the Leapfrog Hopscotch algorithm offers the best solution for this highly stiff system, followed by the Asymmetric and Shifted-Hopscotch algorithms.

T6. I also simulated a multilayer wall integrated with PCMs using an effective heat capacity model and I employed the Leapfrog-Hopscotch methods for that. I validated my approach against established mathematical expressions and models in the literature, investigating various building wall geometries, two types of PCMs used in this investigation, and boundary conditions. The objective was to maintain interior temperatures within comfort zones. Regardless of the wall material, whether brick or concrete, my simulations consistently demonstrated the PCM’s effectiveness in minimizing heat transfer into indoor environment.

4. LIST OF PUBLICATIONS RELATED TO THE TOPIC OF THE RESEARCH FIELD

(1) Á. Nagy, M. Saleh, I. Omle, H. Kareem, and E. Kovács, “New stable, explicit, shifted-hopscotch algorithms for the heat equation,” Math. Comput. Appl., vol. 26, no. 61, 2021.

(2) Á. Nagy, I. Omle, H. Kareem, E. Kovács, I. F. Barna, and G. Bognar, “Stable, Explicit, Leapfrog-Hopscotch Algorithms for the Diffusion Equation,” Computation, vol. 9, no. 8, p. 92, 2021.

(3) H. K. Jalghaf, E. Kovács, J. Majár, Á. Nagy, and A. H. Askar, “Explicit stable finite difference methods for diffusion-reaction type equations,” Mathematics, vol. 9, no. 24, 2021, doi: 10.3390/math9243308.

(4) Kareem, H.; Omle, I.; Kovács, E. A Comparative Study of Explicit and Stable Time Integration Schemes for Heat Conduction in an Insulated Wall. Buildings 2022, 12. https://doi.org/10.3390/buildings12060824.

(5) H. K. Jalghaf, E. Kovács, and B. Bolló, “Comparison of Old and New Stable Explicit Methods for Heat Conduction, Convection, and Radiation in an Insulated Wall with Thermal Bridging,” Buildings, vol. 12, no. 9, pp. 1–26, 2022, doi: 10.3390/buildings12091365.

(6) H. K. Jalghaf, E. Kovács, I. F. Barna, and L. Mátyás, “ Analytical Solution and Numerical Simulation of Heat Transfer in Cylindrical and Spherical Shaped Bodies”, Computation 2023, 11, 131. https://doi.org/10.3390/ computation11070131.

(7) H. K. Jalghaf, E. Kovács, and B. Bolló, “Simulation of transient heat transfer in multilayer walls with photovoltaic cell and air by using efficient numerical methods”, Elsevier, Results Eng. 2023, https://doi.org/10.1016/j.rineng.2023.101715.

(8) H. K. Jalghaf, and E. Kovács, “ Simulation of phase change materials in building walls using effective heat capacity model by recent numerical methods”, Elsevier, Journal of Energy Storage, https://doi.org/10.1016/j.est.2024.110669.

(9) M.S. Alwan, H. K. Jalghaf, and E. Kovács, “ Numerical Study of Integrating the Phase Change Material with Building Envelop for Improved Indoor Thermal Comfort ”, Al-Iraqia Journal for Scientific Engineering Research, http://doi.org/10.58564/IJSER.3.3.2024.235.

(10) M.S. Alwan, and H. K. Jalghaf, “ Long Term Heat Transfer Simulation Through multilayer building wells with Phase Change Materials”, Al-Iraqia Journal for Scientific Engineering Research, http://doi.org/10.58564/IJSER.3.3.2024.226.

5. REFERENCES

[1] Zoppou, C.; Knight, J.H. Analytical solution of a spatially variable coefficient advection-diffusion equation in up to three dimensions. Appl. Math. Model. 1999, 23, 667–685, doi:10.1016/S0307-904X(99)00005-0.

[2] Cusini, M. Dynamic Multilevel Methods for Simulation of Multiphase Flow in Heterogeneous Porous Media; Delft University of Technology: Delft, The Netherlands, 2019.

[3] Gordon, P. Nonsymmetric Difference Equations. J. Soc. Ind. Appl. Math. 1965, 13, 667–673, doi:10.1137/0113044.

[4] Gourlay, A.R. Hopscotch: A Fast Second-order Partial Differential Equation Solver. IMA J. Appl. Math. 1970, 6, 375–390.

[5] Gourlay, A.R.; McGuire, G.R. General Hopscotch Algorithm for the Numerical Solution of Partial Differential Equations. IMA J. Appl. Math. 1971, 7, 216–227.

[6] Gourlay, A.R. Some recent methods for the numerical solution of time-dependent partial differential equations. Proc. R. Soc. London. A Math. Phys. Sci. 1971, 323, 219–235, doi:10.1098/rspa.1971.0099.

[7] Harley, C. Hopscotch method: The numerical solution of the Frank-Kamenetskii partial differential equation. Appl. Math. Comput. 2010, 217, 4065–4075, doi:10.1016/j.amc.2010.10.020.

[8] Al-Bayati, A.; Manaa, S.; Al-Rozbayani, A. Comparison of Finite Difference Solution Methods for Reaction Diffusion System in Two Dimensions. AL-Rafidain J. Comput. Sci. Math. 2011, 8, 21–36, doi:10.33899/csmj.2011.163605.

[9] Xu, J.; Shao, S.; Tang, H. Numerical methods for nonlinear Dirac equation. J. Comput. Phys. 2013, 245, 131–149, doi:10.1016/j.jcp.2013.03.031.

[10] de Goede, E.D.; ten Thije Boonkkamp, J.H.M. Vectorization of the Odd–Even Hopscotch Scheme and the Alternating Direction Implicit Scheme for the Two-Dimensional Burgers Equations. SIAM J. Sci. Stat. Comput. 1990, 11, 354–367, doi:10.1137/0911021.

[11] Maritim, S.; Rotich, J.K.; Bitok, J.K. Hybrid hopscotch Crank-Nicholson-Du Fort and Frankel (HP-CN-DF) method for solving two dimensional system of Burgers' equation. Appl. Math. Sci. 2018, 12, 935–949, doi:10.12988/ams.2018.8798.

[12] Maritim, S.; Rotich, J.K. Hybrid Hopscotch Method for Solving Two Dimensional System of Burgers' Equation. Int. J. Sci. Res. 2018, 8, 492–497.

[13] Saleh, M.; Nagy, Á.; Kovács, E. Construction and investigation of new numerical algorithms for the heat equation: Part 1. Multidiszcip. Tudományok 2020, 10, 323–338, doi:10.35925/j.multi.2020.4.36.

[14] Saleh, M.; Nagy, Á.; Kovács, E. Construction and investigation of new numerical algorithms for the heat equation: Part 2. Multidiszcip. Tudományok 2020, 10, 339–348, doi:10.35925/j.multi.2020.4.37.

[15] Saleh, M.; Nagy, Á.; Kovács, E. Construction and investigation of new numerical algorithms for the heat equation: Part 3. Multidiszcip. Tudományok 2020, 10, 349–360, doi:10.35925/j.multi.2020.4.38.

[16] X. Geng, J. Wang, Y. Gao, and X. Meng, "Location combination optimization of thermal insulation material and phase-change material in multi-layer walls under air-conditioning continuous and intermittent operation," *J. Energy Storage*, vol. 44, p. 103449, Dec. 2021, doi: 10.1016/J.EST.2021.103449.

[17] Z. Liu, J. Hou, D. Wei, X. Meng, and B. J. Dewancker, "Thermal performance analysis of lightweight building walls in different directions integrated with phase change materials (PCM)," *Case Stud. Therm. Eng.*, vol. 40, p. 102536, Dec. 2022, doi: 10.1016/J.CSITE.2022.102536.

[18] E. Tunçbilek, M. Arıcı, M. Krajčík, D. Li, S. Nižetić, and A. M. Papadopoulos, "Enhancing building wall thermal performance with phase change material and insulation: A comparative and synergistic assessment," *Renew. Energy*, vol. 218, p. 119270, Dec. 2023, doi: 10.1016/J.RENENE.2023.119270.

[19] Y. Cascone, A. Capozzoli, and M. Perino, "Optimisation analysis of PCM-enhanced opaque building envelope components for the energy retrofitting of office buildings in Mediterranean climates," *Appl. Energy*, vol. 211, pp. 929–953, Feb. 2018, doi: 10.1016/J.APENERGY.2017.11.081.

[20] R. F. Jam, M. Gholizadeh, M. Deymi-Dashtebayaz, and E. Tayyeban, "Determining the optimal location and thickness of phase change materials in the building walls: an energy-economic analysis," *J. Brazilian Soc. Mech. Sci. Eng.*, vol. 45, no. 10, p. 554, 2023, doi: 10.1007/s40430-023-04472-8.

[21] M. J. Abden, Z. Tao, M. A. Alim, Z. Pan, L. George, and R. Wuhrer, "Combined use of phase change material and thermal insulation to improve energy efficiency of residential buildings," *J. Energy Storage*, vol. 56, p. 105880, Dec. 2022, doi: 10.1016/J.EST.2022.105880.

[22] E. Iffa, D. Hun, M. Salonvaara, S. Shrestha, and M. Lapsa, "Performance evaluation of a dynamic wall integrated with active insulation and thermal energy storage systems," *J. Energy Storage*, vol. 46, p. 103815, Feb. 2022, doi: 10.1016/J.EST.2021.103815.

[23] P. Arumugam, V. Ramalingam, and P. Vellaichamy, "Optimal positioning of phase change material and insulation through numerical investigations to reduce cooling loads in office buildings," *J. Energy Storage*, vol. 52, p. 104946, Aug. 2022, doi: 10.1016/J.EST.2022.104946.

[24] Suárez-Carreño, F.; Rosales-Romero, L. Convergency and stability of explicit and implicit schemes in the simulation of the heat equation. Appl. Sci. 2021, 11, 4468. https://doi.org/10.3390/app11104468.

[25] Chen-Charpentier, B.M.; and Kojouharov, H. V. "An unconditionally positivity preserving scheme for advection-diffusion reaction equations," Math. Comput. Model., vol. 57, pp. 2177–2185, 2013, doi: 10.1016/j.mcm.2011.05.005.

[26] C. Hirsch, Numerical computation of internal and external flows, volume 1: Fundamentals of numerical discretization. Wiley, 1988.

[27] G. Sottas, “Rational Runge-Kutta methods are not suitable for stiff systems of ODEs,” J. Comput. Appl. Math., vol. 10, pp. 169–174, 1984.

[28] A. H. Workie, “New Modification on Heun’s Method Based on Contraharmonic Mean for Solving Initial Value Problems with High Efficiency,” J. Math., vol. 2020, 2020, doi: 10.1155/2020/6650855.

[29] A. R. Gourlay and G. R. McGuire, “General Hopscotch Algorithm for the Numerical Solution of Partial Differential Equations,” IMA J. Appl. Math., vol. 7, no. 2, pp. 216–227, 1971.

[30] Kovács, E. A class of new stable, explicit methods to solve the non-stationary heat equation. Numer. Methods Partial Differ. Equ. 2020, 37, 2469–2489. https://doi.org/10.1002/num.22730.

[31] Holmes, M.H. Introduction to Numerical Methods in Differential Equations; Springer: New York, NY, USA, 2007; ISBN978-0387-30891-3.

[32] Kovács, E.; Nagy, Á.; Saleh, M. A set of new stable, explicit, second order schemes for the non-stationary heat conduction equation. Mathematics 2021, 9, 2284. Available online: https://www.mdpi.com/2227-7390/9/18/2284 (accessed on 3 November 2021).

[33] Á. Nagy, M. Saleh, I. Omle, H. Kareem, and E. Kovács, “New stable, explicit, shifted-hopscotch algorithms for the heat equation,” Math. Comput. Appl., vol. 26, no. 61, 2021, [Online]. Available: https://www.mdpi.com/2297-8747/26/3/61/htm

[34] Á. Nagy, I. Omle, H. Kareem, E. Kovács, I. F. Barna, and G. Bognar, “Stable, Explicit, Leapfrog-Hopscotch Algorithms for the Diffusion Equation,” Computation, vol. 9, no. 8, p. 92, 2021.

[35] H. K. Jalghaf, E. Kovács, J. Majár, Á. Nagy, and A. H. Askar, “Explicit stable finite difference methods for diffusion-reaction type equations,” Mathematics, vol. 9, no. 24, 2021, doi: 10.3390/math9243308.

[36] A. Iserles, *A First Course in the Numerical Analysis of Differential Equations*. Cambridge: Cambridge Univ. Press, 2009. ISBN9788490225370.

[37] Cabezas, S.; Hegedűs, G.; Bencs, P. Thermal experimental and numerical heat transfer analysis of a solid cylinder in longitudinal direction. Analecta Tech. Szeged. 2023, 17, 16–27. https://doi.org/10.14232/analecta.2023.1.16-27.