UNIVERSITY OF MISKOLC
FACULTY OF MECHANICAL ENGINEERING AND INFORMATICS

ELABORATING AND OPTIMIZING METHODS TO INVESTIGATE HEAT TRANSFER PROBLEMS IN BUILDING COMPONENTS

Booklet of PhD Theses

PREPARED BY:

Issa Omle

Engineering of Mechanical and Electrical (BSc),
Engineering of Mechanical and Manufacturing (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
2024

JUDGING COMMITTEE

Chair:

Secretary:

Members:

OFFICIAL REVIEWERS

1. INTRODUCTION

There are lots of numerical methods to solve the heat conduction equation, such as several finite difference schemes (FDM) [1], finite element methods (FEM) [2], or a combination of these [3]. However, they can be computationally demanding since they require the full spatial discretization of the examined system which converts the Partial Differential Equation (PDEs) into a system of Ordinary Differential Equation (ODEs). After that one can solve the system of ODEs at each time level [4]. If the eigenvalues of the problem have a range of several orders of magnitude due to differences in material properties, then the problem is rather stiff and the so-called Courant–Friedrichs–Lewy (CFL) limit can be very small. When the tolerance is not enough small, instability can occur even in professional commercial adaptive time-step size solvers like MATLAB's ode23 and ode45 [5]. This means that almost all explicit finite difference methods are unstable when the time step size is larger than this small threshold. On the other hand, implicit methods work with whole system matrix, thus they can be extremely slow with huge memory usage when the number of cells is large. Still, these methods are used for solving these kinds of equations, [6].

The main issue with the implicit methods is that they cannot be easily parallelized because it requires the solution of an algebraic equation system at each time step. However, when the implicit methods are employed to handle more complex systems, the numerical calculations can be time-consuming. One can observe that the trend toward increasing parallelism in high-performance computing is reinforced, since unfortunately the CPU clock frequencies nowadays increase much slower than a few decades ago [7], [8]. That is one of the reasons why I believe that the importance of easily parallelizable explicit and unconditionally stable methods is going to increase, even if currently not too many scholars work with them (see [9], [10]).

The second problem with most explicit or implicit methods is that they can produce qualitatively unacceptable solutions, such as unphysical oscillations or negative values for variables that would otherwise be non-negative. We explained in our previous investigations [11], [12], that the widely used conventional solvers, either explicit or implicit, have serious difficulties. This information highlights the fact that finding effective numerical methods is still important.

One of the very few easily parallelizable explicit and unconditionally stable methods is the odd-even hopscotch (OEH) algorithm [13], [14]. In the works [11], [13], we showed that this method is robust and powerful for spatially homogeneous grids but, in the case of stiff systems, it can be disastrously inaccurate for large time step sizes. I constructed new hopscotch combinations and found [11] that some of them behave much better, not only for large, but also for medium and small-time step sizes. I extend my research by further modifying the underlying space and time structure. Our research group developed several explicit unconditionally stable methods to calculate heat conduction in arbitrary space dimensions [11]. Unconditional stability here means that the temperature remains finite for arbitrary time step size. These methods either belong to the family of FDMs or are like them. In our original papers, we tested the algorithms under general cases using discontinuous random parameters and initial conditions and using analytical as well as numerical reference solutions, and I have shown that they can provide quite accurate results, and they are much faster than the professionally optimized MATLAB 'ode' routines.

In my research, I worked with my supervisor on investigating and improving families of novel and conventional explicit methods for solving linear and nonlinear heat conduction equations, depending on a new way of thinking. Adapt the most successful methods (especially the Leapfrog and the original hopscotch methods) to cases where there is heat transfer by convection and (Stefan-Boltzmann-type, thus nonlinear) radiation, especially the problems of real-life heat transfer in buildings. In addition, I compare my results and running times with those of the appropriate software, e.g., ANSYS. I investigated the explicit type of all methods. After the heat conduction equation was spatially discretized, they were applied to it. The phenomenon of the simplest Fourier-type heat conduction within a homogeneous medium with a heat source is described using a parabolic PDE as follows:

$$\frac{\partial u}{\partial t} = \alpha \nabla^2 u + q \tag{1.1}$$

The law of Newton's cooling states that the term $K(u_a - u)$ indicates to the convective heat transfer that occurs between a moving fluid and a surface [15]. This calls for the term Ku_a to be included in the equation q. The Stefan-Boltzmann equation [16] states that the radiative heat loss from a surface may be expressed using the term $-\sigma u^4$, when the surface area and the Stefan-Boltzmann constant, both of which are positive, yield the proportionality constant. Like the Ku_a term stated before, the incoming radiation, may be included into the source term q. The terms for convection, radiation, and the heat source are added to the heat conduction Eq.(1.1), we obtain:

$$\frac{\partial u}{\partial t} = \alpha \nabla^2 u + q - Ku - \sigma u^4 \tag{1.2}$$

Note that all terms in Eq.(1.2) are local, except the conduction term. In the case of Eq.(1.1) in one space dimension, I apply to the $\alpha \nabla^2 u$ term the most common central difference equation

$$\frac{\partial^2}{\partial x^2} u(x_i) \approx \frac{\frac{u(x_{i+1}) - u(x_i)}{\Delta x} + \frac{u(x_{i-1}) - u(x_i)}{\Delta x}}{\Delta x} = \frac{u_{i-1} - 2u_i + u_{i+1}}{\Delta x^2} \tag{1.3}$$

which is second order in Δx, where $i = 1,...,N$ and N is the overall number of nodes.

By doing this, we can derive the spatially discretized version of the heat transfer Eq.(1.2) in one space dimension as follows:

$$\frac{du_i}{dt} = \alpha \frac{u_{i-1} - 2u_i + u_{i+1}}{\Delta x^2} + q - Ku_i - \sigma u_i^4 \tag{1.4}$$

Let us now present the discretization of the heat transfer equation assuming that the quantities describing the properties of materials, namely α, k, c, and ρ, are functions of the space, rather than a fixed value. Now in one space dimension, instead of the $\alpha\nabla^2 u$ term, we must deal with:

$$\frac{1}{c(x)\rho(x)}\frac{\partial}{\partial x}\left(k(x)\frac{\partial u}{\partial x}\right) \tag{1.5}$$

In this case, the heat conduction equation can be discretized as follows:

$$c(x_i)\rho(x_i)\left.\frac{\partial u}{\partial t}\right|_{x_i} = \frac{1}{\Delta x}\left[k\left(x_i+\frac{\Delta x}{2}\right)\frac{u(x_i+\Delta x)-u(x_i)}{\Delta x}+k\left(x_i-\frac{\Delta x}{2}\right)\frac{u(x_i-\Delta x)-u(x_i)}{\Delta x}\right].$$

The discretized equation attains the following form:

$$\frac{du_i}{dt}=\frac{1}{c_i\rho_i\Delta x}\left(k_{i,i+1}\frac{u_{i+1}-u_i}{\Delta x}+k_{i-1,i}\frac{u_{i-1}-u_i}{\Delta x}\right)+q-Ku_i-\sigma u_i^4 \tag{1.6}$$

The dimensions of a cell, measured along its length and across its (typical) cross-section, are represented as Δx and S. Where u_i is the temperature of the cell i, $C_i = c_i\rho_i V$ is the heat capacity of that cell in $[\mathrm{J/K}]$ units, and $V = S\Delta x$ is the volume of the cell. I introduce two other quantities, the heat source term q,

$$q_\mathrm{i} = \frac{1}{V_\mathrm{i}}\int_{V_i} q\,dV \approx q \;\text{ in }\left[\frac{K}{s}\right]\text{units,}$$

The thermal resistance between the two neighbouring nodes can be determined as $R_{i,i+1}\approx \Delta x/\left(k_{i,i+1}S\right)$ in $\left(K/W\right)$ units. The distances between the cells center in case of non-equidistant grid are $d_{\mathrm{i},i+1}=\left(\Delta x_\mathrm{i}+\Delta x_{i+1}\right)/2$ and the resistances can be determined by this approximation as $R_{i,i+1}\approx d_\mathrm{ij}/\left(k_{i,i+1}S_\mathrm{ij}\right)$. If the material properties or the sizes of the two neighboring cells are different, we can write for the resistance between cells i and i+1 that $Rx_i \approx \left[\Delta x_i/\left(k_i S_i\right)\right]+\left[\Delta x_{i+1}/\left(k_{i+1}S_{i+1}\right)\right]$, and if the cell j is below the cell i, we have $Rz_i \approx \left[\Delta x_i/\left(k_i S_i\right)\right]+\left[\Delta x_j/\left(k_j S_j\right)\right]$ for the vertical resistance.

One can obtain the following expression for the time derivative of each cell variable:

$$\frac{du_i}{dt}=\frac{u_{i-1}-u_i}{R_{i-1,i}C_i}+\frac{u_{i+1}-u_i}{R_{i+1,i}C_i}+q-Ku_i-\sigma u_i^4 \tag{1.7}$$

It is not hard to generalize Eq. (1.7) even more for the case of arbitrary number of neighbours to obtain the following spatially discretized version of Eq.(1.2):

$$\frac{du_i}{dt}=\sum_{j\neq i}\frac{u_j-u_i}{R_{i,j}C_i}+q-Ku_i-\sigma u_i^4 \tag{1.8}$$

2. METHODOLOGY OF THE STUDY

My goal is to elaborate and optimize numerical methods, and then, using these methods, to efficiently investigate heat transfer problems in building walls. These will be the tools by which I can perform optimizations of building envelopes not only from a thermodynamic but also from an economic point of view. The work can be classified into three directions. In the first direction, I increased the efficiency of some methods, by combining the hopscotch with the leapfrog technique and performing numerical experiments to investigate the performance of those methods, and choose the best combinations, and testing these algorithms on both small and large systems, and for stiff and non-stiff system. In the second direction, I tested the methods on real-life applications to examine how the performance of the individual methods changes and which of them is the best choice under different circumstances. In the third direction, I compared the performance of traditional and recent efficient numerical methods for long-term heat transfer simulations in walls with different shapes of thermal bridges, in addition to optimize thermal insulation.

2.1 Some Explicit Methods

2.1.1 The original odd-even hopscotch (OOEH)

Over half a century has passed since the discovery of the original odd–even hop-scotch (OOEH) algorithm [14]. Its temporal and spatial organization has been described in [17]. It is designed to be a quick, all-purpose algorithm that produces results with little effort from the user or the computer. This completely explicit two-stage approach has, as far as we know, undergone modification and generalization procedures to increase its accuracy, but always in the direction of implicitness. After the first step by the FTCS formula (which is based on explicit Euler time discretization) for the odd cells, the BTCS formula (which is based on implicit Euler time discretization) is used for the even cells. The labels odd and even are interchanged after each time step. If we would like to apply an odd-even hopscotch method, we need a bipartite grid, where all the nearest neighbours of the odd cells are even and vice versa as is shown in Figure 2.1. I modify this method here to include the convection component, which is always considered at the new time level for enhanced stability. The radiation term is handled first explicitly and then implicitly [18]. These are the equations that are being used:

First stage:

$$u_i^{n+1} = \frac{(1-r_i)u_i^n + A_i - \Delta t\sigma\left(u_i^n\right)^4}{1+\Delta tK} \tag{2.1}$$

Second stage:

$$u_i^{n+1} = \frac{u_i^n + A_i^{\text{new}}}{1 + r_i + \Delta t K + \Delta t \sigma \left(u_i^n\right)^3} \tag{2.2}$$

where A_i^{new} is calculated as following:

$$r_i = \Delta t \sum_{j \neq i} \frac{1}{C_i R_{ij}} \quad \text{and} \quad A_i^{new} = \Delta t \sum_{j \neq i} \frac{u_j^n}{C_i R_{ij}}$$

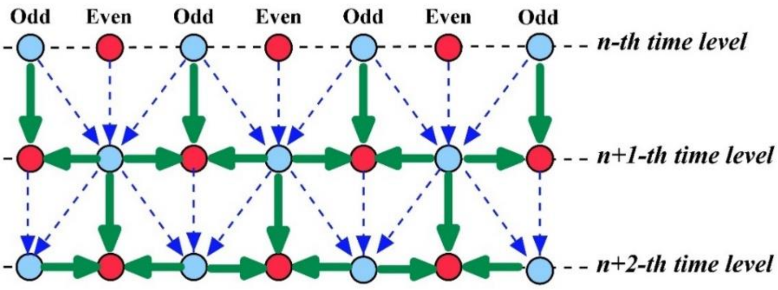

Figure 2.1. The stencil of the original odd-even hopscotch algorithm. Thin blue arrows and thick green arrows indicate operations at the first stage and second stage, respectively.

2.1.2 *The Leapfrog–Hopscotch method*

In the leapfrog-hopscotch (LH) method, we have a structure consisting of two half and several full time steps. The calculation starts again by taking a half-sized time step for the odd nodes using the initial values, then, for the even and odd nodes, full-time steps are taken strictly alternately until the end of the last timestep (orange box in Figure 2.2 B), which should be halved for odd nodes to reach the same final time point as the even nodes.

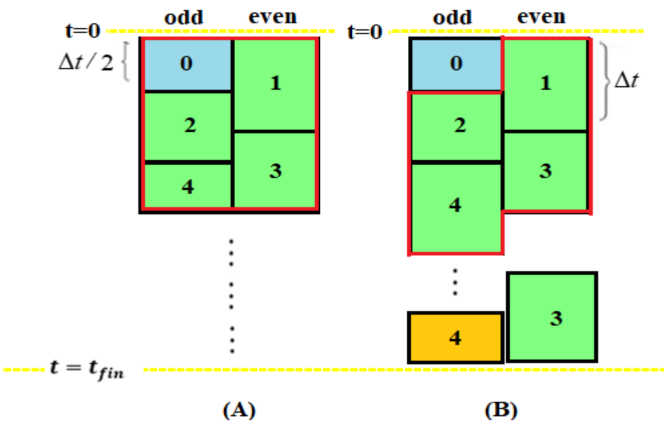

Figure 2.2. (A) The shifted-hopscotch structure. (B) The leapfrog-hopscotch structure.

2.3 *Stable, Explicit, Leapfrog-Hopscotch Algorithms for the Heat Equation*

I construct novel numerical algorithms to solve the heat or diffusion equation. I combined the hopscotch space structure with leapfrog time integration. By applying the theta method with nine different values of θ and the recently invented CNe method, I constructed 10^5 different leapfrog-hopscotch algorithm combinations and via subsequent numerical investigations, this huge number was decreased by excluding the combinations that underperformed and, finally, only the top five of these were retained. I used two two-dimensional stiff systems containing 10,000 cells with completely discontinuous random parameters and initial conditions, I demonstrate the performance of these top five methods in the case of large systems with random parameters and discontinuous initial conditions, by comparing them with other methods. My current work was inspired by the well-known leapfrog method [19] used by the molecular dynamics community to solve Newton's equations of motion.

The computer program calculated the aggregated relative error (ARE) quantities and then sorted the algorithms in descending order according to this quantity to obtain a ranking list of the 10^5 algorithm combinations. Finally, I manually checked the top of these lists for all of the four small systems to select the best 20 combinations, which have the following short form:

$$\begin{gathered}(C, C, C, C, C),\ (\tfrac14, \tfrac12, C, \tfrac12, \tfrac12),\ (\tfrac14, \tfrac12, C, \tfrac12, \tfrac12),\ (\tfrac14, \tfrac12, \tfrac12, \tfrac12, C),\\ (\tfrac14, \tfrac12, C, \tfrac12, C),\ (C, \tfrac12, C, \tfrac12, C),\ (C, \tfrac14, \tfrac12, C, \tfrac12),\ (C, \tfrac12, C, \tfrac12, \tfrac12),\\ (C, \tfrac12, C, C, \tfrac12),\ (C, \tfrac13, \tfrac12, \tfrac23, C),\ (\tfrac14, C, \tfrac14, C, \tfrac34),\ (0, C, \tfrac12, \tfrac12, \tfrac12, \tfrac12),\\ (C, \tfrac12, \tfrac12, \tfrac12, \tfrac12),\ (\tfrac14, C, \tfrac14, \tfrac12, \tfrac34),\ (0, \tfrac12, \tfrac12, \tfrac12, \tfrac12),\ (1, \tfrac12, \tfrac12, \tfrac12, \tfrac12),\\ (\tfrac14, \tfrac12, \tfrac12, \tfrac12, \tfrac12),\ (\tfrac13, \tfrac12, \tfrac12, \tfrac12, \tfrac12),\ (\tfrac12, \tfrac12, \tfrac12, \tfrac12, \tfrac12),\ (\tfrac15, \tfrac12, \tfrac12, \tfrac12, \tfrac12).\end{gathered} \tag{2.3}$$

Comparison with Other Methods for a Large, Very Stiff System

I plotted the L_∞, L_1, and energy errors as a function of the effective time step size h_{EFF}, and based on this (and on similar data that are presented in the next subsection), I selected the following top five combinations from those listed in Eq. (2.3) and discarded the reminder:

$$\begin{gathered}L1\ (C, C, C, C, C),\ L2\ (0, \tfrac12, \tfrac12, \tfrac12, \tfrac12),\\ L3\ (\tfrac15, \tfrac12, \tfrac12, \tfrac12, \tfrac12),\ L4\ (\tfrac14, \tfrac12, C, \tfrac12, \tfrac12),\\ L5\ (\tfrac15, \tfrac12, C, \tfrac12, \tfrac12).\end{gathered}$$

In Figure 2.3 and Figure 2.4 the energy and the L_∞ errors is presented as a function of the time step size and the total running time, respectively. It is not surprising that the implicit methods gained a slight advantage compared to the less stiff case, but the new L2 method outperforms all other examined method if not only the accuracy, but the speed is taken into account.

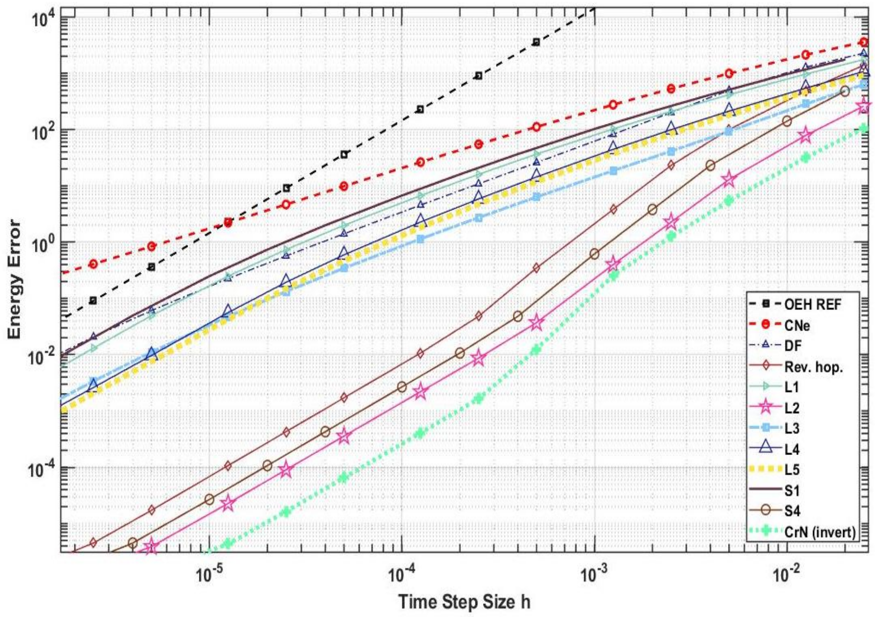

Figure 2.3. Energy errors as a function of the time step size for the second (very stiff) system, in the case of the original OEH method (OEH REF), the new algorithms L1-L5 and some other methods.

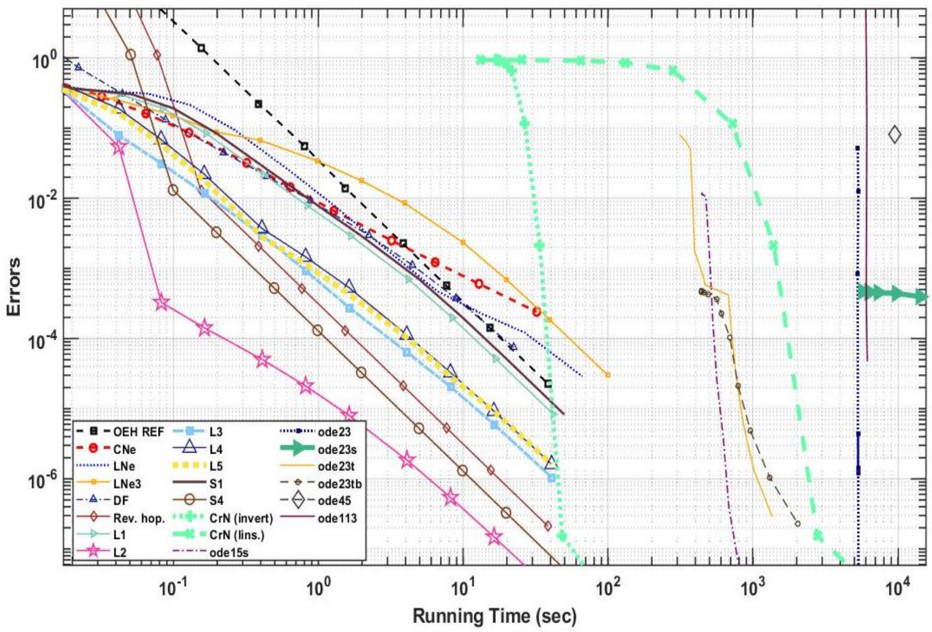

Figure 2.4. L_∞ errors as a function of the running time for the second (very stiff) system, in the case of the original OEH method (OEH REF), one stage CNe method, the new algorithms L1-L5 and a couple of other methods.

I found that, in general, the L2 (0, 1/2, 1/2, 1/2, 1/2) combination is the most competitive. This combination yields accurate results orders of magnitude faster than the well-optimized MATLAB routines, and it is more accurate than all of the other examined explicit and stable methods. Although, unlike the L1 (C, C, C, C, C) algorithm, L2 is not positivity preserving, it is also surprisingly robust for relatively large time step sizes, which is not that case for the original OEH algorithm. Moreover, this new L2 algorithm is easy to implement and requires an even smaller amount of function evaluation and computer memory than the conventional explicit second order

Runge–Kutta methods, such as the Heun method. I can conclude that it combines has the most important advantages of the standard explicit and the implicit methods. Now I present the final formula of the leapfrog-hopscotch method for conduction with the best already proven combination of formulas (L2) as follows:

The first stage has the length of a halved time step, thus we have the following special and general formulas

$$u_i^{\frac{1}{2}} = \frac{u_i^0 + \frac{r}{2}\left(u_{i-1}^0 + u_{i+1}^0\right) + \frac{h}{2}q_i}{1+r}, \quad and \quad u_i^{\frac{1}{2}} = \frac{u_i^0 + \frac{A_i^0}{2} + \frac{h}{2}q_i}{1+\frac{r_i}{2}} \tag{2.4}$$

Then a full-time step is made for the even nodes using

$$u_i^1 = \frac{(1-r)u_i^0 + r\left(u_{i-1}^{1/2} + u_{i+1}^{1/2}\right) + hq_i}{1+r}, \quad and \quad u_i^1 = \frac{\left(1-\frac{r_i}{2}\right)u_i^0 + A_i^{1/2} + hq_i}{1+\frac{r_i}{2}} \tag{2.5}$$

After this, full time steps are taken alternately for the odd and even nodes. Finally, a half-length time step must close the calculations for the odd nodes

$$u_i^T = \frac{(1-\frac{r}{2})u_i^{T-1/2} + \frac{r}{2}\left(u_{i-1}^T + u_{i+1}^T\right) + \frac{h}{2}q_i}{1+\frac{r}{2}}, \quad and \quad u_i^T = \frac{(1-\frac{r_i}{4})u_i^{T-1/2} + \frac{A_i^T}{2} + \frac{h}{2}q_i}{1+\frac{r_i}{4}} \tag{2.6}$$

2.4 A Comparative Study of Explicit and Stable Time Integration Schemes for Heat Conduction in an Insulated Wall

I extensively examine 13 numerical methods to solve the linear heat conduction equation in building walls. Eight of the used methods (including the previously examined leapfrog-hopscotch) are recently invented explicit algorithms which are unconditionally stable. First, I performed verification tests in a 2D case by comparing them to analytical solutions using equidistant and non-equidistant grids. Then I tested them on real-life applications in the case of one-layer (brick) and two-layer (brick and insulator) walls to determine how the errors depend on the real properties of the materials, the mesh type, and the time step size. I applied space-dependent boundary conditions on the brick side and time-dependent boundary conditions on the insulation side. The results show that the best algorithm is usually the original odd-even hopscotch method for uniform cases and the leapfrog-hopscotch algorithm for non-uniform cases. So, I perform systematic tests in the building walls by varying some parameters of the system and the mesh to examine how the performance of the individual methods changes and which of them is the best choice under different circumstances. I note that no comparative study has been conducted until my work even about the four already known explicit and stable methods examined in this work, namely the UPFD, odd-even hopscotch, Dufort-Frankel, and rational Runge-Kutta methods.

2.4.1 Brick wall with Insulation

The errors are plotted for non-equidistant mesh in Figure 2.5. One can now visualize that the LH method is accurate for this large time step size.

I observed that if we apply the insulator or go from equidistant mesh to increasingly non-equidistant meshes (both increase the stiffness), the LH method will be the most accurate among the unconditionally stable methods.

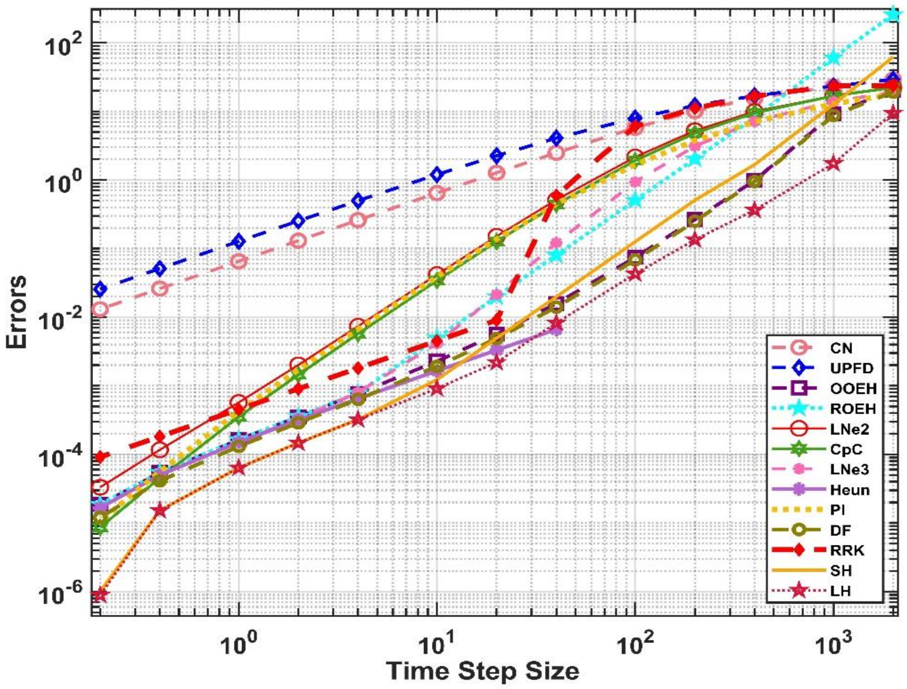

Figure 2.5. The maximum errors as a function of the time step size Δt in the case of non-equidistant mesh for a wall with insulation.

2.5 Comparison of the Performance of Traditional and New Numerical Methods for Long-Term Heat Transfer Simulations in Walls with Thermal Bridges

I extensively measure the running times of the most successful methods and compare them to the performance of other available solvers, for example, ANSYS transient thermal analysis and the built-in routines of MATLAB, where three different mesh resolutions are used. I show that the running time of our methods changes linearly with mesh size, unlike in the case of other methods. After that, I make a long-term simulation (one full winter month) of two-dimensional space systems to test the two best versions of the methods. The real-life engineering problem I solve is the examination of thermal bridges with different shapes in buildings to increase energy efficiency.

I work on transient heat transfer calculations using fundamental physical laws (ab initio approach). Therefore, it is expected that these results are much more accurate than those based on the usual (ISO) standards, which are steady-state calculations without solving the transient PDE and therefore cannot properly consider. My long-term goal is to revolutionize these simulations (at this stage by the numerical methodology) to make transient simulations more available due to reduced computational cost and programming difficulty. In this work, I continue the above-mentioned investigations. I use ANSYS' thermal analysis solutions that help engineers solve the most complex thermal challenges and predict how their designs will perform with temperature changes. However, because simulation by this kind of software takes a long time and requires serious computer resources, I compare my methods with ANSYS to investigate runtime, stability, and other features [20]. Now the goal is to systematically evaluate how the performance of the various solvers (including MATLAB routines and ANSYS) depends on the mesh settings to see which one is optimal for certain accuracy requirements.

2.5.1 Comparing ANSYS Solvers and MATLAB Methods for the Fine Mesh System

The errors for the fine mesh are represented as a function of the running time in Figure 2.6. It is observed that the mesh smoothing has a positive effect on the ANSYS solvers' accuracy; the errors are decreasing with the time step size. Despite this, some of the suggested explicit methods coded in MATLAB are still the best in both speed and accuracy. If there is a large system, the ANSYS solvers are very slow. However, it is believed that with increasing the system size, the MATLAB built-in routines would be slower at a larger rate, so it is also believed they would be the slowest for even finer mesh. The two best methods are the Dufort-Frankel schemes with the pseudo-implicit treatment of both the convection and the radiation term (DF-D) and the leapfrog-hopscotch with the pseudo-implicit treatment of the radiation term (LH-PseudoImp). In fact, their benefits grow as the size of the system increases. For this reason, these two methods are chosen to create a long-term simulation in the following part.

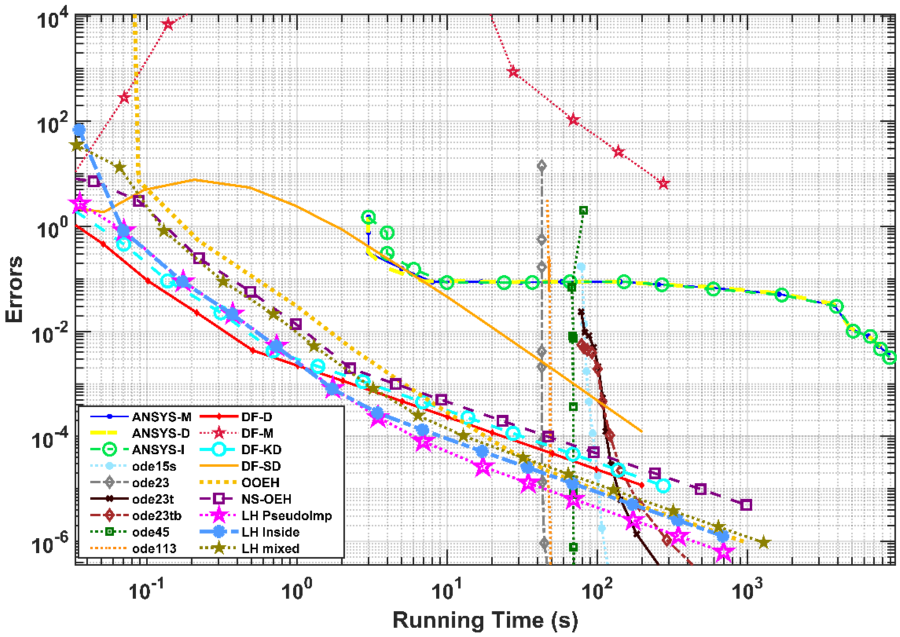

Figure 2.6. The maximum errors for the tested methods in the fine mesh case as a function of the running time.

2.5.2 Long-Term Simulations

Figure 2.7 illustrates a comparison between the energy losses. The largest thermal loss is shown in the case of a one-layer case, and these losses decrease with the presence of insulation. In the presence of a thermal bridge, the losses are larger than those without, and the losses are slightly larger when the thermal bridge is straight compared to a bent bar. The costs of energy consumption and heat loss are displayed in Table 2.1.

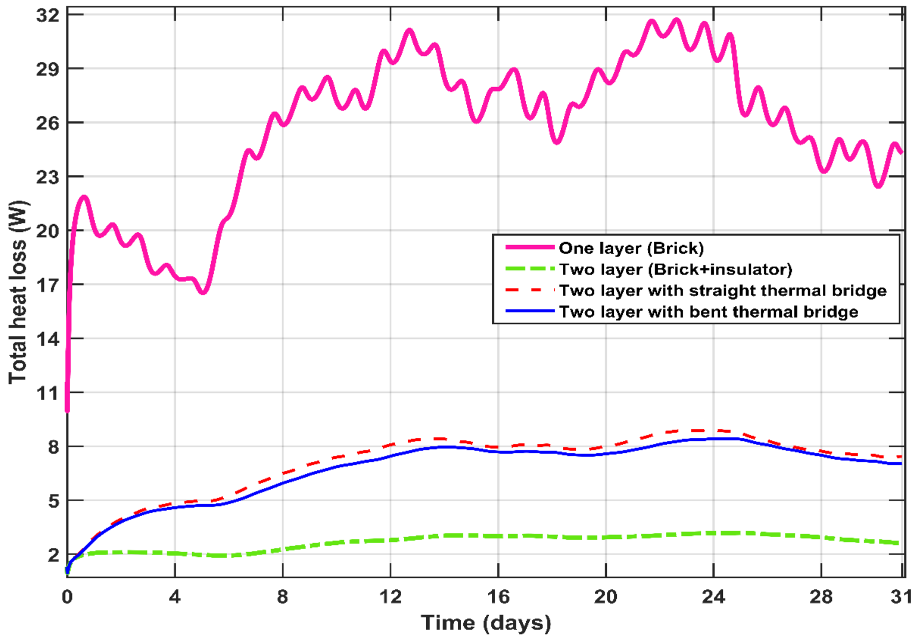

Figure 2.7. The distribution of total heat loss in Watt units as a function of time in days for the long-term simulation of all wall cases.

Table 2.1. Heat loss through the 1 m^2 part of the wall and the energy cost in USD and HUF.

	One Layer	Two Layers	Two Layers with a Straight Bridge	Two Layers with Bent Bridge
Heat loss (full month, kWh)	19.14	1.99	5.29	5.01
The cost in USD (full month)	1.9	0.2	0.53	0.5
The cost in HUF (full month)	717.19	74.63	198.24	187.8

The analysis suggest that my methods are better than all ANSYS solvers and MATLAB routines, whereas ANSYS was less accurate and slower, and it was observed that the best performance was achieved by the leapfrog–hopscotch and the Dufort–Frankel algorithms with the pseudo-implicit treatment of the nonlinear radiation term. Therefore, these two methods were applied to real problems, and a long-term simulation of four cases was performed. The temperature distribution and total heat losses of all cases were calculated. I found the straight thermal bridge to be energetically worse than others, and the total heat loss during the month (one-layer, two-layer, two-layer with a straight thermal bridge, and two-layer with a bent thermal bridge) was, respectively, 19.14, 1.99, 5.29, and 5.01 kWh for a 1 m^2 wall surface. I can conclude that the numerical simulation methodology is established in this work.

2.6 Applying Recent Efficient Numerical Methods for Long-Term Simulations of Heat Transfer in Walls to Optimize Thermal Insulation

Since transient simulations need a lot of resources, the heat loss through the walls of buildings in winter is often estimated by a simple steady-state calculation based on methods like the Degree-days, which is frequently rather inaccurate. So, I carried out transient simulations using the new leapfrog-hopscotch and the modified Dufort-Frankel algorithms, which are the most efficient, stable, and explicit numerical methods to deal with heat transfer problems, according to previous investigations. The optimum thickness of insulation, energy savings, and payback time are determined using an economic model that considers the orientation of the external walls, solar radiation, the cost of insulation materials, the present cost of energy consumption, and the cost over the 25-year lifetime of a building in Miskolc City, and a case is analysed in the cold season. Three materials and a range of thicknesses are investigated: Expanded Polystyrene (EPS), glass-wool, and rock-wool. I found the transient way to calculate heat loss to be quick and accurate. Additionally, it was looked at how well the walls conducted heat under optimal conditions. Comparing this study to others of a similar type, one of its unique characteristics was the use of less expensive local materials, to optimize investment on insulation. I use an ab-initio approach to compute transient heat transfer using fundamental physical rules. It is anticipated that these findings will be substantially more accurate than those based on the common (ISO) standards [21], [22], which use steady-state calculations without solving the transient PDE. To make transient simulations more accessible owing to decreased processing cost and programming difficulties, my long-term aim is to revolutionize these simulations.

Figure 2.8 displays the optimum insulation thicknesses for the three walls according to total life cycle saving and payback time.

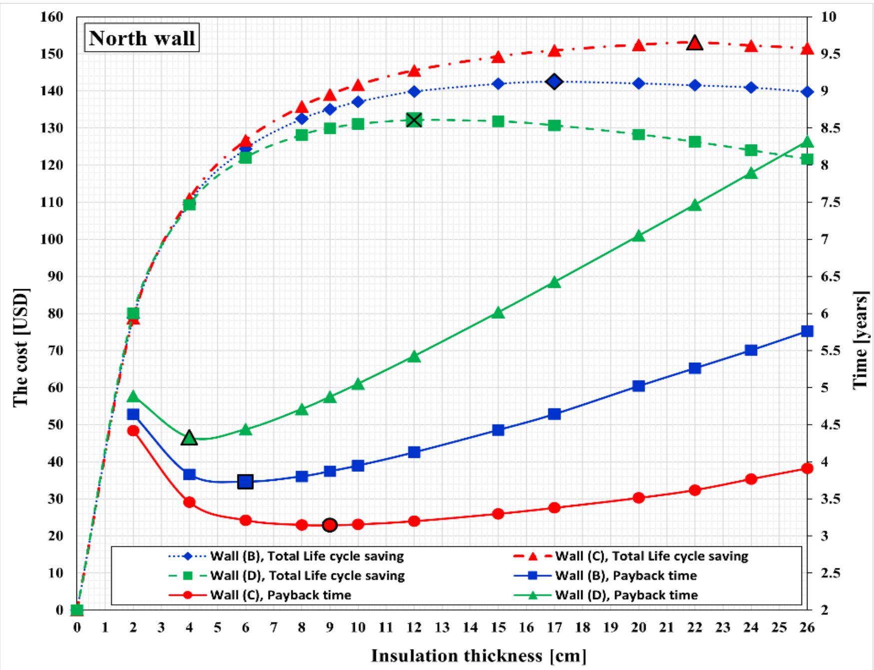

Figure 2.8. The optimum insulation thickness for the three walls according to total life cycle saving and payback time, where the right axis refers to time in years, and the left one refers to the cost in USD.

I chose the best insulation, which is in Wall (C), in terms of the best total life cycle saving and payback time, and I compared the four directions of this wall , and the optimal thicknesses is shown in the Figure 2.9.

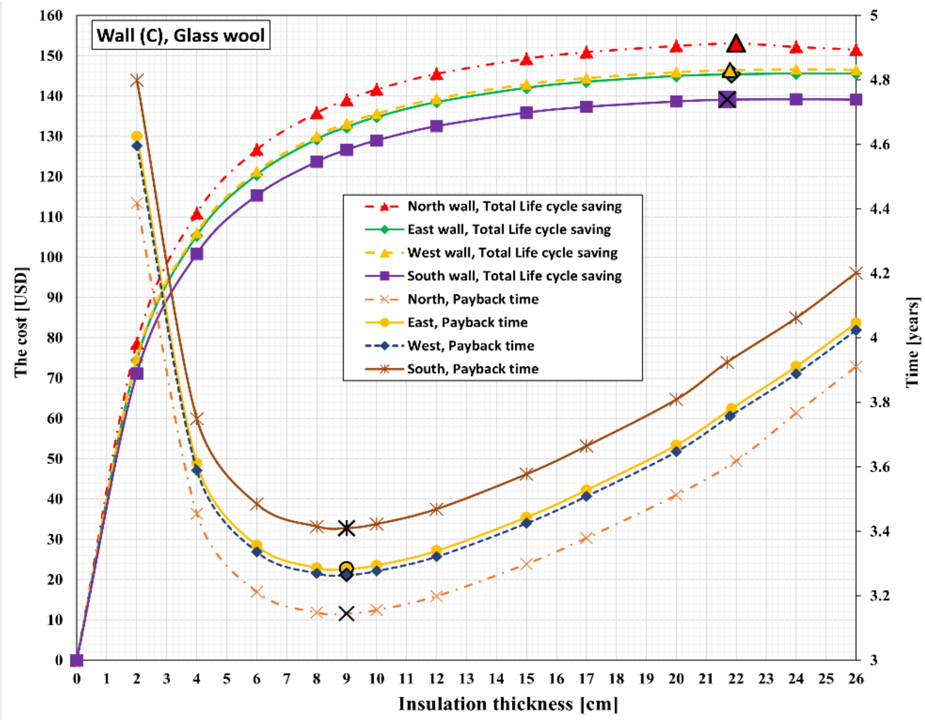

Figure 2.9. The optimum insulation thickness for the four orientations of the wall (C) according to total life cycle saving and payback time, where the right axis refers to time in years , and the left one refers to the cost in USD.

I found the total heat loss (without insulation, with a thickness of 10 cm EPS, glass-wool, and Rock-wool) was, respectively, 2792.628, 470.145, 480.45, and 449.7 kWh for a full flat. The optimum insulation thicknesses for the north-facing wall are 17, 22, and 12 cm, and the life cycle energy savings are 142.5, 153, and 132.12 kWh/m^2 for the EPS, glass-wool, and rock-wool, respectively, and the payback times are 3.73, 3.14, and 4.33 years. The optimal insulating properties can be achieved with 22 cm of thick glass wool, which only slightly depends on the wall orientation, according to the life cycle analysis. During optimization, it was assumed that the insulation cost increases linearly with insulation thickness, whereas the energy cost is found to decrease as insulation thickness increases, while the principle of diminishing returns is fulfilled. The optimum thickness varies based on the substance and how it interacts with the outdoor environment. The current conclusions and results, and particularly the relative performance of glass wool, depend on the specific values of the parameters employed in the economic and thermal analysis.

3. NEW SCIENTIFIC RESULTS – THESES

T1. I tested 10^5 different leapfrog-hopscotch algorithm combinations to solve the heat or diffusion equation by combining the hopscotch space structure with leapfrog time integration. I applied the well-known theta method with nine different values of θ and the recently invented CNe method. Then I generated preliminary numerical results, and using these as a basis, I selected the five most effective methods for more research. The effectiveness of the chosen methods was evaluated for two 2-dimensional systems containing 10000 cells in the case of large systems with completely discontinuous random parameters and initial conditions. I showed the competitiveness of the suggested methods by demonstrating that they can provide results with acceptable accuracy orders of magnitude quicker than the well-optimized MATLAB routines (4).

T2. I extensively investigated 13 numerical methods to solve the linear heat conduction equation in building walls. Eight of the used methods are recently invented explicit algorithms including those mentioned in T1 which are unconditionally stable. First, I performed verification tests in a 2D case by comparing them to analytical solutions, using non-equidistant grids. Then I tested them on real-life applications in the case of one-layer (brick) and two-layer (brick and insulator) walls to determine how the errors depend on the real properties of the materials, the mesh type, and the time step size. I applied zero Dirichlet boundary on the walls. The results show that the best algorithm is usually the original odd-even hopscotch method for uniform cases and the leapfrog-hopscotch and the Dufort-Frankel algorithms for non-uniform cases (3).

T3. I extensively measured the running times of the most successful methods and compared them to the performance of other available solvers, for example, ANSYS transient thermal analysis and the built-in routines of MATLAB. I systematically evaluated how the performance of the various solvers (including MATLAB routines and ANSYS) depends on the mesh settings to see which one is optimal for certain accuracy requirements. I used three mesh sizes: 40 × 40, 80 × 80, and 120 × 120. I showed that the running time of my methods changes linearly with mesh size, unlike in the case of other methods. Three simple analytical solutions of the heat equation were used with an equidistant mesh for verification in the case of homogeneous material properties (one brick layer). All the methods used and the ANSYS solvers are confirmed to be convergent. These experiments suggested that my methods are better than all ANSYS solvers and MATLAB routines, whereas ANSYS was less accurate and slower, and I observed that the best performance was achieved by the leapfrog–hopscotch and the Dufort-Frankel algorithms with the pseudo-implicit treatment of the nonlinear radiation term (2).

T4. I developed and tested a simulation methodology based on fundamental physical principles and laws (ab initio approach) to study transient heat transfer in a two-dimensional wall without an insulator, with an insulator, and different types of thermal bridges. I made a long-term simulation (one full winter) of two-dimensional space systems to test the two best versions of the methods mentioned in T3. The real-life engineering problem I solved is the examination of thermal bridges with different shapes in buildings to increase energy efficiency. The temperature distribution and total heat losses of all cases were calculated. I found the straight thermal bridge to be energetically worse than others. I conclude that the numerical simulation methodology is established in my work (2), (11).

T5. I made transient heat transfer simulations through multilayer walls with different materials and thicknesses that is subject to the typical external temperature and solar radiation specific to Hungary's environment using the leapfrog-hopscotch and modified Dufort-Frankel algorithms. I calculated the winter heating loss (across the walls) and temperature distribution of all cases according to wall orientations. I found the transient way to calculate the heat loss to be quick and accurate more than steady-state calculations based on methods like the Degree-days, which is frequently rather inaccurate (1).

T6. Using the leapfrog-hopscotch and modified Dufort-Frankel algorithms to make transient heat transfer simulations through multilayer walls, I determined the optimum thickness of insulation, energy savings, and payback time using an economic model that considers the orientation of the external walls, solar radiation, the cost of insulation materials, the present cost of energy consumption, and the cost over the 25-year lifetime of a building in Miskolc City, and a case is examined in the cold season. Additionally, it was looked at how well the walls conducted heat under optimal conditions. Comparing this study to others of a similar type, one of its unique characteristics was the use of less expensive local materials, to optimize investment on thermal insulation. I investigated three materials and a range of thicknesses: Expanded Polystyrene (EPS), glass-wool, and rock-wool. The results demonstrated a small, but noticeable impact of wall orientation on the thermal efficiency of the walls during the winter season of the environment under consideration. The small difference between northern and southern orientation can be explained by the fact that the first part of the winter is typically rather cloudy in Hungary. The optimal insulating properties can be achieved with 22 cm of thick glass wool, which only slightly depends on the wall orientation, according to the life cycle energy savings analysis (1).

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

(1) Omle, I.; Kovács, E.; Bolló, B. Applying Recent Efficient Numerical Methods for Long-Term Simulations of Heat Transfer in Walls to Optimize Thermal Insulation. *Elsevier, Results Eng.* 2023, 101476, doi: 10.1016/j.rineng.2023.101476.

(2) Omle, I.; Askar, A.H.; Kovács, E.; Bolló, B. Comparison of the Performance of New and Traditional Numerical Methods for Long-Term Simulations of Heat Transfer in Walls with Thermal Bridges. Energies 2023, 16, 4604. https://doi.org/10.3390/en16124604.

(3) 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.

(4) 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, p. 92, 2021.https://doi.org/10.3390/computation9080092

(5) Askar, A.H.; Omle, I.; Kovács, E.; Majár, J. Testing Some Different Implementations of Heat Convection and Radiation in the Leapfrog-Hopscotch Algorithm. Algorithms 2022.https://doi.org/10.3390/a15110400.

(6) Á. Nagy, M. Saleh, I. Omle, H. Kareem, and E. Kovács, “New Stable, Explicit, Shifted-Hopscotch Algorithms for the Heat Equation,” *Mathematical and Computational Applications*, vol. 26, no. 3, p. 61, Aug. 2021, doi: 10.3390/mca26030061.

(7) Khayrullaev, H.; Omle, I.; Kovács, E. "Systematic Investigation of the Explicit, Dynamically Consistent Methods for Fisher’s Equation". Computation 2024, 12, 49. https:// doi.org/10.3390/computation12030049.

(8) Omle, I.; Askar, A.H.; Kovács, E. “Systematic testing of explicit positivity preserving algorithms for the heat-equation”. https://doi.org/10.28919/jmcs/7407.

(9) Omle, I. “New explicit algorithm based on the asymmetric hopscotch structure to solve the heat conduction equation,” Multidiszcip. Tudományok, vol. 11, no. 5, pp. 233–244, 2021.

(10) Omle, I.; Kovács, E.; Bolló, B. "Multi-objective optimization and simulation for multiple models of walls to estimate heat gain using Artificial Neural Network model." Multidiszciplináris Tudományok 13.3 (2023): 159-174. https://doi.org/10.35925/j.multi.2023.3.17.

(11) Omle, I.; Kovács, E.; “Evaluate Recent Numerical Methods for Long-Term Simulation to Study the Effect of Different Shapes of Thermal Bridges in Walls”. Advances in Mechanical Engineering, 2024.

(12) Omle, I.; Kovács, E.; Bolló, B. "The Effect of Surface Triangular Roughness Inspired by Nature with Different Angles on the Interaction between Free Convection and Surface Radiation in a Square Cavity". Heat Transfer Research 2023, DOI: 10.1615/HeatTransRes.2023048854.

(13) Omle, I.; Askar, A.H.; Kovács, E.; "Impact of Wall Roughness Elements Type and Height on Heat Transfer Inside a Cavity". Pollack Periodica (2024). DOI:10.1556/606.2024.00986

(14) Omle, I.; Kovács, E.; Bolló, B. "Testing the effect of radiation emissivity on heat convection inside a square enclosure". 9th International Zeugma Conference on Scientific Research: The book of full texts, Istanbul, Turkey : (2023) pp. 430-437. , 8 p. ISBN: 9786256404762.

(15) Abed, H. H., Omle, I., Askar, A. H., & Kovács, E. (2023). Experimental Study of the Stability and Thermophysical Properties for Different Particle Size of Al_2o_3-H_2o Nanofluid. Journal of Engineering Science and Technology, 18(6), 2793-2808.

(16) Omle, I.; Askar, A.H.; Kovács, E. “Optimizing the Design of Container Houses Using Argon and Recycled Plastic Materials”. 2024 (**Under Review**, Q1)

5. REFERENCES

[1] S. Haq and I. Ali, "Approximate solution of two-dimensional Sobolev equation using a mixed Lucas and Fibonacci polynomials," *Eng Comput*, vol. 38, pp. 2059–2068, Aug. 2022, doi: 10.1007/s00366-021-01327-5.

[2] S. A. Lima, M. Kamrujjaman, and M. S. Islam, "Numerical solution of convection-diffusion-reaction equations by a finite element method with error correlation," *AIP Adv*, vol. 11, no. 8, p. 85225, Aug. 2021, doi: 10.1063/5.0050792/968138.

[3] M. Ivanovic, M. Svicevic, and S. Savovic, "Numerical solution of Stefan problem with variable space grid method based on mixed finite element/finite difference approach," *Int J Numer Methods Heat Fluid Flow*, vol. 27, no. 12, pp. 2682–2695, 2017, doi: 10.1108/hff-11-2016-0443.

[4] D. A. Anderson, J. C. Tannehill, R. H. Pletcher, R. Munipalli, and V. Shankar, "Computational Fluid Mechanics and Heat Transfer; Fourth Edition." [Online]. Available: https://www.routledge.com/

[5] E. Kovács, Á. Nagy, and M. Saleh, "A set of new stable, explicit, second order schemes for the non-stationary heat conduction equation," *Mathematics*, vol. 9, no. 18, p. 2284, 2021.

[6] I. Ali, S. Haq, K. S. Nisar, and S. U. Arifeen, "Numerical study of 1D and 2D advection-diffusion-reaction equations using Lucas and Fibonacci polynomials," *Arabian Journal of Mathematics*, vol. 10, no. 3, pp. 513–526, Dec. 2021, doi: 10.1007/s40065-021-00330-4.

[7] I. Z. Reguly and G. R. Mudalige, "Productivity, performance, and portability for computational fluid dynamics applications," *Comput Fluids*, vol. 199, p. 104425, Mar. 2020, doi: 10.1016/J.COMPFLUID.2020.104425.

[8] F. Gagliardi, M. Moreto, M. Olivieri, and M. Valero, "The international race towards Exascale in Europe," *CCF Transactions on High Performance Computing*, vol. 1, no. 1. Springer, pp. 3–13, May 06, 2019. doi: 10.1007/s42514-019-00002-y.

[9] F. Sanjaya and S. Mungkasi, "A simple but accurate explicit finite difference method for the advection-diffusion equation," in *Journal of Physics: Conference Series*, 2017, p. 12038.

[10] S. Savović, B. Drljača, and A. Djordjevich, "A comparative study of two different finite difference methods for solving advection–diffusion reaction equation for modeling exponential traveling wave in heat and mass transfer processes," *Ricerche di Matematica*, vol. 71, pp. 245–252, 2022, doi: 10.1007/s11587-021-00665-2.

[11] Á. Nagy, M. Saleh, I. Omle, H. Kareem, and E. Kovács, "New stable, explicit, shifted-hopscotch algorithms for the heat equation," *Mathematical and Computational Applications*, vol. 26, no. 3, p. 61, 2021.

[12] I. Omle, A. H. Askar, and E. Kovács, “Systematic testing of explicit positivity preserving algorithms for the heat-equation,” *J. Math. Comput. Sci.*, vol. 12, no. 0, p. Article ID 162, 2022, doi: 10.28919/10.28919/JMCS/7407.

[13] S. Mahmoud, N. Ádám, and K. Endre, “Construction and investigation of new numerical algorithms for the heat equation: Part III,” *Multidiszciplináris Tudományok*, vol. 10, no. 4, pp. 349–360, 2020.

[14] 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.

[15] A. H. Askar, I. Omle, E. Kovács, and J. Majár, “Testing Some Different Implementations of Heat Convection and Radiation in the Leapfrog-Hopscotch Algorithm,” *Algorithms*, vol. 15, no. 11, p. 400, 2022.

[16] A. H. Askar, Á. Nagy, I. F. Barna, and E. Kovács, “Analytical and Numerical Results for the Diffusion-Reaction Equation When the Reaction Coefficient Depends on Simultaneously the Space and Time Coordinates,” *Computation*, vol. 11, no. 7, p. 127, 2023.

[17] B. M. Chen-Charpentier and H. V Kojouharov, “An unconditionally positivity preserving scheme for advection--diffusion reaction equations,” *Math Comput Model*, vol. 57, no. 9–10, pp. 2177–2185, 2013.

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

[19] “Leapfrog integration.” Accessed: Jan. 12, 2024. [Online]. Available: https://en.wikipedia.org/wiki/Leapfrog_integration

[20] E. M. Alawadhi, *Finite element simulations using ANSYS*. CRC Press, 2009.

[21] Swedish Standards Institute, “Thermal Bridges in Building Construction—Heat Flows and Surface Temperatures—Detailed Calculations. EN ISO 10211: 2017.” Accessed: Apr. 20, 2023. [Online]. Available: https://www.iso.org/%0Astandard/65710.html

[22] Swedish Standards Institute, “Thermal Bridges in Building Construction—Linear Thermal Transmittance— Simplified Methods and Default Values. EN ISO 14683: 2017.” Accessed: Apr. 20, 2023. [Online]. Available: https://www.iso.org/standard/65706.html