Proceedings of Science and Mathematics
Faculty of Science, Universiti Teknologi Malaysia

Vol. 15, 2023, page 128-135

2D-Parabolic Equation Model for Fruit Waste Drying Process Using Instant Control Pressure Technique

Muhammad Noor Akmal Buhari, Norma Alias*
Department of Mathematics, Faculty of Science, UTM, Skudai, Johor Bahru, Malaysia
*Corresponding author: normaalias@utm.my

Abstract
This study intends to convert food waste into useful goods using the Instantaneous Control Pressure Drop (DIC) technique. A mathematical model that combines elliptic and parabolic equations is used to simulate the process, with temperature and pressure being the two main considerations. The model is discretized using the finite difference technique (FDM), and massive volumes of data are handled using CUDA. For numerical simulations using the Gauss-Seidel iteration method, known precise solutions of partial differential equations are compared. On both sequential and parallel methods, both models are examined. All food waste might be recycled in the future of DIC while nutrients are kept intact and temperatures are kept at the proper levels.

Keywords: Mathematical Model; Parabolic Equations; DIC; finite difference method; CUDA.

Introduction
Scientists and corporations place a premium on food safety. Traditional food transformation and preservation procedures frequently fail to remove infections or inhibit microbial growth. Consumer views and international standards are becoming more aligned, yet heat-sensitive components and compounds provide difficulties. Traditional processing techniques frequently result in low production efficiency, ingredient waste, and excessive energy use. More effective, appropriate production processes are required to maintain microbiological safety while also preserving organoleptic and nutritional quality [Hamoud-Agha and Allaf, 2019].

[Sarpong, 2018] discovered that as drying continued, the rate of dehydration slowed down after the first fast decline in moisture content. There was no steady phase in the drying kinetics, and the time of decreasing rate was when drying was most prevalent. Water diffusion was the key factor controlling how long banana samples retained moisture, although no drying rate was consistently maintained under various drying settings. The drying rate was mostly determined by moisture diffusion from the inside to the surface.

DIC is a thermo-mechanical method that uses instantaneous thermodynamics to treat heat-sensitive materials at high temperatures and pressures for a short period of time [Pech-Almeida et al., 2021]. It entails subjecting biological food matrices to saturated steam pressure treatments of 100-900 kPa for a brief period, followed by an abrupt pressure drop of more than 500 kPa per second. Water auto vaporization, rapid cooling of biological products, and cell growth in the matrix are all caused by this immediate regulated pressure decrease.

Equation 2D-Parabolic Models are elliptic and parabolic partial differential equations that frequently appear in time-dependent events [Arendt, 2019]. They exhibit fascinating behaviors like phase transitions, singularities, patterns, and structures. The parabolic equation, a partial differential equation with the ability to instantly transport heat, gives information on how heat energy disperses within a medium [Cassol, 2019]. It is a good indicator of how quickly heat may spread. The test functions method serves as the major method of outcome verification as the necessary preconditions for a weak resolution of parabolic-type equations and systems on the Heisenberg group are shown in this study [Wang, 2020].

Methodology and Model

A specific kind of partial differential equation called a parabolic equation is used to describe how heat or other variables diffuse over time [Cassol, 2019]. In the realm of chemical engineering, it is commonly used to represent reaction-diffusion processes and offers insights into the instantaneous transmission of heat. According to [Saipan @ Saipol (2017)], the non-dimensionalized version of the 2D parabolic equation model is adopted.

$$\frac{\delta^2 u}{\delta x^2} + \frac{\delta^2 u}{\delta y^2}, 0 \leq x \leq L_x, 0 \leq y \leq L_y, t \geq 0$$

With initial conditions:

$$t = 0; u(x, y, 0) = u_0$$

$$x = 0; \frac{\delta u}{\delta x} = 0$$

$$y = 0; \frac{\delta u}{\delta y} = 0$$

And boundary conditions:

$$u(0, y, t) = u(1, y, t) = u(x, 0, t) = u(x, 1, t) = u_0$$

By using finite difference method to discretize the equation, the model becomes

$$\frac{\delta u_{i,j}}{\delta t} = \frac{\delta^2 u_{i,j}}{\delta x^2} + \frac{\delta^2 u_{i,j}}{\delta y^2}$$

Is approximate by

$$\frac{u_{i,j+1} - u_{i,j}}{\Delta t} = \frac{u_{i+1,j} - 2u_{i,j} + u_{i-1,j}}{\Delta x^2} + \frac{u_{i+1,j} - 2u_{i,j} + u_{i-1,j}}{\Delta y^2}$$

At every point, the above equation will generate $u_{i-1,j}$,$u_{i,j}$ and $u_{i+1,j}$ of previous time step and form this point can be calculated to get value of $u_{i,j+1}$.

Utilizing MATLAB and CUDA to demonstrate the differences between sequential and parallel computing to compare sequential and parallel computing, MATLAB will be used.

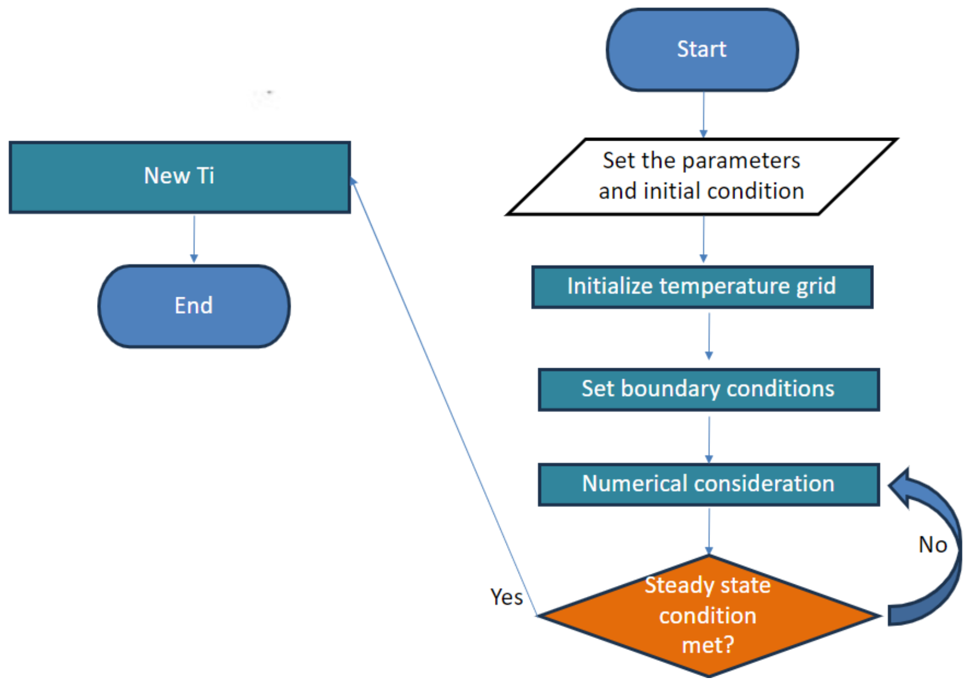

Figure 1 **The flowchart of sequential algorithm**

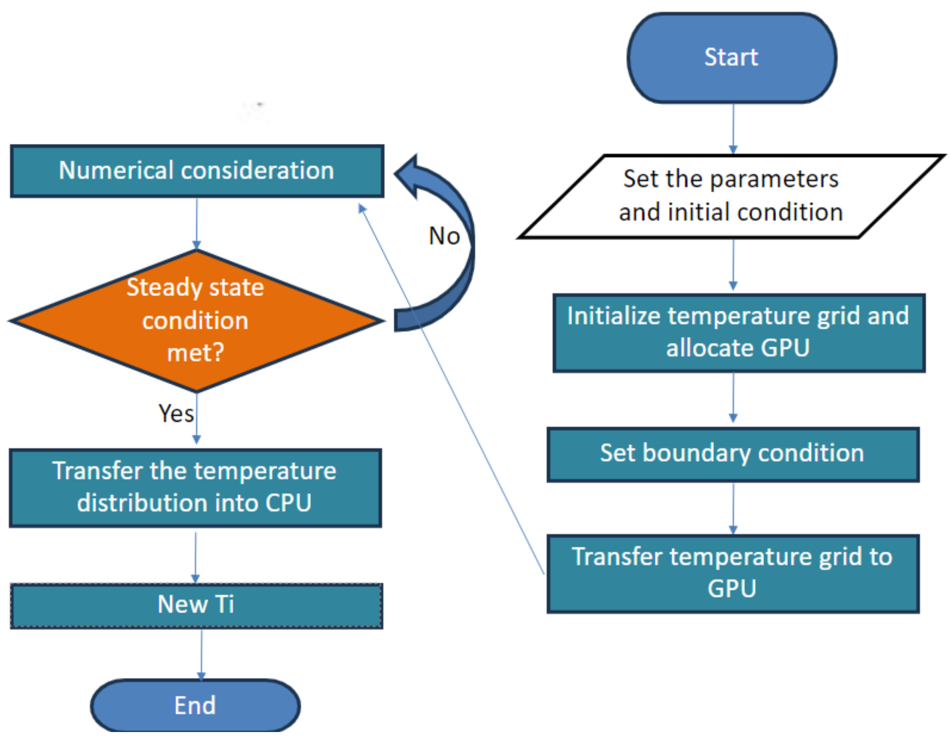

Figure 2 **The flowchart of parallel algorithm**

Result and discussion

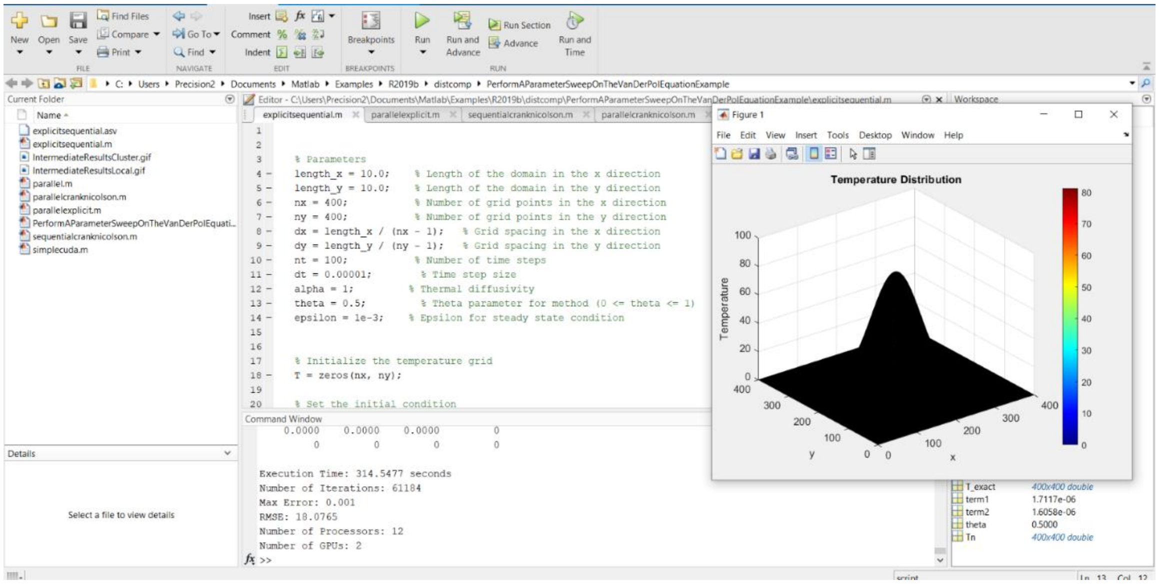

Figure 3 The code and graph of sequential computing

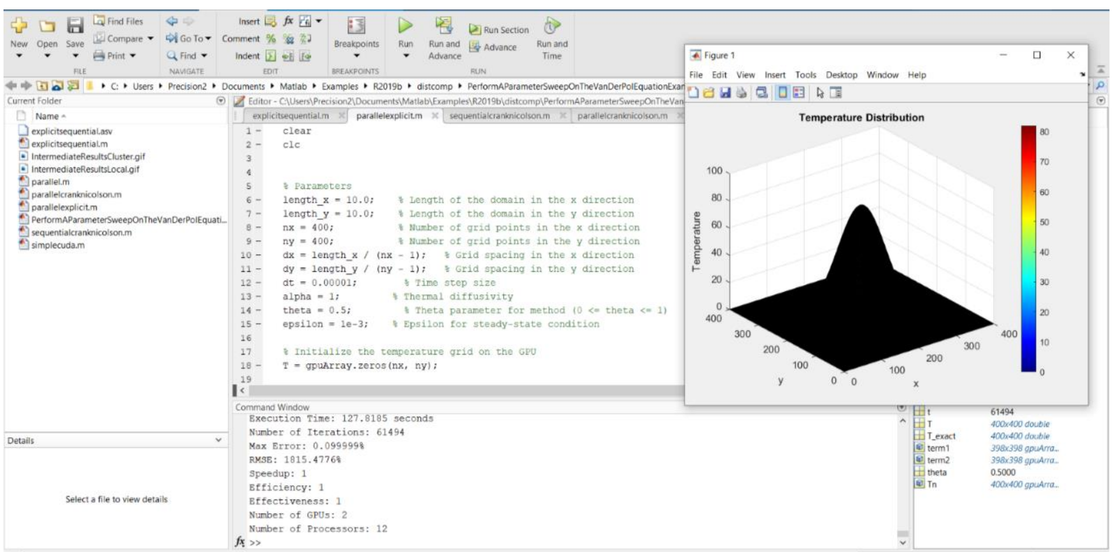

Figure 4 The code and graph for parallel computing

Parameters	Notation	Value
Length of spatial domain in x direction	L_x	10
Length of spatial domain in y direction	L_y	10
Number of grid points in x direction	N_x	400
Number of grid points in y direction	N_y	400
Duration of temporal domain	T	100
Number of time steps	N_t	100
Initial condition	u_0	200

Table 1 Parameters, initial and boundary conditions of 2D-Parabolic Equation Model

The Graphics Processing Unit (GPU) has become a powerful and efficient programmable unit for graphic rendering and general-purpose computations [Garcia-Feal et al., 2020]. It is often referred to as a parallel co-processor to the Central Processing Unit (CPU) due to its stream processing architecture, which is well-suited for executing computationally intensive parallel tasks. The GPU has gained popularity for high-performance scientific computing due to its low cost and high arithmetic computation power. It has spurred the development of general-purpose applications, such as OpenGL, DirectX, and Compute Unified Device Architecture (CUDA), which enable efficient parallel execution without the need for extensive graphics programming knowledge [Dahawi et al., 2021].

MATLAB, a high-level programming language, offers essential capabilities for both engineering and computer fields, including digital image processing and signal processing tasks [Ijemaru,2021]. Its built-in algorithms, ease of implementation, debugging features, data analysis capabilities, and support for simulations make it a valuable resource for handling digital images and related tasks.

Sequential		
Parameters	Crank-Nicolson Method	Explicit Method
$L_x = 10$ $L_y = 10$ $nx = 400$ $ny = 400$ $dx = L_x/(nx-1)$ $dy = L_y/(ny-1)$ $nt = 100$ $dt = 0.00001$ $\alpha = 1$ $\theta = 0$ $\epsilon = 1e-3$	Execute Time = 308.07s No of Iterations = 61184 Max error =0.1% RMSE=18.0765	Execute Time =313.29s No of Iterations=60869 Max error=0.01% RMSE=17.9972

Table 2 Sequential result using MATLAB

Parallel		
Parameters	Crank-Nicolson Method	Explicit Method
$L_x = 10$ $L_y = 10$ $nx = 400$ $ny = 400$ $dx = L_x/(nx-1)$ $dy = L_y/(ny-1)$ $nt = 100$ $dt = 0.00001$ $\alpha = 1$ $\theta = 0$ $\epsilon = 1e-3$	Execute Time = 114.85s No of Iterations = 61494 Max error =0.1% RMSE=18.1548	Execute Time = 116.56s No of Iterations = 65099 Max error =0.01% RMSE=18.0438

Table 3 Parallel result using CUDA MATLAB

Sequential programming is the sequential execution of instructions inside a single thread or process, making it perfect for basic calculations that do not need parallel processing [Yang, 2023]. It considers elements such as hardware, operating system, software, propagation delay, and access time. The decline in efficiency in sequential systems is not a technical mistake, but rather a natural result of the classical computer paradigm [V'egh, 2020]. Supercomputers handle workloads that lead to energy waste while also limiting job scale and complexity.

To increase processing performance and solve challenging computer problems, parallel computing is a technology that processes many instructions or activities concurrently [Ke,2021]. Parallel in time and parallel in space are the two categories that it falls under. The efficiency of a programmer is increased by using parallel processing techniques, which distribute calculations over several processors or cores. To improve computing efficiency and speed up complicated job execution, parallelization divides complex processes into smaller sub activities. On CPU and GPU devices, parallelization is employed to enhance CSEM modelling performance. By combining algorithm development with algorithm execution, parallel computing may dramatically improve the efficiency of a scheme while fostering scheme optimization. Parallel computing also makes the most of modern technology's multi-core capabilities.

According to Table 2, the Crank-Nicolson approach is a numerical method for solving parabolic partial differential equations [Johnson, 2020]. It employs both explicit and implicit strategies to improve accuracy and stability while decreasing processor strain. The technique is stable and converges rapidly, outperforming explicit approaches in terms of speed, stability, and accuracy. For large time step sizes, it converges to exact solutions and is unconditionally stable. However, it may need more processing work than explicit techniques at small time step sizes.

According to Table 3, in terms of the amount of time needed to run a programmer, parallel computing performs better than sequential computing [Rao,2020]. Large performance gains from low-powered hardware combinations that cannot be used independently can be achieved. Processing power may be greatly increased by integrating these devices into a cluster. Using parallel computing increases the effectiveness and efficiency of optimization processes. Compared to sequential systems, it is more suited for numerical simulations since it leverages CPU and imputation techniques to handle simulation mistakes. The GPU-accelerated optimal search procedure for optimization issues is expedited by parallel computing, although this results in a higher total number of simulations than a linear speedup.

Conclusion

The study attempted to apply 2D-parabolic equations (DIC) for dehydration, with mathematical modelling regulated by instantaneous pressure drop and finite difference approaches. To calculate the data, the finite difference method was utilized, and the technology was incorporated into CUDA to test its capacity to describe and visualize the dehydration process. The ultimate purpose was to compare numerical programming results with the present DIC machine technique, assessing the machine's performance regarding dehydration mathematical models. The results shown that the mathematical model is compatible with the mathematical modelling of dehydration.

Future work on the dehydration process using DIC-based 2D-Parabolic Equations will focus on improving the mathematical model, performing sensitivity analysis to determine how various factors affect it, using optimization and control strategies, and investigating DIC-based model-based optimization options. Accuracy, realism, robustness, stability, and optimization efforts may all be enhanced by focusing on these areas. Optimizing process variables, equipment configurations, and operating circumstances can also boost output, reduce energy consumption, and enhance product quality.

References

Hamoud-Agha and Allaf, 2019. Hamoud-Agha, M. M. and Allaf, K. (2019). Instant controlled pressure drop (dic) technology in food preservation: Fundamental and industrial applications. Food preservation and waste exploitation, 1(3):17.

Sarpong, 2018. Sarpong, F. (2018). The kinetics and thermodynamics study of bioactive compounds and antioxidant degradation of dried banana (musa ssp.) slices using controlled humidity convective air drying. Journal of Food Measurement and Characterization.

Pech-Almeida et al., 2021. Pech-Almeida, J. L., T´ellez-P´erez, C., Alonzo-Mac´ıas, M., Teresa-Mart´ınez, G. D., Allaf, K., Allaf, T., and Cardador-Mart´ınez, A. (2021). An overview on food applications of the instant controlled pressuredrop technology, an innovative high pressure-short time process. Molecules, 26(21):6519.

Arendt, 2019. Arendt,W. (2019). Strict positivity for the principal eigenfunction of elliptic operators with various boundary conditions. Advanced Nonlinear Studies.

Cassol, 2019. Cassol, G. O. (2019). Hyperbolicity of the heat equation. IFACPapersOnLine

Wang, 2020. Wang, W. (2020). Stability and error estimates for the variable stepsize bdf2 method for linear and semilinear parabolic equations. Advances in Computational Mathematics.

Saipan @ Saipol, 2017. Saipan @ Saipol, H. F. (2017). Parallelization of multidimensional hyperbolic partial differential equation on d´etente instantan´ee contrˆol´ee dehydration process.

Garcia-Feal et al., 2020. Garcia-Feal, O., Cea, L., Gonzalez-Cao, J., Dom´ınguez J. M., and Gomez-Gesteira, M. (2020). Iberwq: A gpu accelerated tool for 2d water quality modeling in rivers and estuaries. Water, 12(2):413.

Dahawi et al., 2021. Dahawi, A. A., Alias, N. B., and Idris, A. (2021). Gpu parallelization for accelerating 3d primitive equations of ocean modeling. In Saeed, F., Al-Hadhrami, T., Mohammed, F., and Mohammed, E., editors, Advances on Smart and Soft Computing, pages 643–654, Singapore. Springer Singapore.

Ijemaru, 2021. Ijemaru, G. K. (2021). Image processing system using matlabbased analytics. Bulletin of Electrical Engineering and Informatics.

Yang, 2023. Yang, P. (2023). libemm: A fictious wave domain 3d csem modelling library bridging sequential and parallel gpu implementation. Computer Physics Communications, 288:108745.

V'egh, 2020. V'egh, J. (2020). Finally, how many efficiencies the supercomputers have? and, what do they measure? ArXiv, abs/2001.01266.

Ke, 2021. Ke, G. (2021). Parallel incremental attribute-based encryption for mobile cloud data storage and sharing. Journal of Ambient Intelligence and Humanized Computing.

Johnson, 2020. Johnson, O. B. (2020). Crank-nicolson and modified cranknicolson scheme for one dimensional parabolic equation.

Rao, 2020. Rao, C. M. (2020). Analysis and implementation of a parallel computing cluster for solving computational problems in data analytics. 2020 5th International Conference on Computing, Communication and Security (ICCCS).