Isı Bilimi ve Tekniği Dergisi - Journal of Thermal Science and Technology 45:1 (2025) 56-68

Journal Homepage: https://dergipark.org.tr/tr/pub/isibted

Isı Bilimi ve Tekniği Dergisi

Archive: https://tibtd.org.tr/dergi

Machine Learning-Based Reduced Order Modeling for Operational Analysis of Industrial Glass Melting Furnaces Using CFD Solutions

Engin Deniz CANBAZ [1,2] *, Mesut GÜR[1]

[1] *İstanbul Teknik Üniversitesi, Makina Fakültesi, Makina Mühendisliği Bölümü, Beyoğlu, 34437, İstanbul, Türkiye*
[2] *Şişecam Bilim, Teknoloji ve Tasarım Merkezi, Modelleme ve Simülasyon Teknolojileri Müdürlüğü, Gebze, 41400, Kocaeli, Türkiye*

ARTICLE INFO

2025, vol. 45, no.1, pp. 56-68
©2025 TIBTD Online.
doi: 10.47480/isibted.1512812

Research Article
Received: 08 July 2024
Accepted: 04 December 2024

* *Corresponding Author*
e-mail: canbaze@itu.edu.tr
dcanbaz@sisecam.com

Keywords:
Glass Melting Furnaces
Computational Fluid Dynamics
Machine Learning
Reduced Order Model
Autoencoders

ORCID Numbers in author order:
0000-0002-6287-8770
0000-0002-0407-0298

ABSTRACT

Computational fluid dynamics (CFD) models are essential for analyzing industrial glass melting furnaces, providing insights into energy consumption, temperature distribution, and glass quality. However, their computational expense limits practical application in daily operations. This study addresses this issue by developing a machine-learning-based reduced order model (ROM) using parametric solution data from a CFD model of a glass melting tank of a furnace. Key operational parameters, namely pull rate, heat flux from combustion space, and electrical potential difference to supply electrical power, are chosen to create a CFD solution dataset. The ROM utilizes an autoencoder with convolutional neural networks to predict temperature and velocity fields. Operational parameters are linked to its decoder through an auxiliary neural network. The performance of the ROM is assessed for both interpolation and extrapolation. Comparison between the data generated by the ROM and the ground-truth CFD solutions indicates less than 1% deviation and an R2 score of 0.99, demonstrating the ROM's capability. The ROM achieved a mean squared error (MSE) of 2.82E-06 for the interpolation and 2.79E-05 for the extrapolation tests, further supporting its accuracy. Additionally, the ROM offers significant advancements in solution time, achieving reduction by up to three orders of magnitude, which enhances its practical utility.

HAD Çözümlerini Kullanarak Endüstriyel Cam Ergitme Fırınlarının İşletme Analizi İçin Makine Öğrenimi Tabanlı İndirgenmiş Model Geliştirilmesi

MAKALE BİLGİSİ

***Anahtar Kelimeler**:*
Cam Ergitme Fırınları
Hesaplamalı Akışkanlar Dinamiği
Makine Öğrenimi
İndirgenmiş Model
Otokodlayıcı

ÖZET

Endüstriyel cam ergitme fırınlarında enerji tüketimi, cam kalitesi, sıcaklık dağılımının belirlenmesini sağlayan hesaplamalı akışkanlar dinamiği (HAD) modelleri, fırınların tasarım süreçlerinde de kritik öneme sahiptir. Ancak, cam ergitme prosesinin geniş zaman ve uzunluk boyutları nedeniyle, bu modellerin yüksek hesaplama maliyeti günlük fırın operasyonlarında kullanılmasını kısıtlamaktadır. Bu zorluğu aşabilmek için, bu çalışmada bir cam ergitme fırınının cam banyosu HAD modelinden elde edilen sonuçlar kullanılarak, makine öğrenimi tabanlı bir indirgenmiş model geliştirilmiştir. İndirgenmiş model, çekiş hızı, doğal gaz kaynaklı ısı akısı ve elektrik potansiyel farkı sınır koşullarının değişimini içeren parametrik HAD verileri ile oluşturulmuştur. Bu veri seti ile sıcaklık ve hız alanı tahminleri yapabilecek, konvolüsyonel nöral ağ tabanlı bir otokodlayıcı eğitilmiştir. Sonrasında, otokodlayıcının boyutsal yükseltme yapan bölümü, ek bir tam bağlantılı nöral ağ aracılığıyla, işletme parametreleri olan sınır koşullarıyla ilişkilendirilmiştir. Bu iki ağın birleşimiyle indirgenmiş model elde edilmiştir. İndirgenmiş modelin performansı, interpolasyon ve ekstrapolasyon testlerinde değerlendirilmiş; test sonuçlarında %1'den az sapma, 0.99 R^2 skoru elde edilmiştir. İnterpolasyon ve ekstrapolasyon testlerinde kaydedilen ortalama karesel hata ise, sırasıyla, 2.82E-06 ve 2.79E-05'tir. Ayrıca indirgenmiş model, yeni HAD çözümü tahmin süresini üç mertebeye kadar hızlandırmıştır. Bu çalışma, makine öğrenimi tabanlı indirgenmiş modellerin operasyonel analizler için etkili bir araç olduğunu kanıtlamakta ve cam ergitme fırınlarında uygulanabilirliğini başarıyla göstermektedir.

NOMENCLATURE

ARD%	Average Relative Difference Percentage
CFD	Computational Fluid Dynamics
CNN	Convolutional Neural Network
MAE	Mean Absolute Error
MSE	Mean Squared Error
ML	Machine Learning
R^2	Coefficient of Determination
ROM	Reduced Order Model
SELU	Scaled Exponential Linear Unit
c_p	Specific Heat (J/kg.K)
k_{eff}	Effective Thermal Conductivity (W/m.K)
n	Number of Computational Cells

p	Pressure (Pa)
S_{joule}	Joule Heating Source (W/m^3)
T	Temperature (K)
y_i	Ground-Truth Solution
$\bar{y}$	Mean of Ground-Truth Solution
$\hat{y}_i$	ROM Prediction
ρ	Density (kg/m^3)
μ	Dynamic Viscosity (Pa.s)
σ	Electrical Conductivity (1/ohm.m)
v	Velocity (m/s)
φ	Electric Potential (V)

INTRODUCTION

Modern continuous glass furnaces are quite complex thermal-fluid systems involving various physical phenomena, including batch-to-melt conversion, radiation transport in the molten glass, natural convection due to thermal expansion, joule heating with AC electrodes, turbulent combustion, and thermal coupling between the combustion atmosphere and glass bath (M. K. Choudhary et al., 2010; Cravero and Marsano, 2023; Faber et al., 2020; Pigeonneau and Flesselles, 2012; Pokorny et al., 2012, 2020; Viskanta, 1994). The glass furnaces have various types such as regenerative, recuperative, and all-electric furnaces (Atzori et al., 2023; Zier et al., 2021). Continuous regenerative furnaces, the most commonly used in the glass industry, are particularly efficient in conserving energy through heat recovery from exhaust gases. However, their high energy consumption contributes to CO_2 emissions, raising concerns about global warming and the need for sustainable energy solutions. Their design is critical, as it not only influences furnace efficiency but also impacts the greenhouse gas emissions by the furnace. By applying computational fluid dynamics (CFD), these furnaces can be analyzed in depth, allowing engineers to evaluate design and operational parameters and their environmental impact.

Numerous studies focus on different aspects of glass melting and related process. For example, CFD studies have examined the electric field and heat source generated by the immersed electrodes in the glass melt and the use of bubbling to enhance mixing by momentum transfer (Daurer et al., 2022; Matsuno et al., 2008; Simcik and Ruzicka, 2015; Soubeih et al., 2015). Radiative heat transfer in the glass melt, due to its high temperature nature, and the heat transfer mechanism between the combustion space and the melt are also key areas of interest (M. K. Choudhary et al., 2018; Raič et al., 2021). These CFD simulations support the furnace design process and operation (Abbassi and Khoshmanesh, 2008; Han et al., 2022; Jebava and Němec, 2018; Li et al., 2020; Pigeonneau et al., 2023). However, while CFD models provide a comprehensive understanding, multiphysics and multiscale features of glass melting process, the time-intensive nature of model preparation and solution poses challenges for real time decision-making during furnace operation (Gao et al., 2021).

Recently, the concept of creating digital twins for glass furnaces has gained traction to enhance operational efficiency. Digital twins dynamically replicate the physical furnace in a virtual environment, potentially bridging the gap between theoretical modeling and real-time operational insights. One of the approaches creating digital twins is through physics-based models, which defines the system based on the physical laws. However, the use of physics-based models, such as CFD, encounters hurdles in the form of requisite expertise and substantial solution times. Another approach is data-driven methods. Today, large volumes of data containing valuable information about processes and operations can be gathered from industrial equipment. Using this data, along with a bidirectional flow of information between the physical and digital systems, it is possible to replicate the operation and forecast the condition of the physical system (Cassar et al., 2018; Gao et al., 2021; Pazarlioglu et al., 2022; Taşkesen et al., 2023). However, digitally replicating every system using field data is challenging due to raw data being noisy, often incomplete, and requiring significant pre-processing before being effectively used to construct data-driven models (Molinaro et al., 2021).

Addressing these challenges, model order reduction emerges as a promising solution. By significantly reducing the dimensionality of data and the physical model while retaining essential information, models with reduced order offer a pathway to swiftly assess the effects of changing operational parameters on system properties. Techniques such as Galerkin projection of the Navier–Stokes equations onto an orthogonal basis of proper orthogonal decomposition (POD) modes provide a direct link to the governing equations. However, this approach is intrusive and relies on human expertise to develop models based on a working simulation (Brunton et al., 2020; Rowley and Dawson, 2017). Additionally, these models may perform poorly when dealing with advection-dominant problems or systems highly sensitive to minor state changes (Mücke et al., 2021).

To better capture, replicate, or predict the key dynamic physical features, machine learning (ML) algorithms can be utilized (Molinaro et al., 2021). This approach combines data-driven methods with physics-based methods. Obtaining a solution with the new inputs for the underlying governing equations is much faster compared to solving the system of equations. Specifically, this simulation-based digital twin method utilizes the simulation data, covering a broad range of operating conditions, to train a machine learning model, potentially eliminating the need for further simulations (Catsoulis et al., 2022). Meanwhile, with advancements in simulation capabilities and experimental techniques, fluid dynamics is evolving into a data-rich field, making it suitable for ML algorithms (Brunton et al., 2020). A significant part of the progress can also be attributed to the accessibility of open-source software platforms like PyTorch (Paszke et al., 2017) and Tensorflow (Abadi et al., 2015). These platforms have simplified the process of implementation and training, eliminating the need for specialized knowledge. Furthermore, the advancement of computation-accelerating hardware, such as GPUs, has made it feasible to train extremely large models (Mücke et al., 2021). However, the black-box nature of machine learning methods, which often lack physical interpretability, is making it difficult to analyze results based on the underlying physics of the problem. Enhancing the physical interpretability of these models and incorporating the problem's physics into them are still challenging tasks (Masoumi-Verki et al., 2022).

The successful implementation of machine learning methods into physics-based models highlights the potential for integrating similar techniques in other engineering disciplines. Zhang et al. developed a surrogate model for structural seismic response prediction using a physics-guided convolutional neural network (Zhang et al., 2020). The developed deep neural network utilizes existing physics, such as the laws of dynamics, to mitigate overfitting and reduce the requirement for large training datasets. The effectiveness of the proposed method was demonstrated through both numerical and experimental data, with the authors suggesting that the proposed algorithm can be scaled to other structures and to other types of hazard events. Similarly, Cho et al. applied a comparable approach to predict lithium-ion battery lifetime (Cho et al., 2022). They developed a physics-informed neural network by incorporating the energy balance law into the loss function to predict battery temperature with limited data. They plan to expand the study to include actual electric-vehicle driving profiles. Pfaff et al. developed a mesh-based graph network to address physics simulations which takes the advantage of adaptive meshing (Pfaff et al., 2020). This technique overcomes the limitations of grid-based methods (such as convolutional neural networks) by learning predictions on existing computational meshes and it can be applied on a wide range of physical systems, from simulating cloth and structural mechanics to fluid dynamics. The predictions by the trained network showed that acceleration of 1-2 orders of magnitude, faster than the simulation it was trained on.

Meanwhile, machine learning methods in the development of reduced order models (ROM) using CFD solutions have also gained popularity in recent years. Erichson et al. studied shallow decoders to reconstruct flow field from limited sensor data (Erichson et al., 2019). They achieved superior performance with fully connected shallow networks compared to conventional model approximation methods, such as proper orthogonal decomposition, while utilizing fewer sensors. He et al. proposed a ROM for fast prediction of natural convection of nanofluids in annulus pipes (He et al., 2022). They utilized U-net structure of deep convolutional neural networks (CNNs). The Nearest Wall Signed Distance Function is combined with numerical data to train the network for better geometry adaptation. A two-dimensional annulus is utilized to investigate the feasibility, accuracy, and stability of the proposed method. They reported that predicting the velocity field is somewhat more challenging than predicting the temperature field. However, error analysis showed that the proposed network model still achieves satisfactory results, with an average error of less than 1%. Wiewel et al. studied the temporal progression of physical functions of fluid flow problems using deep learning (Wiewel et al., 2019). Their approach uses the combination of learned latent spaces with Long Short-Term Memory (LSTM) to predict the temporal changes in the field solutions. To show the capabilities of the method, they predicted the temporal solutions of a series of liquid simulations and a set of single-phase buoyancy simulations. Using the developed data-driven solver, they achieved 150 times faster simulation time than a standard pressure solver. Kim et al. utilized CNN networks to train on various transient CFD simulations enabling the generation of fluid simulations from the reduced parameters (Kim et al., 2019). They ensured the condition of divergence-free is strictly met using stream functions. The resulting ROM facilitated the interpolation of CFD solutions and time advancement of an existing simulation. Authors also stated up to 700 times faster computation time compared to a CFD simulation. Thuerey et al. extended the application of ROM models to Reynolds-Averaged Navier-Stokes solutions of airfoil flows (Thuerey et al., 2020). They utilized U-net deep learning architecture using physics informed feature normalization to learn from velocity and pressure distribution, and to predict the field solution for unknown airfoil shapes. The paper also explored the impact of training data size and the number of weights on the model accuracy. In the context of glass furnace solutions, a similar application of machine learning methods has been investigated to enhance furnace operation support (Castillo and Kornish, 2017). The authors reported 98% accuracy in generating flow field and post-processed outputs. This body of work collectively demonstrates the growing significance of leveraging machine learning techniques to advance the capabilities of CFD-based reduced order models across diverse engineering domains.

This study focuses on leveraging machine learning methodologies, particularly autoencoders integrated with CNNs, applied to parametric CFD solution data. The primary objective is to represent the complex dynamics of the glass furnace CFD solution, consisting of 2D velocity and temperature fields, in lower dimensions within a latent space, and to predict new CFD solutions by using this reduced domain and up sampling of the decoder network. This approach enables rapid and accurate decision support for daily furnace operations using ROM-generated CFD solutions. The ROM developed using 2D solutions can serve as a baseline for developing a ROM for a full-scale furnace. The novelty of this study lies in the fact that, although ML-based ROMs have been applied to other thermal flow problems, to the authors' knowledge, this is the first time such a model has been applied to a glass furnace with a fully detailed process explanation. Additionally, this study explains the operational benefits of the reduced order model with well-detailed examples.

CFD MODEL

A CFD model of an end-fired glass melting furnace was constructed to generate training and test dataset for the machine learning-based ROM development process. Ansys Fluent 2022R2 is used to prepare the CFD model and to obtain parametric solutions. The results from this CFD model are used to train the ROM and evaluate the accuracy of the ROM-generated CFD predictions by comparing them with the ground-truth CFD solutions. For the sake of simplicity and computation time, a 2D representation of glass melting tank is generated. The schematic view of the furnace and the plane, which is the solution geometry, is shown in Figure 1. This simplified model focuses on the essential features of an industrial glass melting furnace, specifically targeting the middle plane of the melting tank. Industrial type glass furnaces are typically more elongated in length and width compared to their height. Therefore, primary flow patterns occurs in the longitudinal direction and it is assumed that the problem can be studied as a long 2D cavity (Pigeonneau and Flesselles, 2012). The 2D model in this study includes raw batch melting interface and outlet section, where the molten glass proceeds to undergo further conditioning processes.

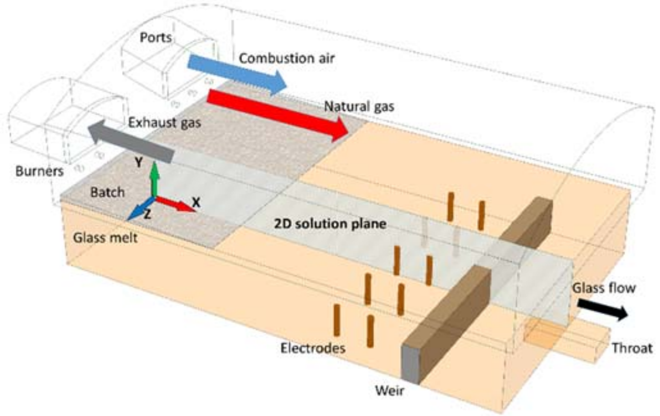


Figure 1. The schematic view of an end-fired glass melting furnace. The top part represents the combustion chamber, while the bottom part illustrates the glass melting tank. The 2D solution plane, which is the focus of the CFD study, is the middle plane of the melting tank along the flow direction.

In glass melting tanks, due to temperature differences within the domain, natural convection is dominant, and it creates characteristic flow currents, which enhances the homogenization process. In this 2D CFD demonstration, we aim to create the flow patterns with the presence of temperature differences; 1 - underneath the batch blanket and open melt surface and 2 - top and bottom sections of the glass domain. With the presence of the unique flow patterns, 2D CFD solutions can provide valuable insights into the operational dynamics of the glass melting furnace.

Governing Equations

The molten glass flow inside the furnace is mathematically described by the conservation Eq. 1-4 listed below (ANSYS Inc., 2022). Laminar flow is assumed due to the high viscosity of the melt under the operating conditions. Gravity source term must be added into the momentum equation to resolve buoyancy driven flow patterns (Schill and Chmelar, 2004).

Due to the high-temperature nature of the glass melt, radiative heat transfer is dominant within the domain. However, if complete absorption of radiation over the depth of the glass melt is assumed, the Rosseland approximation method can be used (Lankhorst et al., 2013). In this study, the glass melt is assumed to be optically thick, allowing the use of the Rosseland approximation (M. K. Choudhary and Potter, 2005). Effective thermal conductivity (k_{eff}) is used in the energy equation, which combines the effects of conductive and radiative heat transfer rates. Furthermore, the energy conservation equation introduces a source term (S_{joule}) to specifically address the joule heating effect resulting from electric current flow within the melt (Eq. 5) (ANSYS Inc., 2022; Matsuno et al., 2008).

To compute electric current, additional electric potential (φ) equation is solved, which describes potential field induced by electrodes immersed in the melt (M. Choudhary, 1985). The gradient of the electric potential yields the electrical current field. Natural Lorentz force is induced by the interaction between current density and magnetic flux in the vicinity of electrodes. This force acts only in the tiny area around the electrode tips and it is weaker compared to other forces such as gravitational and viscous forces. Therefore, it is ignored (Soubeih et al., 2015; Staněk, 1990). The product of the square of the electric current and electrical conductivity (σ) of the glass melt quantifies the joule heating output (Simons et al., 2008). High temperature thermophysical properties of the glass melt are given in Table 1.

$$\nabla . \vec{v} = 0 \tag{1}$$

$$\rho(\vec{v}.\nabla)\vec{v} = -\nabla p + \mu\nabla^2\vec{v} + \rho\vec{g} \tag{2}$$

$$\rho c_p(\vec{v}.\nabla)T = k_{eff}\nabla^2 T + S_{joule} \tag{3}$$

$$\nabla.(\sigma\nabla\varphi) = 0 \tag{4}$$

$$S_{joule} = \sigma|\nabla\varphi|^2 \tag{5}$$

Table 1. Thermophysical properties of the glass melt.

Density (kg/m³)	$\rho(T) = 2521 - 0.138T(K)$
Specific heat (J/kg.K)	1300
Effective thermal conductivity (W/m.K)	$k_{eff} = 556 - 0.87T(K) + 0.00036T(K)^2$
Viscosity (Pa.s) (M. K. Choudhary et al., 2018)	$\log(\mu) = -2.834 + \dfrac{4678.7}{T(K) - 504.6}$
Electrical conductivity (1/ohm.m)	$\sigma = -45.4 + 0.04T(K)$

Geometry and Mesh

A two-dimensional geometry is prepared to demonstrate change of velocity and temperature fields with the operational parameters (Figure 2). This is the middle plane of a typical glass packaging furnace including a batch blanket interface, a couple of electrode rods immersed from the bottom, a weir for flow separation between melting and refining sections, and an outlet. Batch partially covers the melt surface. That predetermined section serves as the inlet face. The rest of the top surface is open to combustion atmosphere. Therefore, constant heat flux boundary condition is applied on the open section. Usually, electrodes are inserted from the bottom into the furnaces. For simplicity, two electrodes are positioned before the weir to generate joule heating with the potential differences applied on their corresponding surfaces. Weirs are usually placed to adjust flow currents between the melting and refining zones. Following the weir is a throat which glass melt leaves through. There are additional structures such as working end and forehearth for further conditioning of the glass melt before molding. However, these structures are beyond the scope of this study. Essential dimensions of the glass melting tank are listed in Table 2. The dimensions are selected based on a packaging glass furnace with a specific pull rate of 2.5 t/day.m². This value corresponds to 100 t/day of molten glass for a furnace with 60 m² of melting area. Discretization uncertainty in the CFD model is evaluated following the method described by (Celik et al., 2008). The numerical uncertainty for the mesh in this study is calculated by comparing the converged temperature values at the outlet, resulting in an approximate relative error of 0.88% and an extrapolated relative error of 2.06%.

Table 2. Dimensions of the glass melting tank

Length	8000 mm
Glass depth (height)	1245 mm
Electrode diameter	100 mm
Electrode height	600 mm
Outlet depth	600 mm
Outlet opening	600 mm
Weir thickness	350 mm
Weir height	645 mm

Boundary Conditions and Parameters

Boundary conditions for CFD simulations are listed in Table 3. The inlet condition is constant velocity at the batch-melt interface derived from the furnace pull rate. It is assumed that batch is converted into glass melt at a constant temperature, and this constant temperature applied at the inlet boundary. A shear-free surface condition is applied at the rest of the top surface and heat flux boundary condition is applied to model the effect of the combustion atmosphere above. Heat loss at the walls of the tank is much smaller compared to the heat flux from the combustion space due to insulation layers installed on the furnace. Therefore, walls are assumed to be perfectly insulated in this CFD setup. The electric potential is kept at 0 V at the first electrode, while a positive potential value is applied on the second to create a potential difference within the domain, influencing the electric current distribution and, consequently, the joule heating effect. Electric current flux is assigned as zero on the other surfaces of the melting tank.

To generate a dataset for the training, validation and test of the ROM, three key parameters are selected. The first one is the inlet velocity, reflecting changes in the pull rate of the furnace. Second is the heat flux at the top surface. This parameter represents variations in fuel consumption within the combustion space. The third parameter is the potential difference between the electrodes capturing variations in the power output of the electrodes. The higher the electric potential between the electrodes, the more joule heating is generated. The values for these parameters are specified in Table 4.

Through systematic changes in these parameters, a total of 27 CFD solutions is obtained, constituting the training and validation dataset for the ROM development. Computation time of each case is approximately 30 minutes running on single central processing unit (CPU). The convergence of each solution is also determined based on whether the relative change in the temperature monitor at the outlet is below 1E-05, in addition to the convergence criteria of the residuals.

This dataset will serve as the foundation for training the ROM, enabling efficient exploration of the complex interactions within the glass furnace under diverse operational conditions. Test dataset is generated using the same model with the same operational parameters. Two sets of data are generated. The first one, namely interpolation cases, consists of solutions of parameters within the specified operational range. The second set, extrapolation cases, involves solutions that at least one parameter is out of the specified operational range. The details about the interpolation and extrapolation cases are given in the results section.

Table 3. Boundary conditions applied in the CFD model.

	Mass-Momentum	**Energy**
Inlet	3.01E-05 – 3.68E-05 m/s	1200 K
Outlet	0 Pa	1500 K for reverse flow
Top surface	Free surface	61.0-73.2 kW/m^2
Walls	No slip	0 W/m^2

Table 4. Operational parameters and their values. By interchanging the values, 27 CFD solutions are generated to train the ROM.

	Inlet velocity (m/s)	**Heat flux (W/m^2)**	**Electric potential (V)**
1	3.01E-05	61000	75
2	3.31E-05	67100	88
3	3.68E-05	73200	100

REDUCED ORDER MODEL

Convolutional Neural Networks (CNNs) have been selected as the foundation for developing the machine learning-based ROM due to their efficacy in reducing dimensionality with significantly fewer weights (network parameters) compared to fully-connected neural networks. This makes CNNs better suited for large-scale and high-dimensional problems (Gao et al., 2021). Another benefit of CNNs is their ability to leverage low-dimensional, high-level abstractions through convolution. The central concept of a CNN is to learn these representations and subsequently use a fully connected layer to map the relationship between the high-level representations and the output (Bhatnagar et al., 2019). The ROM leverages the inherent capabilities of CNNs to streamline the representation of complex field solutions. CFD solutions are sampled with a uniform grid over the solution domain, due to a major limitation of CNN. This limitation lies in the fact that traditional CNNs and their convolution operations were initially developed for processing images, which are sampled on a uniform grid (Chollet, 2021). However, problems with irregular domains are making the use of CNNs inapplicable and this constraint greatly restricts their use in general scientific problems. Rectangular shape of glass furnaces, especially the glass melting tanks, provides the opportunity to apply convolutional filters.

A deep convolutional autoencoder network is constructed to systematically reduce the dimensionality of the field solution of u and v velocity components and temperature, condensing it into a latent space, and subsequently predicting the u and v velocity, and the temperature fields from this reduced representation. This bow-tie structure of autoencoder is comprised of two integral components: the encoder and the decoder. The encoder's role is to efficiently compress the full-sized field data into a smaller, yet information-rich representation contained within the latent space. During the encoding phase, the size of the image is systematically reduced by half using strided convolutions. This process enables the network to capture information that is progressively more general and conceptual across an expanding set of feature channels (Thuerey et al., 2020). Conversely, the decoder is tasked with reconstructing a full-sized CFD-like data from the essential information stored in the latent space. The decoding phase of the network reverses the data reduction process, enhancing the image's spatial detail through up sampling layers while simultaneously decreasing the quantity of feature channels.

The latent space, essentially a low-dimensional representation of the CFD solution, captures the essential features required for accurate predictions. To establish the relationship between changes in this low-dimensional representation and the corresponding alterations in boundary conditions (operational parameters), an auxiliary fully-connected neural network is introduced. This auxiliary network facilitates mapping of boundary conditions to the latent space, establishing the ROM's ability to predict the full flow field solution under varying operating conditions. Schematic representation of the described model is given in Figure 3.

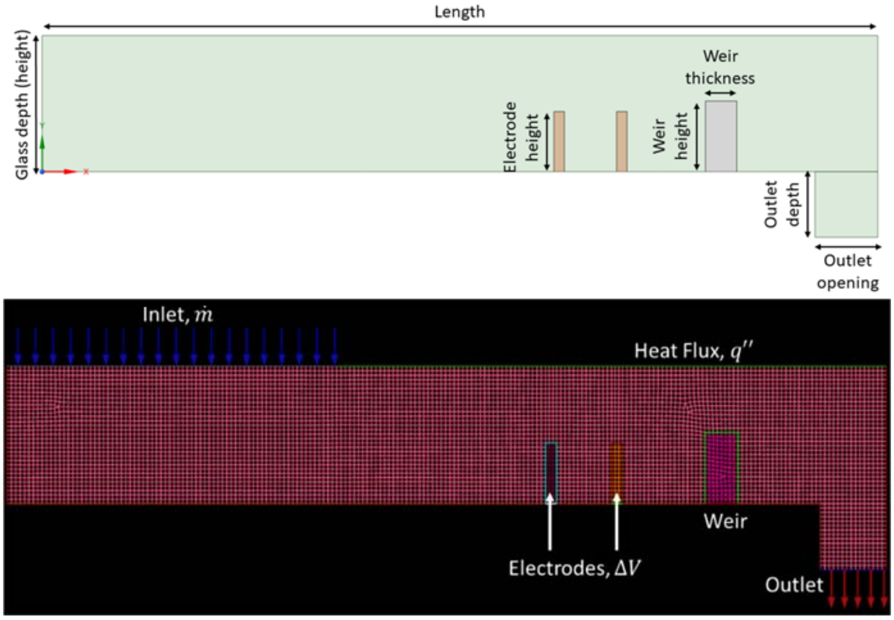


Figure 2. Geometry and mesh of the computational domain. The computational domain consists of glass melting tank including two electrodes, a weir, and a submerged outlet section.

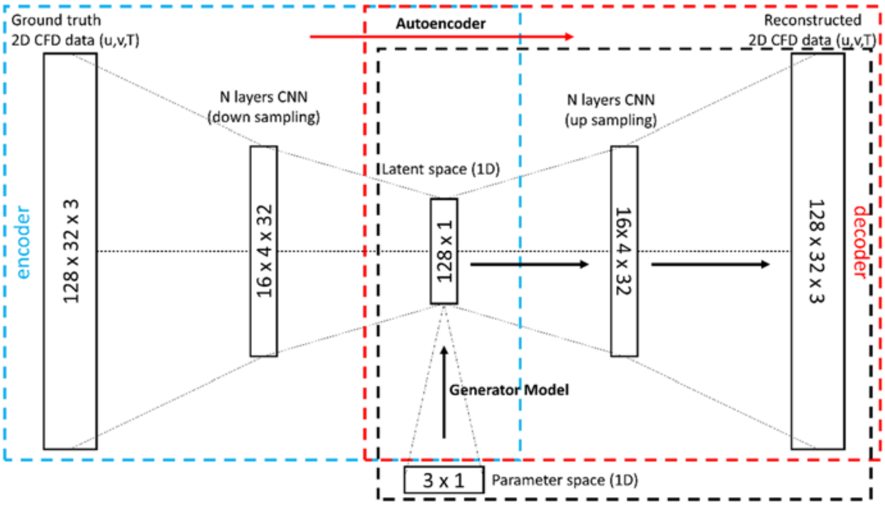


Figure 3. The schematic representation of the autoencoder and generator. The autoencoder takes the structured CFD solution data as input and aims to reconstruct the full CFD solution through its encoder and decoder sections. The generator model uses an auxiliary network, which establishes connection between parameter space (boundary conditions – operational parameters) and the latent space, and pre-trained decoder network to predict the CFD solution corresponding to the given boundary conditions.

Steps taken in this study to develop a successful machine learning-based ROM to predict CFD results using a parametric CFD solution dataset are given here:

1) A CFD model of the process is created.
2) Operational parameters and their range (boundary conditions for the CFD model) are defined. These parameters will be the inputs of the ROM.
3) Parametric CFD solutions are obtained based on the defined parameter range. Optionally, a design of experiments (DOE) method may be employed in this step for an efficient dataset generation.
4) From the parametric CFD solutions, a supervised training dataset is prepared. The CFD solutions are sampled to a 128×32 grid for CNN compatibility. The dataset is then split into training and validation dataset to prevent overfitting.
5) An autoencoder network is constructed using CNNs. This reconstructs CFD solutions by down sampling and up sampling the actual CFD data. This network is trained with the parametric CFD solution dataset.
6) A generator network is built by linking an auxiliary neural network, which connects the operational parameters to the latent space, with the pre-trained decoder from the autoencoder. This network takes operational parameters as inputs and predicts the corresponding CFD solution. During training, only the auxiliary network's weights are optimized. Once trained, this generator network acts as the ROM, predicting CFD solutions based on the input parameters.
7) Performance of the generator network (the ROM) is evaluated through interpolation and extrapolation tests. New set of parameters are defined for testing, and corresponding CFD solutions are obtained. The ROM's predictions for these new parameters are compared to the ground-truth CFD solutions.

Autoencoder

The input to the autoencoder consists of a 128x32 matrix with 3 channels representing u-velocity, v-velocity, and temperature. The CFD solutions are sampled with 128 nodes in X-direction and 32 nodes in Y-direction. Then each field solution concatenated to form the channels of input matrix. Prior to entering the autoencoder, each field is normalized to have values between 0 and 1. The normalization process involves using the minimum and maximum values from the training dataset for each field. This ensures consistent normalization for both training and test datasets.

A comprehensive study on latent space dimensions revealed that a latent space reduced to 128 dimensions is adequate for this specific problem. This dimensionality is found to be effective in extracting and retaining essential information about flow and temperature patterns from the CFD training data. The decoder, structured symmetrically to the encoder, takes an input of 128 variables and generates an output with the same dimensions (128x32 matrix with 3 channels). During the supervised training phase, the decoder is optimized to produce an output with minimal error compared to the input CFD data provided to the encoder. Mean squared error (MSE) loss function is used during training and mean absolute error (MAE) metric is also monitored. MSE and MAE is given by Eq. 6 and 7, respectively, where n is number of cells within the domain, y_i is the ground-thruth solution, and $\hat{y}_i$ is the solution predicted by the ROM (Abadi et al., 2015). Training dataset is split into training and validation sets to prevent overfitting during training, by selecting 5 CFD solutions out of 27 solutions for validation. MSE value of 6.55E-07 and MAE value of 5.85E-04 are obtained for the training data, while validation data resulted in an MSE of 2.10E-06 and MAE of 9.08E-04. After training, coefficient of determination (R^2) and average relative difference percentage (ARD%) scores are calculated using Eq. 8 and 9, respectively (Abooali and Khamehchi, 2019; Pedregosa et al., 2011). An R^2 score of 0.99 and ARD% of 0.20% are obtained for the validation cases. The supervised training runtime was 45 minutes completing 20k epochs using V100 graphics processing unit (GPU). The MSE changes over epochs for both training and validation data are given in Figure 4.

$$MSE = \frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2 \tag{6}$$

$$MAE = \frac{1}{n}\sum_{i=1}^{n}|y_i - \hat{y}_i| \tag{7}$$

$$R^2 = 1 - \frac{\sum_{i=1}^{n}(y_i-\hat{y}_i)^2}{\sum_{i=1}^{n}(y_i-\bar{y})^2} \tag{8}$$

$$ARD\% = \frac{100}{n}\sum_{i=1}^{n}\left|\frac{y_i-\hat{y}_i}{y_i}\right| \tag{9}$$

The layers of the autoencoder (encoder + decoder), with the input/output dimensions of each layer, are detailed in Table 5 and Table 6, respectively.

Hyperparameter optimization is a crucial step, involving the selection of appropriate activation functions, adjustment of the number of convolution filters, and tuning the kernel size of the filters. Additionally, strides are employed to downsize the input of the layers, with a stride of 2 halving the dimension in both x and y directions, ultimately reducing the size to 2. The ROM's training and test evaluation are completed using the best resulting hyperparameters, as listed in Table 5 and Table 6. The resulting network's encoder section has 7 CNN layers and a flattening layer to adjust the form of output matrix into a 128-dimensional vector. Symmetrically, the decoder section starts with a reshaping layer to transform the input into the correct form and then, 7 CNN layers follow to increase the dimensions to the sampled-CFD data size. The resulting autoencoder network has 123,663 weights (network parameters) to be optimized during training process.

Table 5. Encoder network parameters. There are 7 convolutional neural network layers and 1 reshaping layer to flatten the output tensor.

Input/Output				Kernel	Strides
x	y	Channel	Activation	x/y	x/y
128	32	3	SELU	4/4	1/1
128	32	64	SELU	4/4	2/2
64	16	32	SELU	3/3	2/2
32	8	32	SELU	2/2	2/2
16	4	32	SELU	2/2	2/1
8	4	32	SELU	2/2	2/1
4	4	32	SELU	2/2	2/2
2	2	32	SELU	2/2	-
-	-	128	-	-	-

Table 6. Decoder network parameters. The decoder network gradually increases the dimensions using transposed CNN networks, and finally reconstructs full-sized CFD data.

Input/Output				Kernel	Strides
x	y	Channel	Activation	x/y	x/y
-	-	128	-	-	-
2	2	32	SELU	2/2	2/2
4	4	32	SELU	2/2	2/1
8	4	32	SELU	2/2	2/1
16	4	32	SELU	2/2	2/2
32	8	32	SELU	3/3	2/2
64	16	32	SELU	4/4	2/2
128	32	64	Sigmoid	4/4	1/1
128	32	3	-	-	-

Generator Network: Auxiliary Neural Network + Decoder

Following the training phase of the autoencoder, an auxiliary dense neural network has been introduced before the decoder network. Fully-connected, dense neural network was utilized to express the nonlinear relationship between the boundary conditions and the latent space (Figure 3), consequently, the CFD solution. The auxiliary network consists of two layers with 2240 weights to be trained. The network takes a set of boundary conditions, structured as a 3x1 vector, as input and outputs a latent space vector sized 128x1, (Table 7). As mentioned previously, the primary objective of this auxiliary network is to establish a connection between the boundary conditions (parameters) and the latent space. To achieve this, an additional supervised training step is implemented to optimize the weights of the auxiliary network.

Table 7. Parameters of the auxiliary dense neural network placed before the pre-trained decoder. The auxiliary network establishes connection between operational parameters (boundary conditions) and latent space of the autoencoder network, which is also the input of the decoder network.

Layers	Input	Output	Weights
1	3	16	48
2	16	128	2048

It is important to note that, during this training process, the weights of the decoder network are kept constant. This decision is made because the decoder has already undergone training and optimization to generate accurate CFD solution data from the latent space. Keeping the decoder weights fixed ensures that the learned representation of the latent space is preserved and consistently utilized during the training of the auxiliary network. Training dataset is again split into training and validation datasets to avoid overfitting. MSE loss function is used during training and MAE is also monitored. MSE value of 1.94E-06 and MAE value of 9.96E-04 are achieved for the training data. MSE value of 4.45E-06 and MAE value of 1.30E-03 are achieved for the validation data. R^2 and ARD% scores of 0.99 and 0.27% are obtained for the validation dataset, respectively. Runtime of the supervised training of the auxiliary + decoder network was 31 minutes, completing 30k epochs during that time, using V100 GPU. The change of MSE loss over epochs is given in Figure 5.

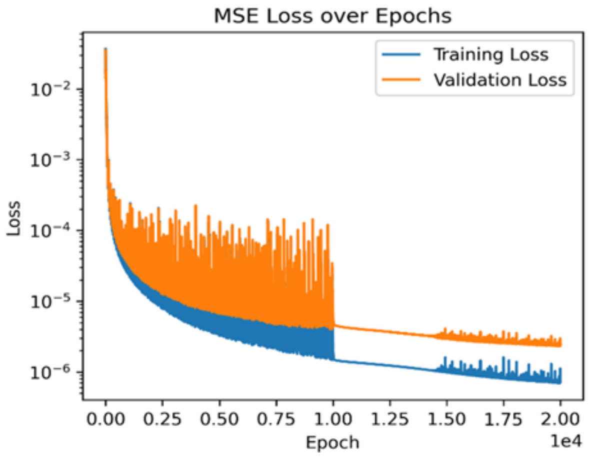


Figure 4. Change of Mean Squared Error (MSE) over epochs for training and validation dataset during supervised training of the autoencoder network. Y-axis in logarithmic scale.

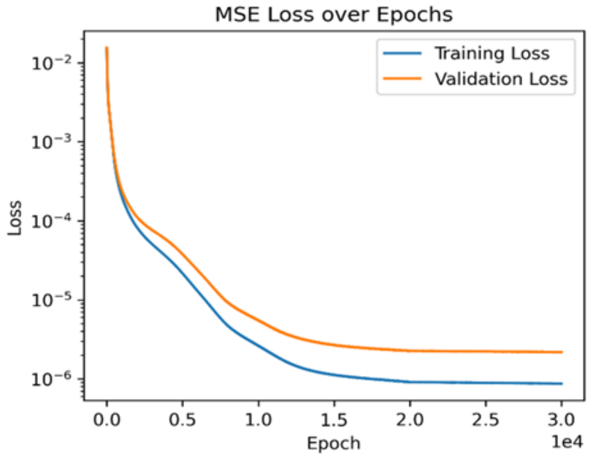


Figure 5. Change of Mean Squared Error (MSE) over epochs for training and validation dataset during supervised training of the generator: auxiliary neural network + decoder network. Y-axis in logarithmic scale.

RESULTS

The ROM's performance is rigorously assessed through both interpolation and extrapolation evaluations using an additional CFD dataset. Both interpolation and extrapolation cases are predicted less than a second using the generator (auxiliary + decoder) network, running on a V100 GPU. In the interpolation test phase, seven cases with randomly generated parameters are selected within the parameter range utilized during the training phase. These cases, detailed in Table 8, are aimed at showcasing the ROM's ability to consistently produce continuous and reliable results across various operational scenarios. This comprehensive demonstration illustrates the ROM's potential as a robust tool for supporting furnace operation, highlighting its capability to generate meaningful outputs within the specified parameter range.

For the extrapolation test, eight distinct cases, as listed in Table 9, are considered. The first two cases involve changes in potential difference beyond the training range, while keeping other parameters at the midpoint of their intervals. The subsequent two cases simulate variations in heat flux parameters, one below and one above the trained range. Case #5 and Case #6 represent changes in the inlet velocity outside of the training range. Finally, Case #7 and Case #8 examine scenarios where all parameters are set at their lowest and highest values, respectively. This selection of parameters for the extrapolation test simulates the ROM's performance when the furnace is operated outside the specified range – a scenario that may occur during actual operations due to various reasons.

Solution data predicted by the ROM for interpolation and extrapolation cases is compared with ground-truth CFD solutions. The error between the predicted and actual solutions is evaluated using several metrics and they are presented in Table 10. As expected, performance of the ROM in the extrapolation test, where the inputs are out of the specified operational range, is lower, than in the interpolation test. This is reflected in the increase in MSE and MAE values. While R^2 score change is negligible, the rise in ARD% score also indicates greater error in the extrapolation cases.

Table 8. Operational parameters of the interpolation cases to analyze interpolation performance of the ROM.

Case No.	Inlet velocity (m/s)	Heat flux (W/m²)	Electric potential (V)
1	3.08E-05	62140	80
2	3.08E-05	72059	93
3	3.08E-05	72059	80
4	3.62E-05	62140	93
5	3.62E-05	62140	80
6	3.62E-05	72059	93
7	3.62E-05	72059	80

Table 9. Operational parameters of extrapolation cases to evaluate the performance of the ROM outside the defined operational range.

Case No.	Inlet velocity (m/s)	Heat flux (W/m²)	Electric potential (V)
1	3.35E-05	67100	70
2	3.35E-05	67100	105
3	3.35E-05	59000	88
4	3.35E-05	75000	88
5	2.90E-05	67100	88
6	3.80E-05	67100	88
7	2.90E-05	59000	70
8	3.80E-05	75000	105

Table 10. Error metrics of the solutions predicted by the ROM for the interpolation and extrapolation test parameters.

Metrics	Interpolation Test	Extrapolation Test
MSE	2.82E-06	2.79E-05
MAE	1.24E-03	2.90E-03
R^2	0.99	0.99
ARD%	0.21%	3.59%

Interpolation Test Results

The results from the interpolation test dataset are summarized in Table 11. MAE and MSE metrics are computed for each test case. Notably, the average MAE value across all seven cases is consistently below 0.01 for all variables. Figure 6 shows the histogram of the relative difference between the ROM's interpolation predictions and ground-truth CFD solutions. The error distribution approximates a normal distribution centered around 0.0, with the majority of errors falling below 0.005. The maximum error is observed in Case #2, where the heat flux is close to the highest value in the parameter range, and the potential difference above the midpoint value. Conversely, Case #7 exhibits the lowest prediction error among all test cases. Electric potential seems to influence this outcome; as higher values corresponds to larger prediction errors. The comparison between Case #4 and #5 confirms this conclusion. In general, a comparative analysis of prediction error of the cases indicates that cases with higher inlet velocity tend to have lower error. The MAE comparison between Case #2 and #6, and Case #3 and #7 supports this outcome.

Ground truth, generated field data, and error contours are presented in Figure 8 for u-velocity and temperature fields, focusing on Case #2 due to its larger error.

A detailed examination of the u-velocity field reveals distributed errors across the domain. However, high concentration of maximum and minimum errors occurs in regions associated with strong forward and backward flows – specifically, before the weir and beneath the inlet boundary condition, respectively. These types of errors are also reported by other studies, which link them to higher gradients in those regions (He et al., 2022; Thuerey et al., 2020). Despite these localized errors, the overall velocity patterns are accurately predicted, and contour lines exhibit a smooth profile. This suggests that the gradient of the predicted field can be effectively utilized for post-processing applications.

Similarly, a comparison is conducted for the temperature field, again focusing on Case #2. The generated temperature field closely resembles the ground-truth solution, albeit with more discontinuous behavior in the contour lines compared to the U-velocity field. However, peak error values for temperature are lower than those observed in the U-velocity field, and errors are more uniformly distributed across the domain.

Extrapolation Test Results

MAE and MSE values for all eight extrapolation test cases are compiled in Table 12. The error histogram for extrapolation cases is shown in Figure 7. While the error between the prediction and the ground-truth is still centered around 0.0, it is skewed and displays a wider range of both negative and positive errors. Case #8, characterized by the highest parameter values, exhibits the highest error for all field variables. In contrast, Case #7, with all parameters set to their minimum values, displays relatively lower error values. However, Case #2 stands out with the second-highest error in Mean Absolute Error (MAE), despite having only one extreme parameter value. When Case #7 and #8 are excluded, Case #1 and #2 exhibit the maximum error, with minimum and maximum electric potential as the input parameters, respectively. Changes in electric potential beyond the defined operational range notably increase the prediction error of the model. The ground truth field data, ROM-predicted field data, and error contours for Case #2 are presented in Figure 9 for u-velocity and temperature fields.

Comparing the generated u-velocity field with the ground-truth solution in Figure 9 reveals less similarity than observed in the interpolation test. False predictions are evident at the core of the forward and backward flows, yet the overall error for this case remains under 1% (MAE). The general flow patterns are generated without major issues, showcasing the model's ability to extrapolate beyond the trained parameter range. Stronger discontinuous behavior is observed in the contour lines for the generated temperature field compared to the results obtained in the interpolation test. Maximum error occurs at the bottom of the weir, where the temperature is over predicted, while temperature values in the vicinity of the outlet section are under predicted. Once again, the overall calculated error remains below 1% (MAE) for the extrapolation Case #2, indicating that the model maintains reasonable accuracy even in extrapolation scenarios.

Error distributions given in Figure 8 and Figure 9 show larger localized error for the velocity field than for the temperature field. However, MAE and MSE values for the velocity field are lower compared to those for the temperature field, as presented in Table 11 and Table 12. This finding also contradicts another result reported in the literature (He et al., 2022). As noted in the introduction section, machine learning methods, such as CNN, often function as black boxes, making it difficult to interpret results based on the underlying physics of the problem. Enhancing the physical interpretability of these models and incorporating the problem's physics into them remain a significant challenge (Masoumi-Verki et al., 2022).

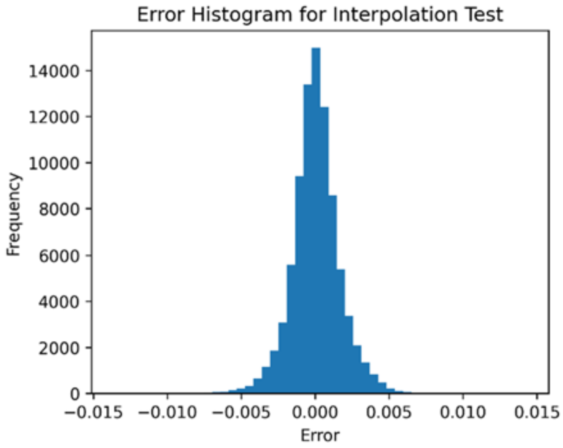


Figure 6. The histogram of differences between the normalized ground-truth interpolation CFD data and the predicted solution data displays a normal distribution.

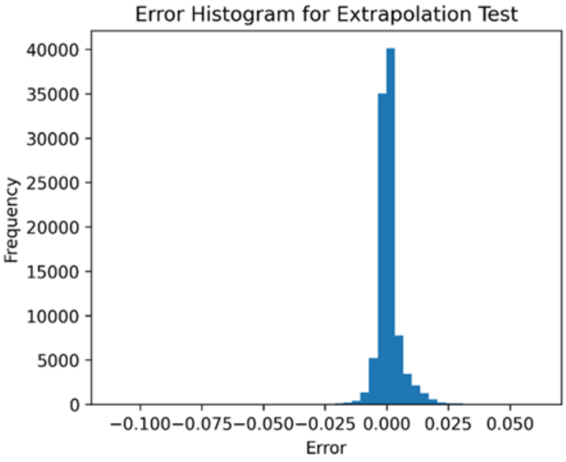


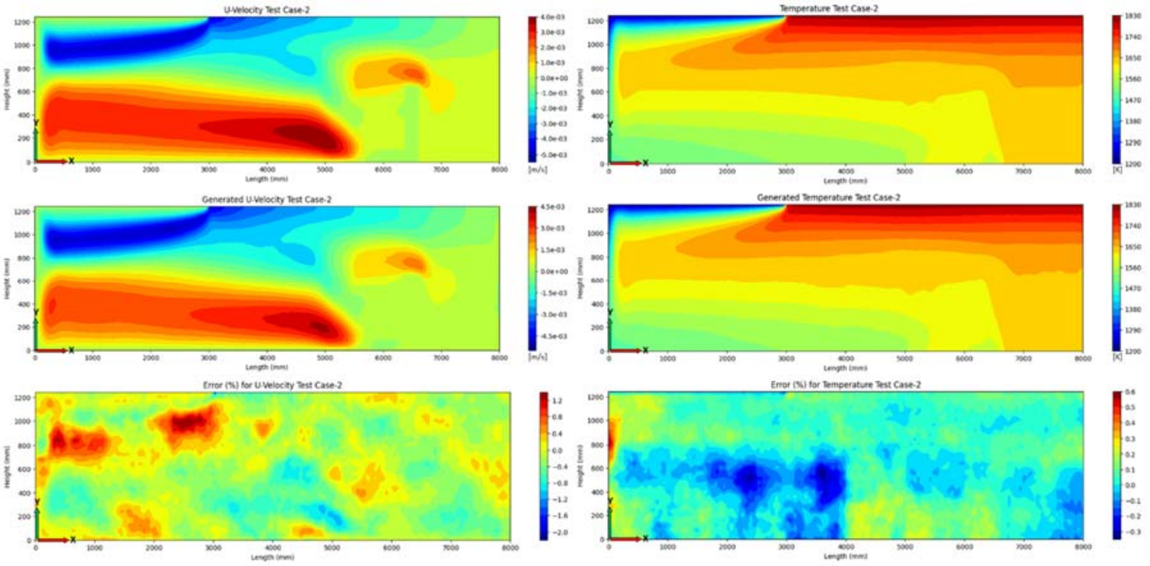
Figure 7. The histogram of differences between the normalized ground-truth extrapolation CFD data and the predicted solution data shows a skewed distribution. Larger negative and positive error values are present compared to those observed in interpolation test

Table 11. MAE and MSE obtained for each interpolation case and variable.

Interpolation Case No.	U-velocity		V-velocity		Temperature	
	MAE	MSE	MAE	MSE	MAE	MSE
1	1.21E-03	2.29E-06	7.63E-04	1.01E-06	1.94E-03	5.73E-06
2	1.96E-03	6.63E-06	9.84E-04	2.22E-06	1.93E-03	6.27E-06
3	1.12E-03	2.00E-06	7.90E-04	1.14E-06	1.49E-03	3.47E-06
4	1.33E-03	2.91E-06	8.55E-04	1.41E-06	1.35E-03	2.73E-06
5	1.19E-03	2.22E-06	7.25E-04	9.17E-07	1.82E-03	5.26E-06
6	1.32E-03	3.16E-06	8.20E-04	1.21E-06	1.58E-03	3.99E-06
7	9.83E-04	1.51E-06	6.75E-04	8.09E-07	1.22E-03	2.33E-06
Average	1.30E-03	2.96E-06	8.02E-04	1.25E-06	1.62E-03	4.25E-06
Deviation	2.02E-04	1.10E-06	7.25E-05	3.26E-07	2.37E-04	1.28E-06

Table 12. MAE and MSE obtained for each extrapolation case and variable.

Extrapolation Case No.	U-velocity		V-velocity		Temperature	
	MAE	MSE	MAE	MSE	MAE	MSE
1	1.70E-03	4.63E-06	9.58E-04	1.80E-06	3.99E-03	2.44E-05
2	7.17E-03	1.06E-04	2.76E-03	3.37E-05	5.55E-03	4.99E-05
3	1.45E-03	3.28E-06	9.83E-04	1.90E-06	2.46E-03	1.35E-05
4	1.42E-03	3.62E-06	8.93E-04	1.70E-06	1.56E-03	7.18E-06
5	1.08E-03	2.21E-06	7.91E-04	1.33E-06	1.41E-03	7.08E-06
6	1.16E-03	2.36E-06	8.19E-04	1.72E-06	1.90E-03	9.52E-06
7	3.51E-03	2.05E-05	1.79E-03	6.32E-06	4.68E-03	3.82E-05
8	8.93E-03	1.52E-04	3.60E-03	5.95E-05	9.07E-03	1.17E-04
Average	3.30E-03	3.69E-05	1.57E-03	1.35E-05	3.83E-03	3.34E-05
Deviation	2.43E-03	4.62E-05	8.57E-04	1.66E-05	1.99E-03	2.63E-05



U-Velocity Test Case-2

Temperature Test Case-2

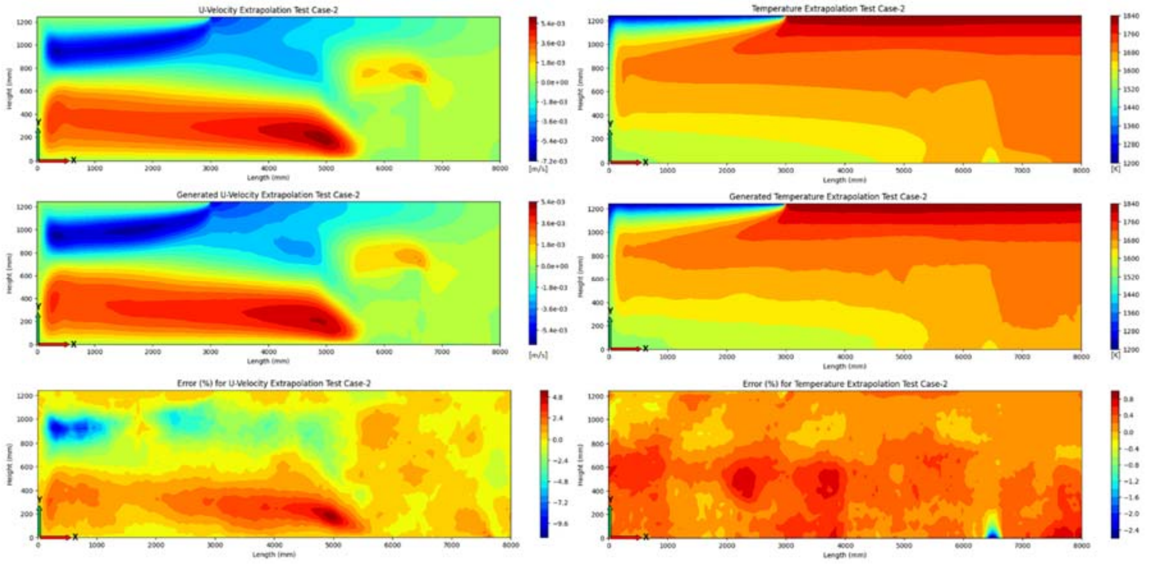
Generated U-Velocity Test Case-2

Generated Temperature Test Case-2

Error (%) for U-Velocity Test Case-2

Error (%) for Temperature Test Case-2

Figure 8. U-velocity and temperature contours for the interpolation Case #2 (top: CFD, middle: ROM, bottom: relative error)



U-Velocity Extrapolation Test Case-2

Temperature Extrapolation Test Case-2

Generated U-Velocity Extrapolation Test Case-2

Generated Temperature Extrapolation Test Case-2

Error (%) for U-Velocity Extrapolation Test Case-2

Error (%) for Temperature Extrapolation Test Case-2

Figure 9. U-velocity and temperature contours for the extrapolation Case #2 (top: CFD, middle: ROM, bottom: relative error)

Using ROM for the Operational Analysis

The positive outcomes obtained from the interpolation and extrapolation test cases open the way for the effective use of the ROM in CFD-assisted operational analysis. Field solutions for velocity and temperature provide numerous opportunities to investigate the condition of the furnace and gain insights.

Similar to parametric CFD studies, a ROM, generated using parametric CFD solutions, can be utilized during furnace operations to address the physical phenomena encountered. The temperature predictions by the ROM, as shown in Figure 10, serve as an example to demonstrate this capability.

Thermocouples (T/C) positioned at the bottom of the tank are used to monitor temperature of the melt. In Figure 10, we see temperature predictions for a possible thermocouple location, which is at the bottom and where x equals to 7 m. The developed ROM can be used to predict temperature at that location and map the change of temperature within the specified operating range. Both an increase in the potential difference between the electrodes (x-axis) and an increase in the heat flux from the combustion space (y-axis) result in a higher temperature at the T/C location. However, the influence of the potential difference on the temperature is greater than that of the heat flux.

This kind of process requires several CFD solutions to obtain a result with satisfactory resolution. The developed ROM enables each CFD prediction to be made in under a second, thereby providing an opportunity for real-time support during furnace operation.

The results of another investigation conducted for the furnace operation using the developed ROM are presented in Figure 11 and Figure 12. In this scenario, the effects of an increase in power input from the combustion space and the electrodes on the depth-wise velocity and temperature profiles at the middle of the melting tank are examined.

Sequentially, the heat flux (denoted as heat flux+), the potential difference (denoted as potential+) and finally both of them are increased by 5%. The resulting profiles are plotted against the base-case to observe the influence of each power input and their combined effect. The resulting profiles are derived from the field solutions predicted by the ROM using the corresponding boundary conditions.

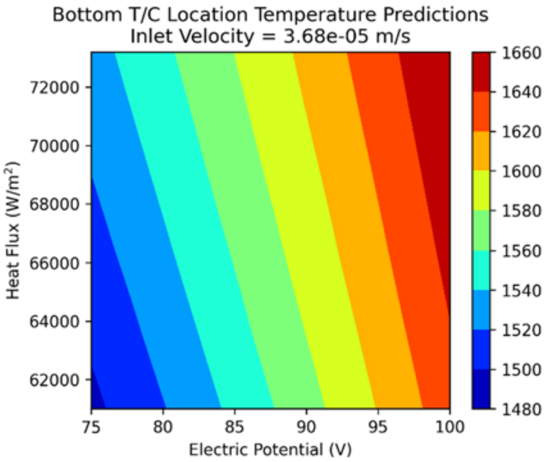


Figure 10. Continuous temperature predictions, generated by the ROM at the T/C location varying with changes in heat flux and potential difference. This demonstrates the on-demand information extraction capability within the operating range.

In Figure 11, a significant increase in the magnitude of backward and forward flow components is observed in the cases of increased potential difference, compared to a sole boost in heat flux.

Similar observation are made in Figure 12, which represents the depth-wise temperature profiles. The temperature difference is more apparent at the bottom section compared to the section near the glass surface. The overall increase in temperature leads to lower viscosity, which can be regarded as the root cause of the faster glass velocities.

These operational analysis examples demonstrate the post-processing capability of the machine learning-based ROMs. Thanks to the unprecedented interpolation feature of the ROMs, it is possible to extract more information than with conventional CFD post-processing.

CONCLUSION

This study presents a new method for integrating Computational Fluid Dynamics (CFD) models into the routine operations of industrial glass melting furnaces. The research achieves this through an autoencoder-based ROM with CNNs trained on parametric CFD data. A 2D CFD model is developed as a foundation for a machine-learning-based reduced-order model (ROM), demonstrating the potential of machine learning (ML) techniques to create efficient ROMs. Although glass melting in a furnace is inherently a 3D process, this study simplifies the analysis by using a 2D approach. Treating the 3D process as a series of 2D snapshots, this methodology can be extended to a 3D study using similar sampling techniques and convolutional neural networks, though a 3D study would involve a larger dataset and more network parameters to train. The developed generator network, i.e., the ROM, accurately predicts temperature and velocity fields, achieving MAE under 1% for validation, interpolation, and extrapolation tests compared to the ground-truth CFD solutions. Mean squared error is 1.94E-06 for training, 4.45E-06 for validation, 2.82E-06 for interpolation, and 2.79E-05 for extrapolation cases. In addition, R^2 score of 0.99 is achieved during training, validation and tests for the generator network. These results indicate ROM's potential as an accurate analysis tool for daily furnace operation, providing insights into key operational parameters. Furthermore, the ROM demonstrates significant improvements in solution time, enabling almost instantaneous solutions for temperature and velocity fields, which enhances its practical usefulness in real-time decision-making. Its robustness has been presented through interpolation and extrapolation tests, demonstrating consistent and reliable results across various operational scenarios.

This application of ML-based ROMs in glass furnace operations represents a substantial step forward in efficiency and optimization. However, a ROM based on CFD data is inherently constrained to the range of parameters it was trained on. Expanding the ROM to accommodate new parameters requires additional CFD data generation and model retraining. Additionally, while the ROM performs well within its trained parameter space, extrapolation beyond this range can lead to decreased accuracy, as demonstrated in this study. This limitation underscores the importance of careful ROM training to match the intended operational range. Overall, the integration of ML-based ROMs within industrial glass furnace operations showcases significant potential for broader applications in computational modeling and decision-support systems in various industrial contexts.

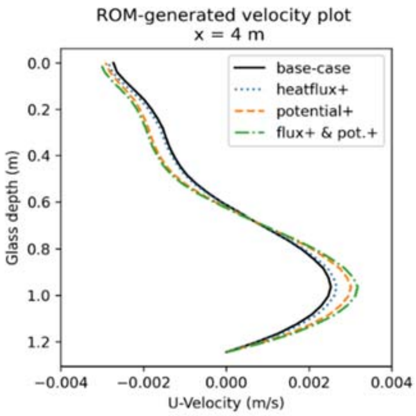


Figure 11. ROM-generated depth-wise u-velocity profiles at the middle of the melting tank. Velocity profiles change with the varying combustion space and electrical power input.

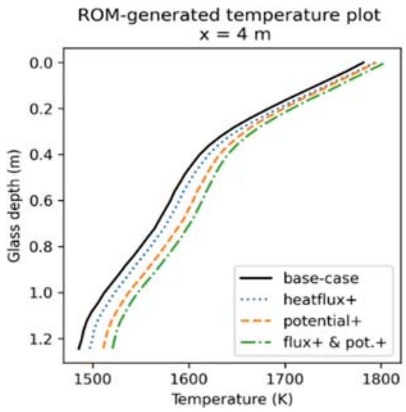


Figure 12. ROM-generated depth-wise temperature profiles at the middle of the melting tank, showing the effect of the varying energy input.

REFERENCES

Abadi, M., Agarwal, A., Barham, P., Brevdo, E., Chen, Z., Citro, C., Corrado, G. S., Davis, A., Dean, J., Devin, M., Ghemawat, S., Goodfellow, I., Harp, A., Irving, G., Isard, M., Jia, Y., Jozefowicz, R., Kaiser, L., Kudlur, M., … Zheng, X. (2015). *TensorFlow: Large-Scale Machine Learning on Heterogeneous Systems, Version 1.15*. https://www.tensorflow.org/

Abbassi, A., & Khoshmanesh, K. (2008). Numerical simulation and experimental analysis of an industrial glass melting furnace. *Applied Thermal Engineering*, 28(5–6), 450–459. https://doi.org/10.1016/j.applthermaleng.2007.05.011

Abooali, D., & Khamehchi, E. (2019). New predictive method for estimation of natural gas hydrate formation temperature using genetic programming. *Neural Computing and Applications*, 31(7), 2485–2494. https://doi.org/10.1007/s00521-017-3208-0

ANSYS Inc. (2022). *Ansys Fluent User's Guide*. https://www.ansys.com

Atzori, D., Tiozzo, S., Vellini, M., Gambini, M., & Mazzoni, S. (2023). Industrial Technologies for CO2 Reduction Applicable to Glass Furnaces. *Thermo*, 3(4), 682–710. https://doi.org/10.3390/thermo3040039

Bhatnagar, S., Afshar, Y., Pan, S., Duraisamy, K., & Kaushik, S. (2019). Prediction of aerodynamic flow fields using convolutional neural networks. *Computational Mechanics*, 64(2), 525–545. https://doi.org/10.1007/s00466-019-01740-0

Brunton, S. L., Noack, B. R., & Koumoutsakos, P. (2020). Machine learning for fluid mechanics. *Annual Review of Fluid Mechanics*, 52(1), 477–508. https://doi.org/10.48550/arXiv.1905.11075

Cassar, D. R., de Carvalho, A. C. P. L. F., & Zanotto, E. D. (2018). Predicting glass transition temperatures using neural networks. *Acta Materialia*, 159, 249–256. https://doi.org/10.1016/j.actamat.2018.08.022

Castillo, V., & Kornish, B. (2017). *Development of Reduced Glass Furnace Model to Optimize Process Operation*, Final Report CRADA No. TC02241.

Catsoulis, S., Singh, J.-S., Narayanan, C., & Lakehal, D. (2022). Integrating supervised learning and applied computational multi-fluid dynamics. *International Journal of Multiphase Flow*, 157, 104221. https://doi.org/10.1016/j.ijmultiphaseflow.2022.104221

Celik, I. B., Ghia, U., Roache, P. J., Freitas, C. J., Coleman, H., & Raad, P. E. (2008). Procedure for estimation and reporting of uncertainty due to discretization in CFD applications. *Journal of Fluids Engineering, Transactions of the ASME*, 130(7), 0780011–0780014. https://doi.org/10.1115/1.2960953

Cho, G., Wang, M., Kim, Y., Kwon, J., & Su, W. (2022). A physics-informed machine learning approach for estimating lithium-ion battery temperature. *IEEE Access*, 10, 88117–88126. https://doi.org/10.1109/ACCESS.2022.3199652

Chollet, F. (2021). *Deep learning with Python*. Simon and Schuster.

Choudhary, M. (1985). Three-dimensional Mathematical Model for Flow and Heat Transfer in Electric Glass Furnaces. *Heat Transfer Engineering*, 6(4), 55–65. https://doi.org/10.1080/01457638508939639

Choudhary, M. K., & Potter, R. M. (2005). Heat transfer in glass-forming melts. In L. D. Pye, A. Montenero, & I. Joseph (Eds.), *Properties of glass-forming melts* (pp. 249–294). Taylor & Francis Publications. https://doi.org/10.1201/9781420027310

Choudhary, M. K., Purnode, B., Lankhorst, A. M., & Habraken, A. F. J. A. (2018). Radiative heat transfer in processing of glass-forming melts. *International Journal of Applied Glass Science*, 9(2), 218–234. https://doi.org/10.1111/ijag.12286

Choudhary, M. K., Venuturumilli, R., & Hyre, M. R. (2010). Mathematical modeling of flow and heat transfer phenomena in glass melting, delivery, and forming processes. *International Journal of Applied Glass Science*, 1(2), 188–214. https://doi.org/10.1111/j.2041-1294.2010.00011.x

Cravero, C., & Marsano, D. (2023). Numerical Simulation of Melted Glass Flow Structures inside a Glass Furnace with Different Heat Release Profiles from Combustion. *Energies*, 16(10), 4187. https://doi.org/10.3390/en16104187

Daurer, G., Raič, J., Demuth, M., Gaber, C., & Hochenauer, C. (2022). Comprehensive and numerically efficient CFD model for bubbling in an industrial glass tank. *Chemical Engineering Research and Design*, 186, 82–96. https://doi.org/10.1016/j.cherd.2022.07.044

Erichson, N. B., Mathelin, L., Yao, Z., Brunton, S. L., Mahoney, M. W., & Kutz, J. N. (2019). Shallow learning for fluid flow reconstruction with limited sensors and limited data. *ArXiv Preprint* ArXiv:1902.07358. https://doi.org/10.1098/rspa.2020.0097

Faber, A.-J., Rongen, M., Lankhorst, A., & Meneses, D. D. S. (2020). Characterization of high temperature optical spectra of glass melts and modeling of thermal radiation conductivity. *International Journal of Applied Glass Science*, 11(3), 442–462. https://doi.org/10.1111/ijag.15116

Gao, H., Sun, L., & Wang, J.-X. (2021). PhyGeoNet: Physics-informed geometry-adaptive convolutional neural networks for solving parameterized steady-state PDEs on irregular domain. *Journal of Computational Physics*, 428, 110079. https://doi.org/10.1016/j.jcp.2020.110079

Han, J., Li, L., Wang, J., Chen, S., Liu, C., & Li, C. (2022). Simulation and evaluation of float glass furnace with different electrode positions. *Journal of the American Ceramic Society*, 105(12), 7097–7110. https://doi.org/10.1111/jace.18700

He, X.-J., Yu, C.-H., Zhao, Q., Peng, J.-Z., Chen, Z.-H., & Hua, Y. (2022). Reduced order modelling of natural convection of nanofluids in horizontal annular pipes based on deep learning. *International Communications in Heat and Mass Transfer*, 138, 106361. https://doi.org/10.1016/j.icheatmasstransfer.2022.106361

Jebava, M., & Němec, L. (2018). Role of glass melt flow in container furnace examined by mathematical modelling. *Ceram-Silik*, 62, 86–96. https://doi.org/10.13168/cs.2017.0049

Kim, B., Azevedo, V. C., Thuerey, N., Kim, T., Gross, M., & Solenthaler, B. (2019). Deep fluids: A generative network for parameterized fluid simulations. *Computer Graphics Forum*, 38(2), 59–70. https://doi.org/10.1111/cgf.13619

Lankhorst, A. M., Thielen, L., Simons, P. J. P. M., & Habraken, A. F. J. A. (2013). Proper modeling of radiative heat transfers in clear glass melts. *73rd Conference on Glass Problems*, 249–258. https://doi.org/10.1002/9781118710838.ch19

Li, L., Han, J., Lin, H.-J., Ruan, J., Wang, J., & Zhao, X. (2020). Simulation of glass furnace with increased production by increasing fuel supply and introducing electric boosting. *International Journal of Applied Glass Science*, 11(1), 170–184. https://doi.org/10.1111/ijag.13907

Masoumi-Verki, S., Haghighat, F., & Eicker, U. (2022). A review of advances towards efficient reduced-order models (ROM) for predicting urban airflow and pollutant dispersion. *Building and Environment*, 216, 108966. https://doi.org/10.1016/j.buildenv.2022.108966

Matsuno, S., Iso, Y., Uchida, H., Oono, I., Fukui, T., & Ooba, T. (2008). CFD modeling coupled with electric field analysis for joule-heated glass melters. *Journal of Power and Energy Systems*, 2(1), 447–455. https://doi.org/10.1299/jpes.2.447

Molinaro, R., Singh, J.-S., Catsoulis, S., Narayanan, C., & Lakehal, D. (2021). Embedding data analytics and CFD into the digital twin concept. *Computers & Fluids*, 214, 104759. https://doi.org/10.1016/j.compfluid.2020.104759

Mücke, N. T., Bohté, S. M., & Oosterlee, C. W. (2021). Reduced order modeling for parameterized time-dependent PDEs using spatially and memory aware deep learning. *Journal of Computational Science*, 53, 101408. https://doi.org/10.1016/j.jocs.2021.101408

Paszke, A., Gross, S., Chintala, S., Chanan, G., Yang, E., DeVito, Z., Lin, Z., Desmaison, A., Antiga, L., & Lerer, A. (2017). *Automatic differentiation in pytorch*.

Pazarlioglu, H. K., Tepe, A. Ü., & Arslan, K. (2022). Optimization of Parameters Affecting Anti-Icing Performance on Wing Leading Edge of Aircraft. *European Journal of Science and Technology*. https://doi.org/10.31590/ejosat.1062495

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., & Duchesnay, É. (2011). Scikit-learn: Machine Learning in Python. *Journal of Machine Learning Research*, 12(85), 2825–2830. http://jmlr.org/papers/v12/pedregosa11a.html

Pfaff, T., Fortunato, M., Sanchez-Gonzalez, A., & Battaglia, P. W. (2020). Learning mesh-based simulation with graph networks. *ArXiv Preprint* ArXiv:2010.03409.

Pigeonneau, F., & Flesselles, J.-M. (2012). Practical laws for natural convection of viscous fluids heated from above in a shallow cavity. *International Journal of Heat and Mass Transfer*, 55(1–3), 436–442. https://doi.org/10.1016/j.ijheatmasstransfer.2011.09.057

Pigeonneau, F., Pereira, L., & Laplace, A. (2023). Dynamics of rising bubble population undergoing mass transfer and coalescence in highly viscous liquid. *Chemical Engineering Journal*, 455, 140920.

Pokorny, R., Hrma, P., Lee, S., Klouzek, J., Choudhary, M. K., & Kruger, A. A. (2020). Modeling batch melting: Roles of heat transfer and reaction kinetics. *Journal of the American Ceramic Society*, 103(2), 701–718. https://doi.org/10.1111/jace.16820

Pokorny, R., Pierce, D. A., & Hrma, P. (2012). Melting of glass batch: model for multiple overlapping gas-evolving reactions. *Thermochimica Acta*, 541, 8–14. https://doi.org/10.1016/j.tca.2012.04.020

Raič, J., Gaber, C., Wachter, P., Demuth, M., Gerhardter, H., Knoll, M., Prieler, R., & Hochenauer, C. (2021). Validation of a coupled 3D CFD simulation model for an oxy-fuel cross-fired glass melting furnace with electric boosting. *Applied Thermal Engineering*, 195, 117166. https://doi.org/10.1016/j.applthermaleng.2021.117166

Rowley, C. W., & Dawson, S. T. M. (2017). Model reduction for flow analysis and control. *Annual Review of Fluid Mechanics*, 49(1), 387–417. https://doi.org/10.1146/annurev-fluid-010816-060042

Schill, P., & Chmelar, J. (2004). Use of computer flow dynamics in glass technology. *Journal of Non-Crystalline Solids*, 345, 771–776. https://doi.org/10.1016/j.jnoncrysol.2004.08.199

Simcik, M., & Ruzicka, M. C. (2015). CFD model for pneumatic mixing with bubble chains: Application to glass melts. *Chemical Engineering Science*, 127, 344–361. https://doi.org/10.1016/j.ces.2015.01.052

Simons, P., Jochem, K., & Aiuchi, K. (2008). A power consistent mathematical formulation for Joulean heat release. *Glass Technology-European Journal of Glass Science and Technology Part A*, 49(3), 109–118.

Soubeih, S., Luedtke, U., & Halbedel, B. (2015). Improving residence time distribution in glass melting tanks using additionally generated Lorentz forces. *J. Chem. Chem. Eng*, 9, 203–210. https://doi.org/10.17265/1934-7375/2015.03.006

Staněk, J. (1990). Problems in electric melting of glass. *Journal of Non-Crystalline Solids*, 123(1), 400–414. https://doi.org/10.1016/0022-3093(90)90812-Z

Taşkesen, E., Dirik, M., Tekir, M., & Pazarlioğlu, H. K. (2023). Predicting heat transfer performance of Fe3O4-Cu/water hybrid nanofluid under constant magnetic field using ANN. *Journal of Thermal Engineering*, 9(3), 811–822. https://doi.org/10.18186/thermal.000000

Thuerey, N., Weißenow, K., Prantl, L., & Hu, X. (2020). Deep learning methods for Reynolds-averaged Navier–Stokes simulations of airfoil flows. *AIAA Journal*, 58(1), 25–36. https://doi.org/10.2514/1.J058291

Viskanta, R. (1994). Review of three-dimensional mathematical modeling of glass melting. *Journal of Non-Crystalline Solids* (Vol. 177). https://doi.org/10.1016/0022-3093(94)90549-5

Wiewel, S., Becher, M., & Thuerey, N. (2019). Latent space physics: Towards learning the temporal evolution of fluid flow. *Computer Graphics Forum*, 38(2), 71–82. https://doi.org/10.48550/arXiv.1802.10123

Zhang, R., Liu, Y., & Sun, H. (2020). Physics-guided convolutional neural network (PhyCNN) for data-driven seismic response modeling. *Engineering Structures*, 215, 110704. https://doi.org/10.1016/j.engstruct.2020.110704

Zier, M., Stenzel, P., Kotzur, L., & Stolten, D. (2021). A review of decarbonization options for the glass industry. *Energy Conversion and Management: X*, 10(February), 100083. https://doi.org/10.1016/j.ecmx.2021.100083