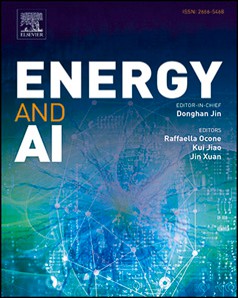
[Energy and AI xxx (xxxx) xxx](https://doi.org/10.1016/j.egyai.2023.100310)



Contents lists available at [ScienceDirect](https://www.elsevier.com/locate/egyai)

Energy and AI

journal homepage: [www.elsevier.com/locate/egyai](http://www.elsevier.com/locate/egyai)

Effective thermal conductivity estimation using a convolutional neural network and its application in topology optimization

Andre Adam [a](#_bookmark0), Huazhen Fang [a](#_bookmark0), Xianglin Li [a](#_bookmark0),[b](#_bookmark1),[∗](#_bookmark2)

a *Department of Mechanical Engineering, University of Kansas, 1530 W 15th St., Lawrence, 66045, KS, USA*

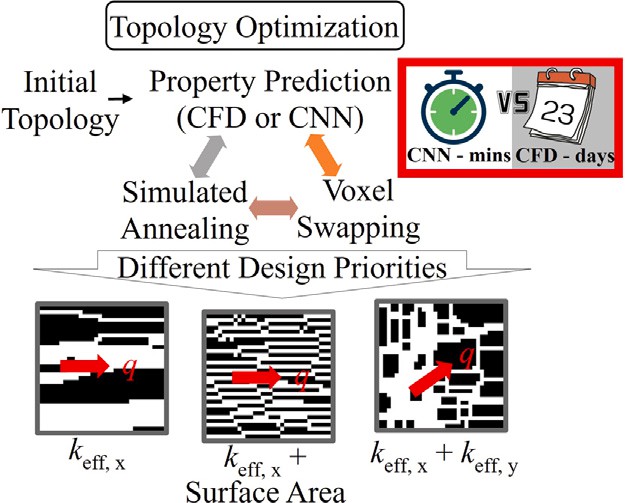
b *Department of Mechanical Engineering and Materials Science, Washington University in St. Louis, One Brookings Drive, St. Louis, 63130, MO, USA*

H I G H L I G H T S

* Dataset with 130,000 images and 520,000 effective thermal conductivity simula- tions.
* CNN has MAPEs between 0.35% and 2.35% for ratios of thermal conductivity between 10 and 341.7.
* CNN predicts effective thermal conduc- tivity in 15 ms on commercial GPU.
* Prediction speed is 20 ms on CPU for a single image.
* The CNN model is successful in 4 cases of topology optimization.

A R T I C L E I N F O

G R A P H I C A L A B S T R A C T



A B S T R A C T

*Keywords:*

Machine learning

Effective thermal conductivity Convolutional neural network VGG

Numerical heat transfer Topology optimization

Topology optimization of heterogeneous structures can find significant use in a wide range of applications, and its fabrication has been made possible by recent advances in additive manufacturing. However, the optimization procedure is computationally expensive, as each structural update requires the re-evaluation of the properties. The computational time is the major limiting factor in large-scale and complex structural optimization. In this study, a convolutional neural network (CNN) model for predicting effective thermal conductivity inspired by the VGG networks is proposed. Trained using 130,000 unique binary images, the model achieves high predictive accuracy. Specifically, it shows a mean absolute percent error (MAPE) of 0.35% in testing when the thermal conductivity of the solid is ten times larger than the fluid, and when the thermal conductivities assigned are that of aluminum and water, the MAPE is 2.35%. The prediction time is 15 ms

for a single image with 128 × 128 pixels, which is 3 to 5 orders of magnitude faster than a finite volume

simulation. When employed in topology optimization, the CNN retains a MAPE between 0.67% and 11.8% for different cases. The CNN model correctly predicts trends in effective thermal conductivity and improves the structure to close proximity of a theoretical maximum in all cases.

∗ Corresponding author at: Department of Mechanical Engineering and Materials Science, Washington University in St. Louis, One Brookings Drive, St. Louis, 63130, MO, USA.

*E-mail address:* [lxianglin@wustl.edu](mailto:lxianglin@wustl.edu) (X. Li).

<https://doi.org/10.1016/j.egyai.2023.100310>

Received 25 July 2023; Received in revised form 9 September 2023; Accepted 14 October 2023

Available online 19 October 2023

2666-5468/© 2023 The Author(s). Published by Elsevier Ltd. This is an open access article under the CC BY-NC-ND license ([http://creativecommons.org/licenses/by-](http://creativecommons.org/licenses/by-nc-nd/4.0/) [nc-nd/4.0/](http://creativecommons.org/licenses/by-nc-nd/4.0/)).

Please cite this article as: Andre Adam et al., *Energy and AI*, https://doi.org/10.1016/j.egyai.2023.100310

# Introduction

Heterogeneous materials are an intrinsic aspect of many fields of study in science and engineering for their incredible inherent me- chanical and transport properties [[1](#_bookmark71)]. In energy research, transport properties related to the geometry, orientation, and topology of the material are often intrinsic to overall system performance. Studies have shown several physical properties, such as effective diffusivity [[2](#_bookmark72),[3](#_bookmark73)], permeability [[2](#_bookmark72),[4](#_bookmark50),[5](#_bookmark51)], or even structural properties [[6](#_bookmark52)], have integral dependence on the distribution of the phases involved.

One such property is the effective thermal conductivity [[2](#_bookmark72),[3](#_bookmark73)], which refers to thermal conductivity when the convective and radiative heat transfer are negligible and at least two phases with different conducive properties are involved. This property has a wide range of applica- tions, especially in porous media that impede general fluid flow and do not have significant temperature gradients across the structure, and is relevant across the fields of petroleum engineering, geology, electrochemistry, and food science, among others.

A multitude of analytical and empirical correlations have been made to predict the effective thermal conductivity of a system [[7](#_bookmark53),[8](#_bookmark54)], but such models often assume simplified geometries, thus having very limited ranges of validity covering small subsets of topology and porosity. The alternative is experimenting or simulating to predict the property, both of which have significant physical time and cost associated. An alternative method for obtaining both effective thermal conductiv- ity and effective diffusivity efficiently is based on the random-walk theory [[8](#_bookmark54),[9](#_bookmark55)]. These studies highlight the speed advantage of random- walk-based models over conventional numercial heat transfer (NHT) models, while providing higher accuracy than empirical models. This type of model has the added benefit of providing fast computations for either 2D or 3D domains.

Recent progress in machine learning (ML) and computer vision have allowed to build data-driven predictive models that provide compu- tational understanding of topological features. A convolutional neural network (CNN) is a particular type of neural network that was orig- inally introduced as the ‘‘*Neocognitron*’’ [[10](#_bookmark56)], a computational model inspired by Hubel and Wiesel’s hierarchical receptive field model of the visual cortex. Since then, a number of improvements have been made to the topic of image segmentation [[11](#_bookmark57)]. Particularly suitable for the implementation of CNNs is graphic processing units (GPUs), which have seen fast growth in their computing power and scalability in recent years.

Two recent studies [[12](#_bookmark58),[13](#_bookmark59)] have successfully implemented CNN- based models for structure–property integration for materials science. These studies utilized the ability of convolutional layers to extract key features from a structure and feed them to fully connected layers to predict material properties. Through training, the model learns to extract crucial topological features and make an educated guess at the relevant physical feature from the extracted information.

In energy research, Wu et al. [[14](#_bookmark60)] considered CNN architectures based on AlexNet [[15](#_bookmark61)] and ResNet [[16](#_bookmark62)] to predict the effective dif- fusivity of porous media solely from images, with a training dataset of 1960 samples. The proposed model performs more accurately than empirical. However, the accuracy is still limited to a certain extent, possibly because of the relatively few images to train the CNN. The prediction time is not discussed, and there is no validation of the CNN model with data outside the dataset.

Wei et al. [[17](#_bookmark63)] used a CNN as well as other ML methods to predict the effective thermal conductivity of composite materials and porous media. With an original CNN architecture, using a maximum dataset size of 1400 samples, an accuracy in terms of correlation coefficient (R2) of 0.986 or a relative mean squared error (RMSE) of 1.9% is re- ported, which is often more accurate than popular analytic models such

display a R2 value of 0.93 and a mean absolute error of 5.2 W∕(m K) in

testing. Other studies also compare different ML methods for predicting and optimizing thermal transport models [[19](#_bookmark65)], highlighting the effec- tiveness of such models in property prediction and their use in material optimization.

While the aforementioned studies [[14](#_bookmark60),[17](#_bookmark63),[18](#_bookmark64)] use CNNs for predict- ing properties relevant to energy research, the accuracy and efficiency need to be further refined in order to be employed in practical situa- tions, as there are no discussions on prediction time, CNN optimization, and general predictions outside of the dataset. This study uses recent advancements in training CNNs and parameter optimization to pro- pose a CNN-based model to predict effective thermal conductivity. We thoroughly discuss how the accuracy of the NHT model, CNN archi- tecture, hyperparameter selection, dataset size, and training schedule all collectively impact the accuracy of the CNN in predicting effective thermal conductivity, while also taking in consideration the prediction speed. Finally, this CNN model is validated with structures that possess analytical results and data outside the dataset, to make sure the CNN model can retain accuracy in prediction even if the input is significantly different from the training dataset. Our model is optimized for speed, accuracy, and generalization across multiple types of 2D structures, by drawing inspiration from the VGG-class [[20](#_bookmark66)] networks. The CNN-based models can make predictions from newly entered data in milliseconds. In contrast, a prediction of a 3D model using the random-walk approach takes seconds [[8](#_bookmark54),[9](#_bookmark55)], and a 3D NHT model takes minutes.

We created a dataset with more than 130,000 images and their thermal conductivities simulated by central-differencing based NHT model. The NHT model is parallelized and all NHT simulations were performed with the resources from the Bridges2 supercomputer from the Pittsburgh Supercomputing Center. The size of the dataset can be hard to manage on a conventional computer, so the training and testing of the proposed CNN models and architectures are all done with resources from the GPU partition of the Expanse supercomputer at the San Diego Supercomputer Center.

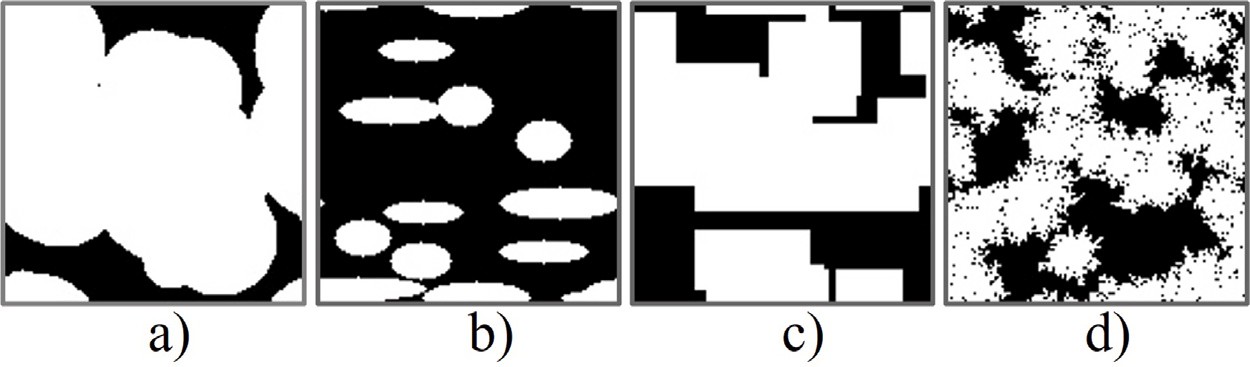
Another objective of the study is to assess whether or not the CNN predictive model can be applied to iterative topology optimization problems. Topology optimization is a process of changing structures to maximize or minimize a specific property [[21](#_bookmark67)]. The main challenge in iterative topology optimization is that the optimized properties of the structure must be reassessed at every structural update. Thus the computational cost often limits the size of the domain as well as the complexity of the model [[22](#_bookmark68),[23](#_bookmark69)]. The CNN-based predictive model could reduce the computational time by orders of magnitude, enabling much larger domains as well as complex cost functions. The prediction accuracy of material properties at each step is critical to the success of topology optimization. Despite some existing studies on ML for material optimization [[24](#_bookmark70)], a CNN model that is accurate enough for iterative topology optimization has yet to be successfully demonstrated, to the best of the authors knowledge.

The following section will outline the dataset generation. Section [3](#_bookmark24) then presents the choice of architecture and hyperparameter optimiza- tion, which is an adaptation from the VGG [[20](#_bookmark66)] family of CNNs. Section [4](#_bookmark28) investigates the accuracy of the model in training and testing, potential sources of error, prediction speed in different environments, and validations of the proposed model using test cases from analytical solutions. Section [5](#_bookmark32) applies the trained CNN models to test four cases in topology optimization to demonstrate a practical application of the model. Section [6](#_bookmark41) concludes the work. The trained network, the training dataset [[25](#_bookmark74)], and a sample code showing how to load and use the models are available to download.[1](#_bookmark3)

as Maxwell-Eucken model or the Bruggeman model. Zhu et al. [[18](#_bookmark64)]

proposes another CNN based model with original architecture, and for 300 samples on a 90/10 split between train/test dataset, the authors

1 GitHub to trained networks and sample code: [https://github.com/adama-](https://github.com/adama-wzr/CNN-Keff) [wzr/CNN-Keff](https://github.com/adama-wzr/CNN-Keff)



packing. **(d)** shows a sample image generated with the QSGS algorithm. All images have 128 × 128 pixels. **/ig. 1.** One example from each generated dataset. **(a)** shows a structure of randomly packed circles. **(b)** is a sample of an ellipse-packed domain. **(c)** is an example of rectangle

# Training dataset generation

* 1. *Image generation*

A key component of this study is the generation of images for the training dataset, as well as simulating their effective thermal conductiv- ity. Ideally, these images would encompass as many different types of heterogeneous structures as possible, such that the CNN model applies to a broad range of structures after training. Sharing the images in an open-access format encourages other researchers to contribute to this dataset, making it better generalized over time, and exposing the CNN in training to as many features as possible.

While there are several methods [[26](#_bookmark75),[27](#_bookmark76)] to digitally reconstruct structures, most methods are constrained in speed and/or require dig- ital image scans, analytical correlations, or detailed structure informa- tion. Since the goal is to compile the most general possible dataset, a stochastic approach is taken by packing a domain with regular structures of arbitrary sizes. Past research [[28](#_bookmark77)–[30](#_bookmark79)] has established some resemblance between randomly packed regular shapes with actual structures.

the rate of growth for the particles in each iteration is set to 0.0005 in the diagonals and 0.001 on the sides. The QSGS dataset is highly stochastic, so the 10,000 QSGS are not modified by rotating or flipping

each with the size of 128 × 128 pixels, to train the CNN. the color scheme. Our dataset includes 130,000 total binary images,

* 1. *NHT model*

The efficiency and accuracy of the NHT model are paramount to the CNN dataset creation and training. Wu et al. [[14](#_bookmark60)] used 1,960 pairs of structures and effective diffusivity, Wei et al. [[18](#_bookmark64)] had 300 samples in the training dataset for effective thermal conductivity prediction. Those studies used the lattice Boltzmann method (LBM) and cited issues with computational time as a limiting factor. Hence, here we propose and validate a simple NHT model with well-established ranges of validity and high efficiency. This study focuses on 2D heat conduction problems, thus, the goal is to solve the 2D steady-state heat equation, shown in Eq. ([1](#_bookmark6)).

Four datasets are created for this study: random quadrilateral pack-

*𝜕* (

*𝜕𝑇* )

*𝜕* (

*𝜕𝑇* )

ing, random sphere packing, random ellipse packing, and quartet struc- ture generation set (QSGS) [[31](#_bookmark80)], containing 10,000 images each. The

*𝜕𝑥*

*𝑘*

*𝜕𝑥*

+

*𝜕𝑦*

*𝑘*

*𝜕𝑦*

= 0*,* (1)

random parameters of each dataset were established to provide a near- uniform distribution in effective thermal conductivity between the theoretical minimum and maximum. The dataset extensively covers the range of porosities and increases the odds of covering the full range of thermal conductivities. [Fig.](#_bookmark4) [1](#_bookmark4) displays a sample image from each dataset. [Appendix](#_bookmark44) [A](#_bookmark44) contains further details, such as porosity histograms and porosity versus effective thermal conductivity data.

For all the randomly-packed regular shapes, the number of objects is a uniform distribution between 1 and 30, the center of each individual object is also random, and the objects are allowed to overlap. In the random quadrilateral packing, the parameters controlling the images are the height and width, which are uniform distributions between 8 to

40 and 8 to 60 pixels, respectively. For the random circle packing, the

where *𝑘* is the local thermal conductivity.

The second-order partial differential equation is discretized using a central differencing scheme. This model uses each pixel of the image as

a control volume, so for a 128 × 128 image, the computational domain

includes 128 × 128 cells. The temperatures are calculated at the center

of the cells by the finite volume method. These approximations are verified using well-known heat transfer cases in [Table](#_bookmark9) [1](#_bookmark9) by gathering results from the NHT model and comparing them with the analytic solutions.

Eq. ([1](#_bookmark6)) can be generalized by defining non-dimensional parameters.

A non-dimensional temperature, *𝜃*, is defined in Eq. ([2](#_bookmark7)) below.

*𝑇* − *𝑇*cold

radii follow an uniform distribution between 4 and 40 pixels. Finally,

*𝜃* =

*𝑇*

hot

– *𝑇*

cold

*,* (2)

for the ellipse-packed structures, the semi-major axis varied between

10 and 30 pixels, and the semi-minor axis varied between 4 and 10

where *𝑇𝑐𝑜𝑙𝑑*

and *𝑇*

*ℎ𝑜𝑡*

are the temperatures of the cold and hot sides of

pixels, also follow uniform distributions. The 30,000 images generated this way represent simple yet realistic approximations of structures. The dataset can be further extended from the original images by rotating each by 90◦, flipping the color scheme, and doing both, yielding

the domain. Dimensionless length scales can be defined as in Eq. ([3](#_bookmark8)), as functions of the height of the domain, *H*, and the width of the domain, *W*.

*𝜉* = *𝑥*

120,000 unique images. The 10,000 QSGS [[31](#_bookmark80)] images were created in two batches of 5000 with different parameters. For each subset of

*𝑊*

*𝜂* = *𝑦*

*𝐻*

(3)

5,000 images, the target solid volume fraction (*𝑃𝑠*) varies between 0.1

and 0.9 uniformly, such that it starts at 0.1 and increases by 0.04 every

*𝑘̄*

is the non-dimensionalized thermal conductivity calculated by

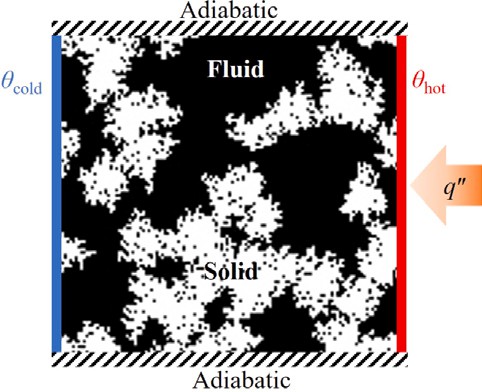
250 images. For the first subset, the seed probability, *𝐶𝑑* , was set at 0.0025 for each voxel, which on a 1282 domain yields approximately 40

the diagonals of a solid pixel, *𝐷𝑑𝑖𝑎𝑔* , is set to 0.0001, and the probability particle seeds. The growth probability of each particle per iteration in of growth on sides, *𝐷𝑠𝑖𝑑𝑒𝑠*, is set to 0.0004. For the second subset of images, *𝐶𝑑* is set to 0.00125, yielding 20 particle seeds per domain, and

dividing the thermal conductivity of solid with the thermal conductivity of the fluid, as shown in Eq. ([4](#_bookmark12)). With this approach, the dimensionless

effective thermal conductivity, *𝑘̄* eff, is a measure of how much the

possible value of *𝑘̄* eff is 1, which represents a domain filled with fluid. solid contributes to the thermal conductivity of the system. The lowest Likewise, the maximum possible value is *𝑘̄* eff = *𝑘̄* s, for a domain with

**Table 1**

Statistics showing the effect of increasing the *𝑘̄ 𝑠* on the performance and convergence

of the NHT simulations.

*𝑘̄*

s

r

4*.*22 ⋅ 10−8

l

Avg. iterations Avg. *𝑞*′′ + *𝑞*′′

Avg. Time (s)

|  |  |  |  |
| --- | --- | --- | --- |
| 10  50 | 32,903  72,350 | | |  1*.*91 ⋅ 10−7 | 34.2  81.6 |
| 100 | 105,096 | 1*.*00 ⋅ 10−6 | 137.3 |
| 341.7 | 154,307 | 9*.*75 ⋅ 10−5 | 371.2 |

**/ig. 2.** Simulation domain with boundary conditions.

the same from the simulation setup, it is convenient to define *𝑞*′′ as

in Eq. ([9](#_bookmark11)).

avg

*𝑞*′′ + *𝑞*′′

no fluid phase.

′′

avg

= r l

2

*𝑞*

(9)

*𝑘̄* s

= *𝑘*s

*𝑘*f

The dimensionless effective thermal conductivity can be calculated from Eq. ([10](#_bookmark13)).

*̄ 𝑘*f

*𝑞*′′ *𝜉*

*𝑘*f = *𝑘* = 1

f

(4)

*𝑘̄*

eff =

avg

(*𝜃*hot − *𝜃*cold)

(10)

*𝑘̄*

eff

= *𝑘*eff

*𝑘*s

Thus, the steady-state dimensionless heat equation is shown Eq. ([5](#_bookmark14))

* 1. *NHT solution and details*

This study used the tridiagonal matrix algorithm (TDMA) iteratively

*𝜕* (*𝑘̄ 𝜕𝜃* ) + *𝜕* (*𝑘̄ 𝜕𝜃* ) = 0 (5)

*𝜕𝜉*

*𝜕𝜉*

*𝜕𝜂*

*𝜕𝜂*

control volume size *𝛥𝜉* = *𝛥𝜂*, which further simplifies the finite volume This model setup yields a uniform structured mesh, meaning each

convention for *i + 1*, *𝑖* − 1, *j + 1*, and *j* − *1*, respectively. For the central scheme. It is convenient to adopt the East, West, North, and South

location *i,j*, we adopt the letter P. It is also important to note that all heat fluxes are calculated at the interfaces between two adjacent cells, hence the thermal conductivity *k* used in the interface is the harmonic average between the two thermal conductivities.

We use the lowercase east, west, north, and south notation for the interface between cells. Since all cells have the same size, the harmonic

mean for *𝑘̄* w at the west interface of the central cell can be calculated

using Eq. ([6](#_bookmark15)) below

2 ⋅ *𝑘̄ 𝑘̄*

to maximize the efficiency of both memory storage and computational

time. The TDMA approach deals with the three-diagonal system of

linear equations. In testing, the iterative TDMA was 40% faster than the Jacobi iteration using the same convergence criteria.

There are two options in terms of parallel algorithms to increase speed. The first is to implement a parallel solver, which would involve using more computer cores to speed up each iteration. The second option is to simply have separate computing cores working on different images and only sharing the results amongst each other. The second option requires a scheme to coordinate the load balancing between cores, which is naturally much more efficient and simple than the first option. Therefore, the NHT model was implemented in C code, using OpenMP for the load balancing. [Table](#_bookmark9) [1](#_bookmark9) summarizes the relevant prac- tical statistics from the NHT model. Note that the convergence speed and accuracy heavily depends on the ratio of thermal conductivities

*𝑘̄* w =

W P *,* (6)

*𝑘̄* W + *𝑘̄* P

of the solid and fluid phases as well as the overall connectivity of the solid phase. When the conductivity ratio between the solid and fluid

and the thermal conductivities at the other interfaces are calculated likewise.

The dimensionless temperature *𝜃*hot is set to 1 and *𝜃*cold is set to 0.

Those are the constant temperature boundaries enforced on the right

are adiabatic. The arrow *𝑞*′′ simply indicates the direction of heat flux, and left of the domain, respectively. The top and bottom boundaries

and there are no heat injections from the exterior. This setup is shown in [Fig.](#_bookmark10) [2](#_bookmark10).

From the boundary conditions, it is clear that the heat flux should

flux across the left boundary is defined as *𝑞*′′, the heat flux across the be the same across any vertical slice for a converged solution. The heat

l

right boundary is defined as *𝑞*′′

is 10, the NHT simulation per image per computer core takes 34.2 s on average. When the ratio of solid and fluid thermal conductivities is 341.7, the NHT simulations of the same images averaged over 370 s

per image per core. The same model was also evaluated for *𝑘̄* s of 50

and 100. Since each dataset has 130,000 images, a total of 520,000 2D

cases were simulated and used for CNN training.

* 1. *NHT validation*

Parallel and series-arranged structures are used to validate the NHT simulations, as the theoretical solution arises from thermal resistance

′′ ′′

r , and since the heat flux is supposed

models [[8](#_bookmark54)]. *𝑘̄* eff is a function of the thermal conductivity of solid and

to be constant, *𝑞*l − *𝑞*r is chosen as the convergence criteria for the simulation. An example of how to calculate *𝑞*′′ for *n* cells at the left

l

fluid phases, *𝑘̄* s and *𝑘̄* f, as well as the porosity, *𝜖*. The theoretical values

are calculated as follows:

boundary is shown in Eq. ([7](#_bookmark17)).

*𝑞*′′ = 1 ∑ *𝑘̄ 𝜃*i − *𝜃*cold

*𝑛*

(7)

*𝑠𝑒𝑟𝑖𝑒𝑠* ∶ *𝑘̄*

eff =

*𝜖*

*𝑘̄*

(

+ 1 − *𝜖* −1

*𝑘̄*

)

(11)

l *𝑛*

i

*𝑖*=1

*𝛥𝜉*

2

f s

*̄ ̄ ̄*

where *𝜃*i is the dimensionless temperature at each of the *n* cells adjacent

to the left boundary. Similarly, the heat flux at the right boundary can

be calculated as in Eq. ([8](#_bookmark20)).

*𝑝𝑎𝑟𝑎𝑙𝑙𝑒𝑙* ∶ *𝑘*eff = *𝜖* ⋅ *𝑘*f + (1 − *𝜖*) ⋅ *𝑘*s (12)

In the NHT simulations, the 128 × 128 images for the parallel

and series structures were generated for five different porosities. Seven

*𝑞*′′ = 1 ∑ *𝑘̄ 𝜃*hot − *𝜃*i

*𝑛*

(8)

different ratios of *𝑘̄* s

were simulated. The results are benchmarked

r *𝑛*

i

*𝑖*=1

*𝛥𝜉*

2

versus the analytical relations in Eqs. ([11](#_bookmark18)) and ([12](#_bookmark19)) are displayed in [Fig.](#_bookmark21) [3](#_bookmark21).

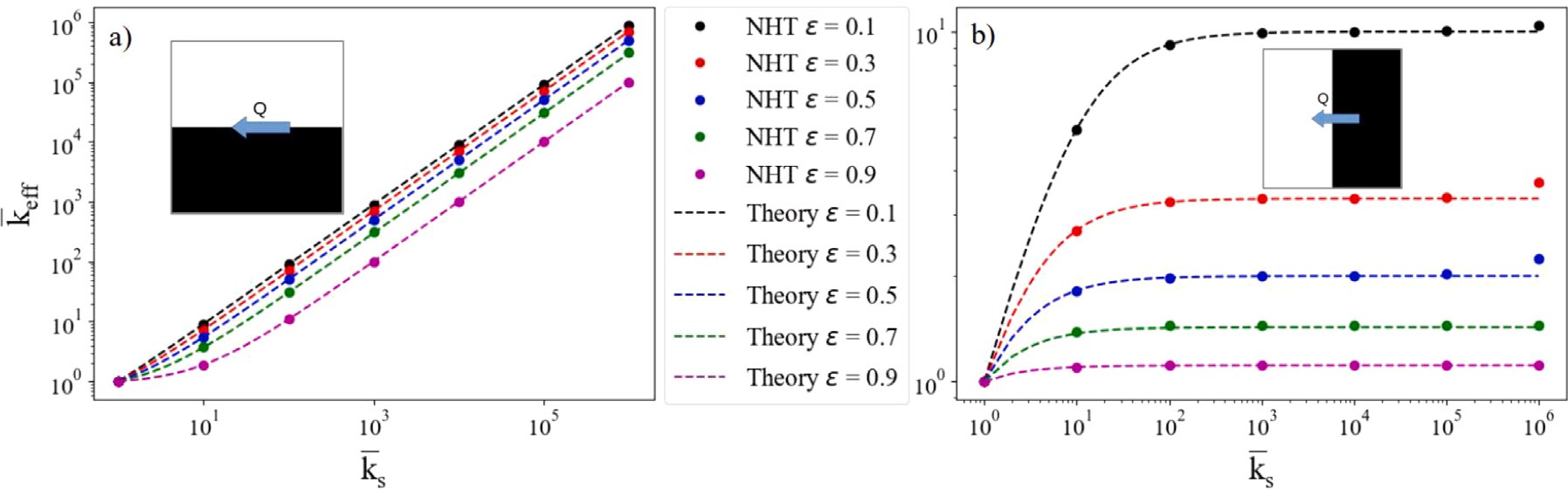
where in this case *𝜃*i is the dimensionless temperature at each of the *n*

cells adjacent to the right boundary. Finally, since *𝑞*′′ and *𝑞*′′ should be

The largest relative error from any NHT result compared to the

theoretical from Eq. ([12](#_bookmark19)) for the parallel case is in the order of 10−7.

l r



**/ig. 3.** Theoretical results versus the NHT model. The parallel arrangement results are on the left, and the series arrangement results are on the right.

However, upon inspection of [Fig.](#_bookmark21) [3](#_bookmark21), it is clear that there is a much larger relative error for the series case. The validation shows that the lower the conductivity ratio, the less error is introduced. For a porosity of 0.1, the relative error reaches a maximum of 4.5%, when the thermal

conductivity ratio is 106. Introducing such errors into the training of

ratio of thermal conductivities is kept below 104 for the remainder of the CNN can be detrimental to the CNNs predictive ability. Hence the

the study. As the conductivity gradients are more moderate, the central difference algorithm is good for discretizing the governing equation. The coefficient matrix from the NHT model has a lower condition number when the conductivity ratio is approaching 1, which is also evident from the accuracies displayed in [Table](#_bookmark9) [1](#_bookmark9).

In [Appendix](#_bookmark45) [B](#_bookmark45) the accuracy of the finite volume method is compared with two other studies [[32](#_bookmark81),[33](#_bookmark82)]. These studies employed analogous

theoretical test cases to assess the precision of the LBM.

**Table 2**

Effect of number of convolutional layers on training and validation accuracy.

|  |  |  |
| --- | --- | --- |
| Conv layers | MAPE Training | MAPE Test |
| 2 | 1.11% | 4.27% |
| 4 | 1.18% | 4.04% |
| 6 | 0.96% | 2.58% |
| 8 | 1.09% | 2.53% |
| 12 | 0.92% | 1.09% |
| 16 | 0.93% | 0.90% |

effective thermal conductivity. The loss function is the mean absolute percent error (MAPE) function from the Keras API. For *n* samples, the equation to calculate MAPE is shown in Eq. ([13](#_bookmark23)).

*𝑛* |*𝑌* − *𝑌* |

*𝑖 𝑖*

∑

MAPE = 100%

*𝑛*

true

*𝑖*

true

pred

**3. CNN selection & optimization**

*𝑖*=1

*𝑌*

(13)

A multitude of CNNs are available for the problem of image seg- mentation and detection. For the problem of predicting physical prop- erties based on geometry, the ideal network provides a combination of sufficient accuracy with a reasonable computational cost. The VGG- class [[20](#_bookmark66)] networks, namely VGG16 and 19, are peculiar in the way the layers are stacked, having some unique advantages that are particularly relevant in terms of efficiency, memory storage, and training speed. The

VGG networks only use 3 × 3 filters in the convolutional layers with a

stride of 1. The 3 × 3 filter is the smallest filter that can distinguish up,

multiple layers of 3 × 3 filters before a max pooling operation. As an down, left, and right. An effective receptive field is created by stacking example, if two 3 × 3 layers are stacked, an effective receptive field of 5 × 5 is created. As explained in the original manuscript for the VGG

the network: a four-layer stack of 3 × 3 filters has an effective field network [[20](#_bookmark66)], the main advantage of this practice is the efficiency of of view of 9 × 9, while only storing and training 4(32*𝐶*2) parameters, where C is the number of channels. In total, that is 36*𝐶*2 parameters. On the other hand, a regular layer with a 9 × 9 filter has 92*𝐶*2 or 81*𝐶*2 parameters, which requires 125% more parameters for the same

effective field of view.

Other than architectural advantages, selecting appropriate hyperpa- rameters for the chosen network is an essential step that often requires a lot of trials and a general understanding of machine learning techniques and algorithms [[34](#_bookmark83)]. For this section, since the optimization process involves an extensive trial of the network, a subset of the dataset containing only the original 30,000 regular shape-packed images is employed, thus reducing computational time and allowing for a deep and extensive search for ideal architecture and hyperparameters. The ratio of thermal conductivities used is set to 10 to minimize the computational error and computational time of the NHT model. The objective of the study is to tackle ambiguities in the prior art and justify the selection of general hyperparameters for problems that focus on

* 1. *CNN architecture optimization*

The VGG network and most other networks used in literature are designed to excel at image classification tasks. The scope of this work is to predict physical properties using the extracted feature maps. Hence

this study uses 128 × 128 × 1 binary images as the input as opposed

to 224 × 224 × 3 RGB images used in the original VGG-network.

Other adaptations to the network must also follow to optimize it for

the tasks of predicting effective thermal conductivity. Wu et al. [[14](#_bookmark60)] conclude from their work that the mean-squared error (MSE) trends to be smaller when the CNN has fewer convolutional layers. Similarly, Zhu et al. [[18](#_bookmark64)] arrived at a similar conclusion: the best accuracies in training and testing for effective thermal conductivity prediction were achieved when the CNN had 2 convolutional layers. Conversely, others [[20](#_bookmark66),[35](#_bookmark84)] suggest that deeper networks are more capable of extracting relevant features. The deepest layers, as well as the fully-connected (FC) layers play a major role in the accuracy of prediction [[36](#_bookmark85)].

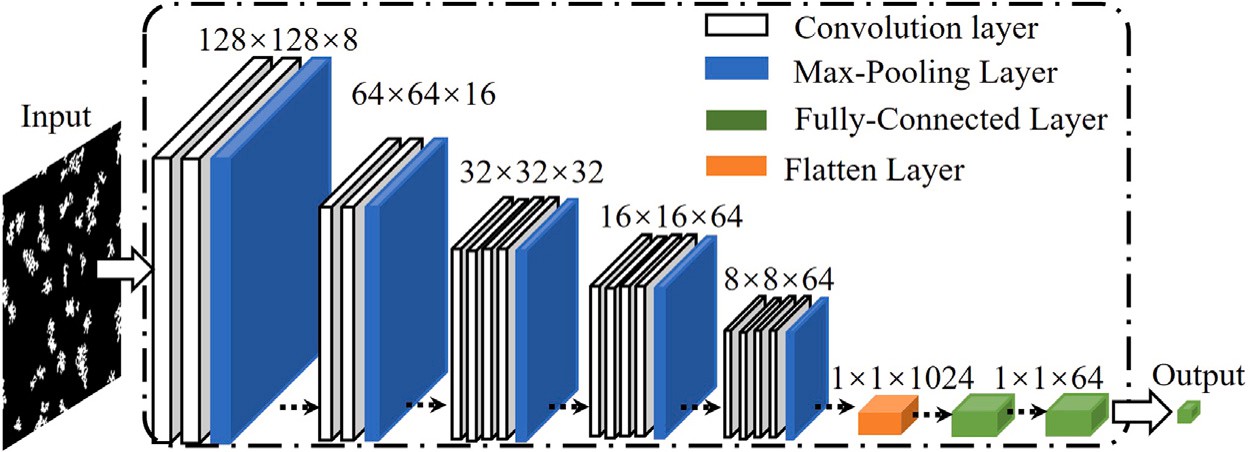
In this study, the number of nodes is increased from 2/2/1 to 128/128/1 in the FC layers to compare which architecture has the highest accuracy in testing and training. All convolutional layers and FC layers have the rectified linear unit (ReLU) activation function. The training is performed in 60 epochs with the Adam optimizer learning

rate of *𝜆* = 0*.*001 and the batch size of 32. The training uses the same

80/20 train/test split for the trials shown in [Table](#_bookmark22) [2](#_bookmark22) while holding the

3 FC layers constant. The first two FCs have 64 nodes, and the last FC has one node as the model output.

The architecture with 16 convolutional layers looks the same as VGG19, which has 16 convolutional layers and 3 FC layers. Based on the trend in test MAPE displayed in [Table](#_bookmark22) [2](#_bookmark22), the more information is encoded through the convolutional layers, the more accurate the predictions in testing get. It is clear that overfitting is more prominent with fewer convolutional layers when the VGG-inspired network is used



**/ig. 4.** Architecture compiling the results from this section outlining the best performing network.

**Table 3**

Effect of number of FC nodes on training accuracy, validation accuracy, and number of parameters.

**Table 4**

Training outline and accuracy for selected combinations of parameters, all using 30,000 images.

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| FC Nodes | MAPE Training | MAPE Test | Trainable parameters |  | Model | Batch Size | Learning Rate | Epochs | MAPE Train | MAPE Test |
| 2/2/1 | 44.68% | 44.65% | 321,809 |  | A | 32 | 10−4 | 600 | 0.52% | 0.51% |
| 4/4/1 | 1.07% | 0.96% | 317,653 |  | B | 32 | 10−4 | 1600 | 0.28% | 0.53% |
| 8/8/1 | 1.04% | 0.90% | 321,587 |  | C | 64 | 10−4 | 600 | 0.37% | 0.85% |
| 16/16/1 | 1.00% | 0.86% | 330,217 |  | D | 32 | 10−5 | 2000 | 0.33% | 0.77% |
| 32/32/1 | 1.15% | 0.85% | 347,417 |  |  |  |  |  |  |  |
| 64/64/1 | 0.99% | 0.93% | 383,353 |  |  |  |  |  |  |  |
| 128/128/1 | 0.99% | 1.09% | 461,369 |  |  |  |  |  |  |  |

for problems focusing on the effective thermal conductivity. Hence, the remainder of this study uses 16 convolutional layers, as displayed in [Fig.](#_bookmark25) [4](#_bookmark25).

Another aspect we considered is the number of nodes on the FC layers. In this architecture, the convolutional layers’ job is to extract useful features to predict the relevant physical property. The FC layers’ job is to make the actual prediction. With too many nodes in the FC layers, the network tends to overfit regardless of the convolutional layers’ architecture. On the contrary, the network will be inaccurate without enough nodes. Additionally, the number of nodes of the FC layers heavily influences the number of trainable parameters in the network, which influences the training efficiency, memory storage, and prediction speed. We analyze the number of nodes on each FC layer using a process similar to the one used to determine convolutional layer design. Originally, the VGG19 for the classification problem with 1000 possible outputs has 4096/4096/1000 nodes on the fully connected layers. Those numbers are not suitable for this project. For example, we only need one output in the third layer. While keeping all training parameters the same as the study on the number of convolutional layers, we changed the numbers of nodes in the FC layers. Results are laid out in [Table](#_bookmark26) [3](#_bookmark26).

The number of nodes does not seem to have a big effect on the prediction accuracy as long as the number exceeds a minimum number. When the numbers of nodes are as small as 2/2/1, the prediction is highly inaccurate. When the numbers of nodes increase to 4/4/1 and to 64/64/1, all testing and training results have reasonable accuracy. We adopt 64/64/1 as it provides more consistency between testing

complex problems but at the cost of having 20% more parameters than and training accuracy while reserving sufficient nodes to tackle more

4/4/1. Once the numbers of nodes are increased to 128/128/1, we observed slight trends towards overfitting.

Finally, the last aspects of the network’s architecture that could be optimized are the grouping of the convolutional layers. After testing a few combinations of the number of channels, we find that the number of channels per convolutional layer does not significantly impact the accuracy. We adopt the configuration shown in [Fig.](#_bookmark25) [4](#_bookmark25) in order to balance the number of parameters. CNNs for classification problems use the *softmax* function in the last layer. But this study focuses on property prediction and uses a simple linear activation function in the last layer.

* 1. *CNN training*

The training of a CNN is a sensitive process that plays an intrinsic role in the ability of the model to extrapolate from training data to new data. A variety of methods are available, some of which include different mechanisms for decaying the learning rate as a function of the training epoch. Some recent studies suggest that instead of decreasing the learning rate, increasing the batch size is preferred [[37](#_bookmark86)]. We used the Adam [[38](#_bookmark87)] optimizer, which is notable for automatically handling learning rate decays. Hence the entered learning rate serves only as an upper limit, which in turn facilitates the optimization process by removing the momentum-related parameters.

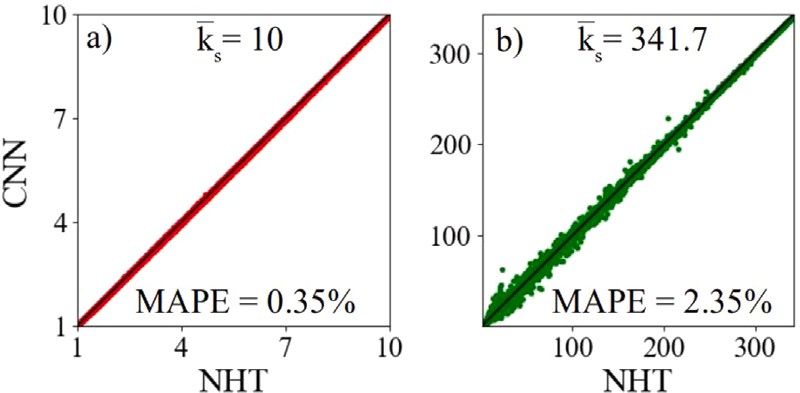
This study has considered a wide range of learning rates, number of epochs, and batch sizes. But this section only highlights some combina- tions to display how specific parameters affect the accuracy and justify the final decision on which schedule to use. All of them were used to train the network shown in [Fig.](#_bookmark25) [4](#_bookmark25). [Table](#_bookmark27) [4](#_bookmark27) outlines the parameters of 4 different training schedules with the accompanying MAPEs in training and testing for our network.

Model A presents the best consistency between training and valida- tion, does not overfit, and oscillates the least in later epochs. Model B performs better in training than in testing, which indicates a certain de- gree of overfitting. Model B is the one outlined in prior art [[14](#_bookmark60)] for the problem of effective diffusivity. When model B is applied to the current problem, the number of epochs is excessive and does not provide any benefits in testing compared to model A. On the other hand, increasing the batch size for model C only results in more overfitting and higher MAPE compared to the similar Model A. Increasing the batch size from Model A creates more overfitting, while decreasing batch size did not provide any benefits. Finally, lowering the learning rate and training for longer in model D does not improve the overall MAPE and generates the most overfitting. However, we note that model D considerably decreases MAPE oscillations in later epochs when compared to the other models. A more general and diverse dataset would neutralize the oscillations observed in models A, B, and C.

# Results

* 1. *Accuracy in training and testing*

considers four values of *𝑘̄* s: 10, 50, 100, and 341.7. The ratio of 341.7 This subsection uses the entire 130,000 images in the dataset and



26,000 images on the testing dataset. **(a)** predictions from the model using *𝑘̄* s of 10. **/ig. 5.** CNN predictions versus NHT model effective thermal conductivity for the

**(b)** predictions on the model trained with an aluminum-water combination.

**Table 5**

Statistics showing the effect of increasing the *𝑘̄* s on the accuracy of the

CNN prediction in training and testing.

|  |  |  |
| --- | --- | --- |
| *𝑘̄* s | MAPE Train | MAPE Test |
| 10 | 0.25% | 0.35% |
| 50 | 0.53% | 1.08% |
| 100 | 0.61% | 1.74% |
| 341.7 | 0.76% | 2.35% |

represents an aluminum-water system. All cases split 80% of the data for training and 20% for testing using the parameters for model A ( [Table](#_bookmark27) [4](#_bookmark27)). [Fig.](#_bookmark29) [5](#_bookmark29) compares between CNN-predicted results versus the NHT-simulated results of each of the 26,000 images in the testing dataset for the conductivity ratios of 10 and 341.7. Each CNN model was trained with 104,000 training data.

increase for (b), with a higher *𝑘̄* s. For the four ratios considered, [Table](#_bookmark30) [5](#_bookmark30) The visual inspection of [Fig.](#_bookmark29) [5](#_bookmark29) shows that the prediction errors

outlines the MAPEs and confirms the trend in decreasing accuracy with increasing thermal conductivity ratio. Section [2.4](#_bookmark16) and [Table](#_bookmark9) [1](#_bookmark9) also show increasing errors in NHT simulation as the thermal conductivity ratios increase, hence the result displayed in [Fig.](#_bookmark29) [5](#_bookmark29) and [Table](#_bookmark30) [5](#_bookmark30) hints at the error in CNN being a reflection and amplification of error introduced from NHT simulations. Also, note that the results outlined in [Table](#_bookmark30) [5](#_bookmark30) are for the model presented in [Fig.](#_bookmark25) [4](#_bookmark25) along with training scheme A from [Table](#_bookmark27) [4](#_bookmark27), with the only difference that the full dataset was employed in this section as opposed to the 30,000 images employed in Section [3](#_bookmark24). With the addition of the full dataset to the training process, the MAPE of the lowest thermal conductivity ratio dropped from 0.51% to 0.35% in testing.

* 1. *Performance*

Regardless of the thermal conductivity ratio, the batch CNN predic- tion time per batch is about 4 ms on a node with 4 NVIDIA V100 GPUs in the Expanse supercomputer. The CNN prediction time on an off-the- shelf NVIDIA GeForce RTX3070 is in the order of 15 ms. On a single

CPU core, the CNN prediction time is about 20 ms when using a 3*.*6 GHz

AMD Ryzen 7 3700X CPU. With the average NHT times reported in

[Table](#_bookmark9) [1](#_bookmark9), using one CPU core, the CNN is 3 to 5 orders of magnitude faster than the NHT code.

The training of the CNN model takes around an hour to load the data and run 600 epochs on one HPC node with 4 NVIDIA V100 GPUs. We did not evaluate this process on a regular office computer due to constraints in RAM when loading the entire dataset.

* 1. *Validation in theoretical cases*

trained for the four values of *𝑘̄* s. The results are benchmarked versus The same test cases used in Section [2.4](#_bookmark16) can be fed into the CNNs

theoretical solutions from Eqs. ([11](#_bookmark18)) and ([12](#_bookmark19)), and shown in [Fig.](#_bookmark33) [6](#_bookmark33).

When the conductivity ratio is 10, the MAPE is 0.44% for the parallel case and 1.61% for the series case. When the aluminum-water

system is simulated, the MAPE is 0.72% for the parallel case, and 1.57% for the series case. These results are consistent with trends observed in the previous subsection: larger errors (more than twice the MAPE) are observed when the effective properties approach that of the fluid phase. When the conductivity ratio is 10, the MAPE of the predictions is larger than the MAPE in testing shown in [Table](#_bookmark30) [5](#_bookmark30). But for the aluminum-water case, the MAPE of the predictions remains well below that in [Table](#_bookmark30) [5](#_bookmark30) for test. Finally, notice how the error for lower porosity presented in [Fig.](#_bookmark33) [6](#_bookmark33)(b) resembles the error from [Fig.](#_bookmark21) [3](#_bookmark21)(b), further evidence to support that the NHT accuracy is intrinsic to the CNN accuracy.

# CNN-aided topology optimization

While there are several topology optimization approaches, this study simplifies the process to swapping voxels in the solid phase for voxels in the fluid phase. The objectives of this section are twofold: demonstrating that the CNN can present reasonable accuracy in real- world scenarios involving data outside of the training dataset, as well showing a potential application of the CNN in non-gradient iterative topology optimization as an efficient alternative to a conventional sim- ulated physical property prediction. Thus, the focus is not on devising a unique optimization scheme, instead, the focus is on assessing the accuracy and efficiency of the CNN in different scenarios. The only restriction when picking the voxels for swapping is to pick voxels at the interfaces between the solid and fluid phases. In other words, the randomly selected voxel must have at least one adjacent voxel belonging to a different phase. During each iteration, if the voxel swap results in higher thermal conductivity or cost function, the swap is accepted. Otherwise, the two picked voxels will be returned to their original coordinates.

The simple voxel swapping approach can have large amounts of

A simple example employing a CNN on a 128 × 128 voxel domain noise, thus taking millions, if not billions of iterations to converge [[29](#_bookmark78)]. yielded more than 16,000 successful swaps. The *𝑘̄* eff converged to within 10% of the theoretical maximum. We chose to bundle 4 × 4

pixels together during the swap to save computational time.

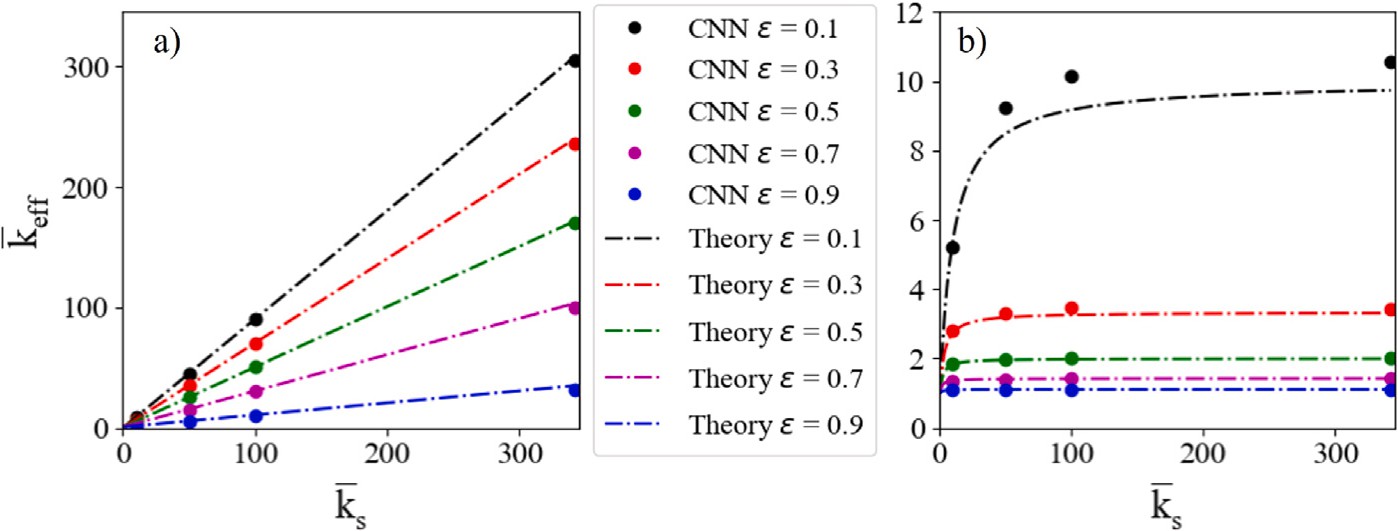
The aforementioned process of random voxel swapping can lead to local maxima, where the thermal conductivity is not close to the theoretical maximum and yet no successful swaps can be found. For that reason, the simulated annealing algorithm (SAA) [[39](#_bookmark88)] is employed to determine if mutations occur or not. If the value of the cost function decrease, either the swap is reverted or a mutation occurs. Mutations are swaps that, albeit counterproductive to the optimization process from the point of view of the cost function, get accepted to avoid local minima/maxima. We applied the CNN model and SAA for one

million iterations on a 32 × 32 domain of voxels with a porosity of

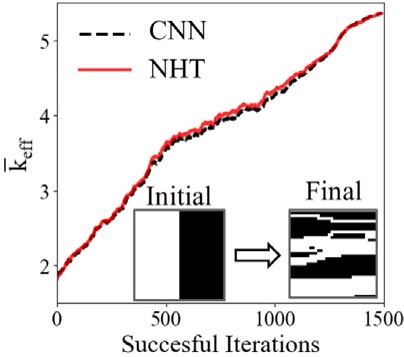
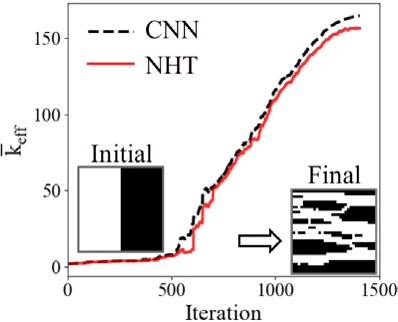
0.5 as an example. We showcase four optimization cases: maximizing

the thermal conductivity in the *x*-direction with a thermal conductivity ratio of 10; maximizing the thermal conductivity of the aluminum- water combination with a conductivity ratio of 341.7 in the *x*-direction; optimizing the heat conduction in both *x* and *y* directions simulta- neously with a thermal conductivity ratio of 10; and optimizing the thermal conductivity in the *x*-direction while also maximizing surface area. The initial phase configuration for all cases is the two phases connected in series, which has the lowest theoretical effective thermal conduction in the *x*-direction and the theoretical maximum in the *y*-direction.

The major benefit of CNN over NHT simulation is the prediction speed. From the benchmarks in Section [4.2](#_bookmark31), an aluminum-water NHT simulation would take, on average, 370 s per iteration. That would require around 102,778 h to attempt one million voxel swaps in topol- ogy optimization. In comparison, CNN completed one million topology optimization iterations on a commercial NVIDIA GEFORCE RTX 3070 in four and a half hours, with an average prediction speed of 15 ms per iteration.



**/ig. 6.** Theoretical results versus the CNN model. (a) Parallel configuration cases are on the left. (b) Series configuration results.

**/ig. 7.** Evolution of the *𝑘̄* eff from topology optimization, comparing the NHT and CNN predictions for every step, as well as the evolution from initial to final structure.

* 1. *Case A: 𝑘eff in the x-direction, 𝑘̄ s* = 10

Within the one million swaps attempted, 1490 were deemed suc- cessful by CNN, and 911 mutations (not shown in [Fig.](#_bookmark34) [7](#_bookmark34)) occurred

in which the swaps were accepted. [Fig.](#_bookmark34) [7](#_bookmark34) shows the final result of the topology optimization, which starts from a theoretical worse case

scenario. The final structure has a *𝑘̄* eff of 5.36 according to the NHT

model, which is 2.58% lower from the theoretical maximum of 5.5

has a *𝑘̄* eff of 5.358, which only has an error of 0.075% compared with calculated from Eq. ([12](#_bookmark19)). The CNN algorithm predicts the final structure

the NHT prediction. [Fig.](#_bookmark34) [7](#_bookmark34) shows the CNN predictions for all accepted successful swaps and the NHT evaluation of the same structures.

The CNN is able to accurately predict the *𝑘̄* eff at each iteration. The

predicted change of *𝑘̄* eff with iteration by CNN follows a similar pattern

to that of the NHT. The MAPE between the NHT and CNN models

prediction for all successful swaps is 1.1%, which is worse than the reported testing MAPE from [Table](#_bookmark30) [5](#_bookmark30). But the higher MAPE is expected due to the random nature of structures resulting from voxel swapping. This trend can be confirmed from [Fig.](#_bookmark34) [7](#_bookmark34): the NHT and CNN predictions

agree at both low and high *𝑘̄* eff values, with a bulge of disagreement

in the middle of the domain, where the structures resemble random

distributions of voxels.

In this topology optimization case the NHT simulation averages

34.2 s per iteration (from [Table](#_bookmark9) [1](#_bookmark9)), and the CNN averages 15 ms per iteration on a commercial GPU. The CNN based optimization is 2280 times faster per iteration, which saves over 9,495 h in the span of one million iterations.

* 1. *Case B: 𝑘eff in the x-direction, 𝑘̄ s* = 341*.*7 *(Aluminum-water)*

Effective properties of the aluminum-water system predicted by CNN, as seen in Section [4](#_bookmark28), have considerably higher errors than the case with the thermal conductivity ratio of 10. The ratio of thermal conductivities is close to other metals filled with fluids, so this is a case with more real applications. If the topology optimization using CNN correctly predicts the trend in thermal conductivity in the aluminum- water system with changes to the structure, then this modeling tool can be extrapolated to other material combinations.

**/ig. 8.** Evolution of the *𝑘̄* eff from topology optimization, comparing the NHT and CNN

predictions for every step, along with initial and final structures.

In this case, 1406 iterations were successful, and 743 mutations occurred. The final image, displayed in [Fig.](#_bookmark36) [8](#_bookmark36) has a *𝑘̄* eff of 156.43

according to the NHT model, while the CNN predicts the ratio to be 164.81, which is 5.36% higher than the NHT model. The theoretical maximum according to Eq. ([12](#_bookmark19)) is 171.35. Hence the NHT prediction

shows the optimized structure achieved a *𝑘̄* eff within 9.12% from the

theoretical maximum. The evolution of *𝑘̄* eff with successful voxel swaps

is displayed in [Fig.](#_bookmark36) [8](#_bookmark36), with the NHT results for every structure also

included for comparison.

Throughout the optimization, the CNN and NHT follow the same

increases in the *𝑘̄* eff. CNN predictions’ MAPE relative to NHT results general trend, which confirms that the CNN model correctly predicts

was 11.84% on the successful iterations. The relative increase in MAPE compared to case A is similar to what is observed through validation in Section [4](#_bookmark28). The increase in error from testing, similar to Section [5.1](#_bookmark35), is attributed to intermediate structures that are a natural outcome of random voxel swapping. As stated before, the regular shapes packed in the majority of the training dataset are there to simulate realistic structures. It is noteworthy that besides the reported accuracy, in both cases A and B, the CNN and NHT models agree better at the initial state and near the converged state, as seen in [Fig.](#_bookmark36) [8](#_bookmark36). Near the initial and converged states, the structures are better connected and more resembling the training data.

With *𝑘̄* s = 341*.*7, the NHT simulation averages 371.2 s per iteration

(from [Table](#_bookmark9) [1](#_bookmark9)). The CNN prediction time is not dependent on *𝑘̄* s, and

still averages 15 ms per iteration on a commercial GPU. The CNN

based optimization is 24,746 times faster per iteration, which saves over 103,105 h in the span of one million iterations.

* 1. *Case C: 𝑘eff in both x and y directions, 𝑘̄ s* = 10

This case simultaneously optimizes *𝑘̄* eff for the *𝑥* and *𝑦* directions, with a cost function defined in Eq. ([14](#_bookmark37)):

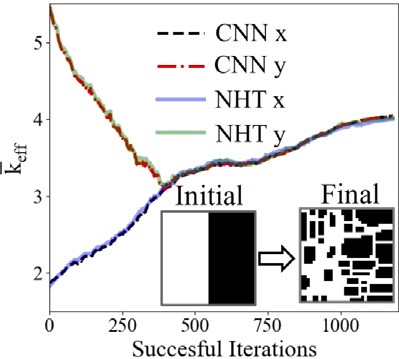
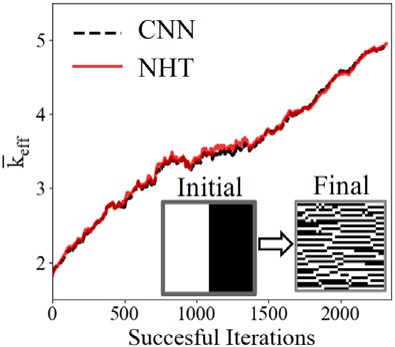
cost = *𝑘̄* eff,x + *𝑘̄* eff,y − *𝑘̄* eff,x − *𝑘̄* eff,y (14)

| |

The cost function includes the effective thermal conductivities in the

*x*-direction, *𝑘̄* eff,x, and *𝑦*-direction, *𝑘̄* eff,y. The simulation only considers

the thermal conductivity ratio of 10. The absolute difference between

**/ig. 9.** Evolution of the *𝑘̄*

eff,x

and *𝑘̄*

eff,y

from topology optimization, comparing the NHT

**/ig. 10.** Evolution of the *𝑘̄* eff from topology optimization, comparing the NHT and

CNN predictions for every step. The final structure is also compared to the original

and CNN predictions for every step, along with the final structure generated.

*𝑘̄* eff,x and *𝑘̄* eff,y is added to the cost function such that the algorithm does not maximize one of the properties to the detriment of the other, thus avoiding local maxima outcomes. The penalizing term ensures that

*𝑘̄* eff,x and *𝑘̄* eff,y will always be similar through the CNNs prediction, thus

ensuring *𝑘̄* eff of the structure is an isotropic property in the 2D plane.

image.

The cost function is then expressed in Eq. ([17](#_bookmark40)), where the terms

*𝑤*k and *𝑤*SA are weights introduced to regulate which parameter the

are assigned equal importance, so *𝑤*k = *𝑤*SA = 1. topology optimization algorithm prioritizes. In this study both variables

Within the one million swaps attempted, 1180 of them were suc- cessful, and 593 mutations occurred. The final structure can be seen in

*̄*

k eff,norm

cost = *𝑤 𝑘*

+ *𝑤*SA

*𝑆̄𝐴* (17)

[Fig.](#_bookmark38) [9](#_bookmark38), along with the comparison of NHT and CNN predictions for *𝑘̄* eff,x

and *𝑘̄* eff,y. The cost function of the final structure predicted by the CNN

is 8.08. In comparison, the cost function simulated by the NHT model

is 8.02 for the final structure.

When using a GPU, the CNN model can evaluate both images (x and

*𝑦* direction) simultaneously, hence taking the same time as evaluating

one image alone, and additional advantage of the CNN model. In the

entire domain, the CNN model has a MAPE of 0.67% compared to the NHT model, with trends similar to the ones seen in [Fig.](#_bookmark34) [7](#_bookmark34).

With *𝑘̄* s = 10, the NHT simulation averages 34.2 s per iteration,

However in this case the NHT has to simulate both *𝑥* and *𝑦* directions and the CNN averages 15 ms per iteration on a commercial GPU.

independently, hence taking twice as long. Because modern GPUs can

of time to evaluate both *𝑥* and *𝑦* directions simultaneously. Therefore, take batches of images, even a commercial GPU takes the same amount

GPU-based estimation is 4560 times faster per iteration, which saves over 18,990 h in the span of one million iterations.

* 1. *Case D: 𝑘eff in the x-direction, 𝑘̄ s* = 10*, and surface area*

Surface area is an important parameter in energy related heteroge- neous systems. This case study is meant to showcase how the output of the CNN can be used alongside another variable which does not have the same units for optimization. The first step in combining thermal conductivity and surface area is normalizing and non-dimensionalizing both quantities.

Since *𝑘̄* eff,parallel represents the maximum possible *𝑘̄* eff and *𝑘̄* eff,series

represents the minimum, for a given porosity, *𝜖*, and a thermal conduc- tivity ratio, *𝑘̄* s, *𝑘̄* eff is normalized by Eq. ([15](#_bookmark42)).

From the one million attempted voxel swaps, using *𝑘̄* s = 10, 2,319

iterations were successful and 1349 mutations occurred. The *𝑘̄* eff pre-

dicted by the CNN is 4.93 and simulated by the NHT model it is 4.95.

The error in CNN prediction relative to the NHT model is 0.4% for the final image, displayed in [Fig.](#_bookmark39) [10](#_bookmark39). The MAPE in prediction, from data shown in [Fig.](#_bookmark39) [10](#_bookmark39), is 0.88%, slightly better than what was displayed in

case A. The NHT calculated *𝑘̄* eff for the final structure is 10% off from

the theoretical maximum of 5.5, while *𝑆̄𝐴* from Eq. ([16](#_bookmark43)) is 0.467.

The results from this study are better put into perspective when

compared to the results displayed in [Fig.](#_bookmark34) [7](#_bookmark34), in which the same condi- tions for *x*-direction thermal conductivity were optimized but without any constraint to the surface area. It is clear by visually comparing [Figs.](#_bookmark34) [7](#_bookmark34) and [10](#_bookmark39) that the latter presents a much larger surface area,

while the *𝑘̄* eff is only 3.8% lower. Holding similar weighing factors for

the surface area and *𝑘̄* eff in the cost function in Eq. ([17](#_bookmark40)) penalizes the

minimal increase in *𝑘̄* eff. The CNN maintained the ability to correctly optimization algorithm to move a pixel with a large surface area for a predict trends in *𝑘̄* eff for structures with high surface areas and optimize

the cost function.

With *𝑘̄* s = 10, the NHT simulation averages 34.2 s per iteration, and

the CNN averages 15 ms per iteration on a commercial GPU. The time

taken to evaluate surface area is negligible in both cases. Therefore, the speedup observed here is the same as Section [5.1](#_bookmark35).

# Conclusion

The proposed model provides an on-demand alternative to estimat- ing effective thermal conductivity on a system with two phases without any compromises or simplifications to the geometry of the system.

*𝑘̄* eff,norm =

*𝑘*

*̄*

*𝑘̄* eff − *𝑘̄* eff,series

– *𝑘̄*

(15)

The trained networks at four different ratios of thermal conductivity among the two phases are simple to use and produce results orders of

eff,parallel

eff,series

magnitude faster than NHT simulations. The model is also capable of

While it seems natural to use *𝑘̄* eff,norm as the output of the CNN instead of training it for multiple values of *𝑘̄* s, for any given structure,

*𝑘̄* eff,norm does change with the ratio of *𝑘̄* s. Currently there is no way

to extrapolate *𝑘̄* eff,norm from one thermal conductivity ratio to all ra- tios. [Appendix](#_bookmark49) [C](#_bookmark49) briefly explores the issues with using *𝑘̄* eff,norm as the

network output.

where the maximum surface area, *𝑆𝐴*max, for a porosity of *𝜖* = 0*.*5 For the surface area, a similar equation is proposed in Eq. ([16](#_bookmark43)), is a perfect checkerboard pattern, and the minimum, *𝑆𝐴*min, is a

where all voxels of any given phase are clumped together. *𝑆𝐴* is also configuration such as the parallel or series configurations from [Fig.](#_bookmark21) [3](#_bookmark21),

normalized, using Eq. ([16](#_bookmark43)).

extrapolating from the learned data to new structures, as seen in the case of topology optimization through voxel swapping. In that case, the intermediate structures do not appear realistic and were not directly encoded in training, yet the model retains reasonable consistency and accuracy in prediction.

Through the studies presented, we determined that the two main sources of error are the NHT model used to evaluate the training dataset and the diversity of the dataset. Thus, the main restriction of creating highly efficient and accurate CNNs to predict properties, such as effective diffusivity, structural properties, effective permeability, average pore size, and surface area, among others, is the creation of a diverse and accurate training datasets. The training of the CNN itself is

*𝑆̄𝐴* = *𝑆𝐴* − *𝑆𝐴*min

*𝑆𝐴*max − *𝑆𝐴*min

(16)

minuscule in terms of computational time compared to creating a large dataset. The full dataset used in this study [[25](#_bookmark74)], the models trained on

the four different thermal conductivity ratios, as well as example code to use the trained models, are available to the reader.

One critical limitation of the CNN model is that it is completely blind to physics, and does not produce a temperature or heat flux map of the domain as an output. In that sense, CNN cannot be paired with the Evolutionary Structural Optimization (ESO) [[40](#_bookmark89)] approach, which is a powerful method for taking educated guesses at potential voxel swaps in the structural optimization field. The CNN model can be improved by taking physical properties in addition to images, as input. In that case, the model would be able to generalize to a wider variety of material combinations at the cost of increased computational time.

Nonetheless, the efficiency and accuracy displayed by the CNN are attractive for both academic and industrial design of heterogeneous materials and on demand effective thermal conductivity estimation. Furthermore, this study shows that the accuracy and generalizability of the training dataset attribute to the majority of errors on CNN predictions. Thus, future research directions of the CNN model include increasing the size and improving the quality of the training dataset, adapting the model to three dimensions, and extending the model to other materials with different properties.

# Declaration of competing interest

The authors declare that they have no known competing finan- cial interests or personal relationships that could have appeared to influence the work reported in this paper.

# Data availability

The data will be publicly available.

# Acknowledgments

A. A. and X. L. highly appreciate the funding support from National Science Foundation, USA (Award: 1941083) and NASA EPSCoR, USA (Award: 80NSSC22M0221).

This work was only possible with the Extreme Science and Engi- neering Discovery Environment (XSEDE), USA grants for computational time in the Bridges2 and Expanse supercomputers, under the awards MAT210007 and MAT210014.

# Appendix A. Dataset details

fraction (SVF) and *𝑘̄* eff. Also, for any image generated, the *𝑘*eff is bound Often, there is a strong correlation between porosity/solid volume

to stay between the theoretical maximum and the theoretical minimum. With these two key ideas in mind, the dataset was specifically designed

By comparing [Fig.](#_bookmark47) [A.11](#_bookmark47) (a) and (b), it becomes evident how it is not only beneficial to augment the dataset in terms of encoding additional information, but also by flipping the color scheme of the images the augmented dataset has a more symmetrical distribution of SVF.

Note on [Fig.](#_bookmark47) [A.11](#_bookmark47) (c) the design space of *𝑘̄* eff as a function of SVF and

how it is bounded by the theoretical cases. Near the edges (SVF close

to 0 or 1), fewer images are needed since the theoretical maximum and minimum converge to the same value. On the other hand, between SVF of 0.3 and 0.7 is where the majority of images need to be to cover the design space.

Finally, from [Fig.](#_bookmark47) [A.11](#_bookmark47) (c) we observe that the theoretical cases used for validation were not encoded in training and are not part of this dataset. Otherwise the red dots would overlap with the dashed black line.

# Appendix B. /inite volume and LBM comparison

From the results presented in [Fig.](#_bookmark21) [3](#_bookmark21), the deviation from the NHT simulation and the analytical results is negligible for ratios of thermal

conductivity less than 104. Above that number, we note an increase in

error for the series structures, while the parallel structures still present

little to no error, as already discussed.

For comparison, Zhu et al. [[33](#_bookmark82)] utilizes similar theoretical test cases in 3D to validate a conventional LBM as well as a modified one in 3D for effective thermal conductivity. While numerical comparisons between LBM and theoretical cases are not explicitly expressed, by visual inspection we note significant differences between the models and theoretical results, especially the convectional LBM.

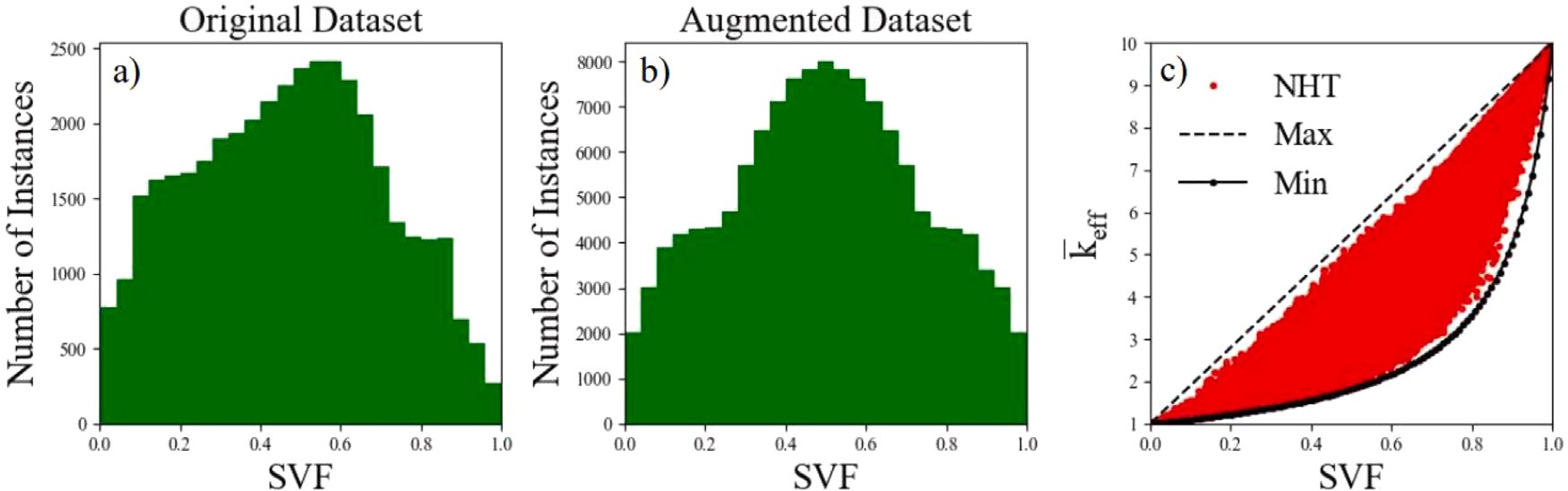
Ke et al. [[32](#_bookmark81)] proposed another modified LBM method for 2D effective thermal conductivity estimation. For the same theoretical test cases, with thermal conductivity ratios varying between 0.01 and 100, the authors reported deviations of less than 1.0% for both parallel and series test cases. For these thermal conductivity ratios, our error is less than 0.01% with the convergence criteria used in the training dataset. Neither study has additional data and results published in open access format, so we refrain from further comparisons. However, we do note that a version of our NHT simulation for effective thermal conductivity, both in 2D and 3D, is available on GitHub.[2](#_bookmark46) The entire dataset and all simulation results used to train the CNNs are also

available [[25](#_bookmark74)] to facilitate comparisons.

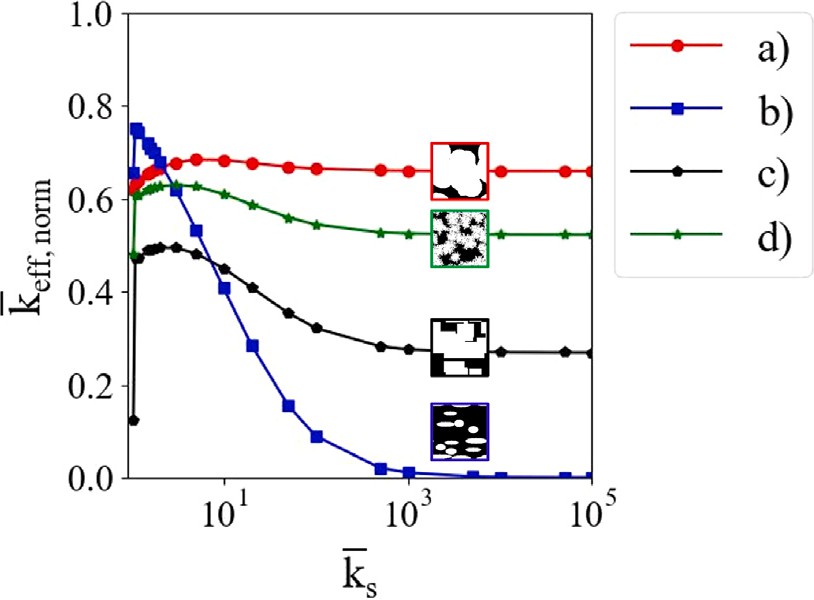
Regarding the NHT model, several mechanisms can augment accu- racy, such as mesh refinement, convergence criteria, and the prescribed cutoff iterations for the iterative solver. However, considering the foun- dational framework of this study involves the execution of hundreds of thousands of simulations, the determination of convergence criteria and the establishment of a maximum iteration count were strategic

to cover as diverse of a *𝑘̄* eff as possible. [Fig.](#_bookmark47) [A.11](#_bookmark47) below contains more

details. 2 <https://github.com/adama-wzr/Keff-CFD>



Compares the *𝑘̄* eff as a function of SVF for the entire augmented dataset. **/ig. A.11.** (a) shows a histogram of the SVF distribution of the original 40,000 images. (b) shows the same histogram but for the augmented dataset of 130,000 images. c()



**/ig. C.12.** Evolution of *𝑘̄* eff,norm as a function of *𝑘̄* s.

choices aimed at achieving precise solutions within a practical time frame. Hence those were fixed values for all simulations and com- parisons to the theoretical model. Thus, to increase the accuracy, we recommend simply increasing the maximum number of iterations and making the convergence criteria more strict. Through testing, we have observed that this adjustment significantly decrease the error versus the theoretical model at the cost of computational time.

# Appendix C. *𝒌*eff Normalization issues

The normalized form of *𝑘̄* eff,norm, from Eq. ([15](#_bookmark42)), is a convenient way

of expressing the effective thermal conductivity. For a given porosity,

1. Liu X, Song F, Xu Q, Luo Q, Tian Y, Li J, et al. The influence of pore size distribution on thermal conductivity, permeability, and phase change behavior of hierarchical porous materials. Sci China Technol Sci 2021;64(11):2485–94. <http://dx.doi.org/10.1007/s11431-021-1813-0>.
2. Tian S, Ren W, Li G, Yang R, Wang T. A theoretical analysis of pore size distribution effects on shale apparent permeability. Geofluids 2017;2017. [http:](http://dx.doi.org/10.1155/2017/7492328)

[//dx.doi.org/10.1155/2017/7492328](http://dx.doi.org/10.1155/2017/7492328).

1. Bhattacharjee B. Influence of pore size distribution on the properties of a stabilized soil cement system. In: Geo-China 2016. American Society of Civil Engineers; 2016, p. 53–60. <http://dx.doi.org/10.1061/9780784480069.007>, URL <https://ascelibrary.org/doi/10.1061/9780784480069.007>.
2. Ranut P. On the effective thermal conductivity of aluminum metal foams: Review and improvement of the available empirical and analytical models. Appl Therm Eng 2016;101:496–524. [http://dx.doi.org/10.1016/j.applthermaleng.](http://dx.doi.org/10.1016/j.applthermaleng.2015.09.094) [2015.09.094](http://dx.doi.org/10.1016/j.applthermaleng.2015.09.094).
3. Wang F, Li X. The stagnant thermal conductivity of porous media predicted by the random walk theory. Int J Heat Mass Transfer 2017;107:520–33. [http:](http://dx.doi.org/10.1016/j.ijheatmasstransfer.2016.11.069)

[//dx.doi.org/10.1016/j.ijheatmasstransfer.2016.11.069](http://dx.doi.org/10.1016/j.ijheatmasstransfer.2016.11.069).

1. Wang F, Li X, Tan J, Hao X, Xiong B. Pore-scale prediction of the oxygen effective diffusivity in porous battery electrodes using the random walk theory. Int J Heat Mass Transfer 2022;183:122085. [http://dx.doi.org/10.1016/j.ijheatmasstransfer.](http://dx.doi.org/10.1016/j.ijheatmasstransfer.2021.122085) [2021.122085](http://dx.doi.org/10.1016/j.ijheatmasstransfer.2021.122085).
2. [Fukushima K. Biological cybernetics neocognitron: A self-organizing neural](http://refhub.elsevier.com/S2666-5468(23)00082-4/sb10) [network model for a mechanism of pattern recognition unaffected by shift in](http://refhub.elsevier.com/S2666-5468(23)00082-4/sb10) [position. Biol Cybern 1980;36:202.](http://refhub.elsevier.com/S2666-5468(23)00082-4/sb10)
3. Minaee S, Boykov YY, Porikli F, Plaza AJ, Kehtarnavaz N, Terzopoulos D. Image segmentation using deep learning: A survey. IEEE Trans Pattern Anal Mach Intell 2021. <http://dx.doi.org/10.1109/TPAMI.2021.3059968>, [arXiv:2001.05566](http://arxiv.org/abs/2001.05566).
4. Yang Z, Yabansu YC, Al-Bahrani R, keng Liao W, Choudhary AN, Ka- lidindi SR, et al. Deep learning approaches for mining structure-property linkages in high contrast composites from simulation datasets. Comput Mater Sci 2018;151(April):278–87. <http://dx.doi.org/10.1016/j.commatsci.2018.05.014>.
5. Cecen A, Dai H, Yabansu YC, Kalidindi SR, Song L. Material structure-property linkages using three-dimensional convolutional neural networks. Acta Mater 2018;146:76–84. <http://dx.doi.org/10.1016/j.actamat.2017.11.053>.

*𝑘̄*

eff,norm

is a measure between 0 and 1, respectively representing the

1. Wu H, Fang WZ, Kang Q, Tao WQ, Qiao R. Predicting effective diffusivity

of porous media from images by deep learning. Sci Rep 2019;9(1):1–12.

However, as we will shown in this section, the variable *𝑘̄* eff,norm is still maximum and minimum theoretical effective thermal conductivity. a function of *𝑘̄* s, and no equations currently exist to fit the curves of

*𝑘̄* eff,norm as a function of *𝑘̄* s and *𝜖* for a given geometry.

For this study, consider the geometries presented shown in [Fig.](#_bookmark4) [1](#_bookmark4).

of *𝑘̄* s vary between *𝑘̄* s = 1*.*05 and 105. For each geometry and *𝑘̄* s, The porosities are 0.20, 0.75, 0.38, and 0.33 respectively. The values

the maximum and minimum are calculated according to Eqs. ([11](#_bookmark18)) and

([12](#_bookmark19)) respectively. Finally, the value of *𝑘̄* eff is calculated using the NHT

algorithm outlined in Section [2](#_bookmark5). The results are shown in [Fig.](#_bookmark48) [C.12](#_bookmark48).

[Figs.](#_bookmark4) [1](#_bookmark4)c and [1](#_bookmark4)d have similar porosities, yet, as seen in [C.12](#_bookmark48), it is not

Curve (b) peaks around *𝑘̄* s = 2, curve (c) peaks around *𝑘̄* s = 5, while obvious how the curves behave and what variables best fit the curve. curves (a) and (d) peak around *𝑘̄* s = 10. Curve (b), as *𝑘̄* s increases,

converges to 0, while all other curves converge to their own number.

In addition, curve (b) peaks the highest, and converges to the lowest value out of the ones presented.

Currently, there is no research discussing what variables define these curves other than porosity and the ratio of thermal conductivities.

Having the ability to infer *𝑘̄* eff for any *𝑘̄* s for a given geometry from only

one simulation would also allow for one CNN to predict *𝑘̄* eff,norm for all

combinations of thermal conductivity ratios. That is currently not the

<http://dx.doi.org/10.1038/s41598-019-56309-x>, URL [https://www.nature.com/](https://www.nature.com/articles/s41598-019-56309-x)

[articles/s41598-019-56309-x](https://www.nature.com/articles/s41598-019-56309-x).

1. Krizhevsky A, Sutskever I, Hinton GE. ImageNet classification with deep con- volutional neural networks. Commun ACM 2017;60(6):84–90. [http://dx.doi.org/](http://dx.doi.org/10.1145/3065386) [10.1145/3065386](http://dx.doi.org/10.1145/3065386), URL <http://code.google.com/p/cuda-convnet/>.
2. He K, Zhang X, Ren S, Sun J. Deep residual learning for image recognition. In: Proceedings of the IEEE computer society conference on computer vision and pattern recognition, vol. 2016-Decem. IEEE Computer Society; 2016, p. 770–8. <http://dx.doi.org/10.1109/CVPR.2016.90>, [arXiv:1512.03385](http://arxiv.org/abs/1512.03385).
3. Wei H, Zhao S, Rong Q, Bao H. Predicting the effective thermal conductivities of composite materials and porous media by machine learning methods. Int J Heat Mass Transfer 2018;127:908–16. [http://dx.doi.org/10.1016/j.ijheatmasstransfer.](http://dx.doi.org/10.1016/j.ijheatmasstransfer.2018.08.082) [2018.08.082](http://dx.doi.org/10.1016/j.ijheatmasstransfer.2018.08.082).
4. Zhu MX, Song HG, Yu QC, Chen JM, Zhang HY. Machine-learning-driven discovery of polymers molecular structures with high thermal conductivity. Int J Heat Mass Transfer 2020;162. [http://dx.doi.org/10.1016/j.ijheatmasstransfer.](http://dx.doi.org/10.1016/j.ijheatmasstransfer.2020.120381) [2020.120381](http://dx.doi.org/10.1016/j.ijheatmasstransfer.2020.120381).
5. Wei H, Bao H, Ruan X. Perspective: Predicting and optimizing thermal transport properties with machine learning methods. Energy AI 2022;8. [http://dx.doi.org/](http://dx.doi.org/10.1016/j.egyai.2022.100153) [10.1016/j.egyai.2022.100153](http://dx.doi.org/10.1016/j.egyai.2022.100153).
6. Simonyan K, Zisserman A. Very deep convolutional networks for large-scale image recognition. In: 3rd international conference on learning representations, ICLR 2015 - conference track proceedings. 2015, p. 1–14, [arXiv:1409.1556](http://arxiv.org/abs/1409.1556).
7. Bendsøe MP. Optimization of structural topology, shape, and material. Berlin Heidelberg: Springer; 1995, <http://dx.doi.org/10.1007/978-3-662-03115-5>.

case, which is a major reason why we define *𝑘̄* four different ratios of *𝑘̄* s.

# References

eff

and train the CNN for

1. Sevart CD, Bergman TL. An itertive design method to reduce the overall thermal resistance in a conjugate conduction-free convection configuration. Front Heat Mass Transf 2019;13. <http://dx.doi.org/10.5098/hmt.13.18>.
2. Sevart CD, Bergman TL. Evolutionary design method for a conducting solid cooled by combined free convection and radiation. J Heat Transfer 2021;143(4):1–9. <http://dx.doi.org/10.1115/1.4049841>.
3. Wu X, Zhu Y. Heterogeneous materials: A new class of materials with unprece- dented mechanical properties. Mater Res Lett 2017;5(8):527–32. [http://dx.doi.](http://dx.doi.org/10.1080/21663831.2017.1343208) [org/10.1080/21663831.2017.1343208](http://dx.doi.org/10.1080/21663831.2017.1343208).
4. Lin G, Liu S, Qu G, Song Y, Li T, Liu F, et al. Effect of pore size distribution in the gas diffusion layer adjusted by composite carbon black on fuel cell performance. Int J Energy Res 2021;45(5):7689–702. <http://dx.doi.org/10.1002/er.6350>, URL <https://onlinelibrary.wiley.com/doi/full/10.1002/er.6350>.
5. Cui CL, Schweich D, Villermaux J. Influence of pore diameter distribution on the determination of effective diffusivity in porous particles. Chem Eng Process 1989;26(2):121–6. <http://dx.doi.org/10.1016/0255-2701(89)90004-4>.
6. Ding R, Zhang S, Chen Y, Rui Z, Hua K, Wu Y, et al. Application of machine learning in optimizing proton exchange membrane fuel cells: A review. Energy AI 2022;9:100170. <http://dx.doi.org/10.1016/j.egyai.2022.100170>.
7. Adam A, Li X, Fang H. 2D binary images and effective thermal conductivity CFD results. Mendeley; 2023, <http://dx.doi.org/10.17632/454DSRMDYF.1>, URL <https://data.mendeley.com/datasets/454dsrmdyf/1>.
8. Bargmann S, Klusemann B, Markmann J, Schnabel JE, Schneider K, Soyarslan C, et al. Generation of 3D representative volume elements for heterogeneous materials: A review. Prog Mater Sci 2018;96:322–84. [http://dx.doi.org/10.1016/](http://dx.doi.org/10.1016/j.pmatsci.2018.02.003) [j.pmatsci.2018.02.003](http://dx.doi.org/10.1016/j.pmatsci.2018.02.003).
9. Bostanabad R, Zhang Y, Li X, Kearney T, Brinson LC, Apley DW, et al. Computational microstructure characterization and reconstruction: Review of the state-of-the-art techniques. Prog Mater Sci 2018;95:1–41. [http://dx.doi.org/10.](http://dx.doi.org/10.1016/j.pmatsci.2018.01.005) [1016/j.pmatsci.2018.01.005](http://dx.doi.org/10.1016/j.pmatsci.2018.01.005).
10. Ju Y, Huang Y, Zheng J, Qian X, Xie H, Zhao X. Multi-thread parallel algorithm for reconstructing 3D large-scale porous structures. Comput Geosci 2017;101(June 2016):10–20. <http://dx.doi.org/10.1016/j.cageo.2017.01.003>.
11. Adam A, Wang F, Li X. Efficient reconstruction and validation of heterogeneous microstructures for energy applications. Int J Energy Res 2022. [http://dx.doi.](http://dx.doi.org/10.1002/er.8578) [org/10.1002/er.8578](http://dx.doi.org/10.1002/er.8578), URL [https://onlinelibrary.wiley.com/doi/full/10.1002/er.](https://onlinelibrary.wiley.com/doi/full/10.1002/er.8578) [8578](https://onlinelibrary.wiley.com/doi/full/10.1002/er.8578).
12. Politis MG, Kikkinides ES, Kainourgiakis ME, Stubos AK. A hybrid process- based and stochastic reconstruction method of porous media. Microporous Mesoporous Mater 2008;110(1):92–9. [http://dx.doi.org/10.1016/J.MICROMESO.](http://dx.doi.org/10.1016/J.MICROMESO.2007.09.024) [2007.09.024](http://dx.doi.org/10.1016/J.MICROMESO.2007.09.024).
13. Wang M, Wang J, Pan N, Chen S. Mesoscopic predictions of the effective thermal conductivity for microscale random porous media. Phys Rev E 2007;75(3). <http://dx.doi.org/10.1103/PhysRevE.75.036702>, URL [https://escholarship.org/](https://escholarship.org/uc/item/7891j8jt) [uc/item/7891j8jt](https://escholarship.org/uc/item/7891j8jt).
14. Ke X, Duan Y. A spatially-varying relaxation parameter Lattice Boltzmann Method (SVRP-LBM) for predicting the effective thermal conductivity of composite material. Comput Mater Sci 2019;169. [http://dx.doi.org/10.1016/j.commatsci.](http://dx.doi.org/10.1016/j.commatsci.2019.109080) [2019.109080](http://dx.doi.org/10.1016/j.commatsci.2019.109080).
15. Zhu W, Kan A, Chen Z, Zhang Q, Zhang J. A modified Lattice Boltzmann method for predicting the effective thermal conductivity of open-cell foam materials. Int Commun Heat Mass Transfer 2022;133:105957. [http://dx.doi.org/10.1016/j.](http://dx.doi.org/10.1016/j.icheatmasstransfer.2022.105957) [icheatmasstransfer.2022.105957](http://dx.doi.org/10.1016/j.icheatmasstransfer.2022.105957).
16. Yang L, Shami A. On hyperparameter optimization of machine learning algo- rithms: Theory and practice. Neurocomputing 2020;415:295–316. [http://dx.doi.](http://dx.doi.org/10.1016/j.neucom.2020.07.061) [org/10.1016/j.neucom.2020.07.061](http://dx.doi.org/10.1016/j.neucom.2020.07.061), [arXiv:2007.15745](http://arxiv.org/abs/2007.15745).
17. Goodfellow IJ, Bulatov Y, Ibarz J, Arnoud S, Shet V. Multi-digit number recog- nition from street view imagery using deep convolutional neural networks. In: 2nd international conference on learning representations, ICLR 2014 - conference track proceedings. International Conference on Learning Representations, ICLR; 2014, <http://dx.doi.org/10.48550/arxiv.1312.6082>, [arXiv:1312.6082](http://arxiv.org/abs/1312.6082), URL [https:](https://arxiv.org/abs/1312.6082v4)

[//arxiv.org/abs/1312.6082v4](https://arxiv.org/abs/1312.6082v4).

1. Li X, Zhang Y, Zhao H, Burkhart C, Brinson LC, Chen W. A transfer learning approach for microstructure reconstruction and structure-property predictions. Sci Rep 2018;8(1):1–13. <http://dx.doi.org/10.1038/s41598-018-31571-7>, [arXiv:](http://arxiv.org/abs/1805.02784)

[1805.02784](http://arxiv.org/abs/1805.02784), URL <http://dx.doi.org/10.1038/s41598-018-31571-7>.

1. Smith SL, Kindermans PJ, Ying C, Le QV. Don’t decay the learning rate, increase the batch size. In: 6th international conference on learning representations, ICLR 2018 - conference track proceedings. 2018, [arXiv:1711.00489](http://arxiv.org/abs/1711.00489).
2. Kingma DP, Ba JL. Adam: A method for stochastic optimization. In: 3rd international conference on learning representations, ICLR 2015 - conference track proceedings. 2015, [arXiv:1412.6980](http://arxiv.org/abs/1412.6980).
3. Kirkpatrick S, Gelatt CD, Vecchi MP. Optimization by simulated annealing. Science 1983;220(4598):671–80. [http://dx.doi.org/10.1126/science.220.4598.](http://dx.doi.org/10.1126/science.220.4598.671)

[671](http://dx.doi.org/10.1126/science.220.4598.671).

1. [Xie Y, Steven G. A simple approach to structural optimization. Compurers Struct](http://refhub.elsevier.com/S2666-5468(23)00082-4/sb40) [1993;49(5):885–96.](http://refhub.elsevier.com/S2666-5468(23)00082-4/sb40)