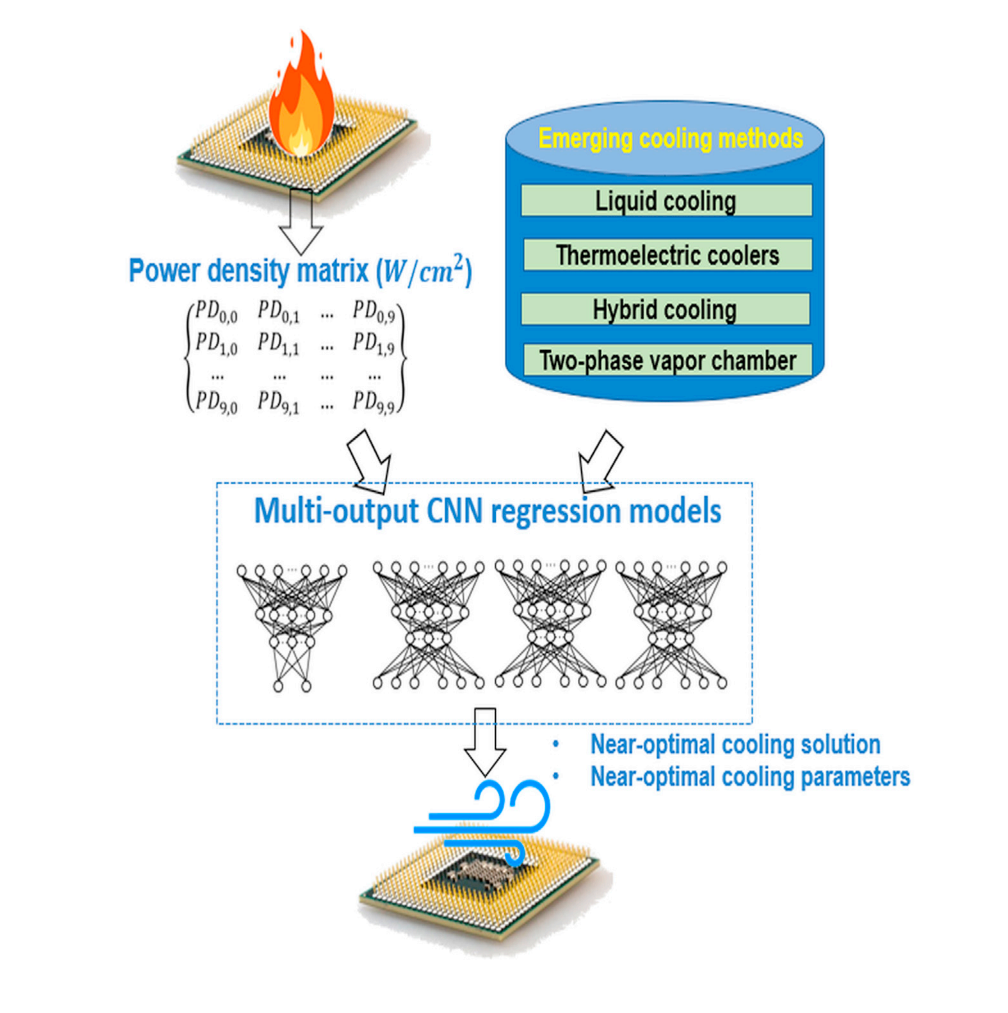
iScience

CellPress OPEN ACCESS

Article

Neural network-based cooling design for high-performance processors


Power density matrix (W/cm^2)
$\begin{pmatrix} PD_{0,0} & PD_{0,1} & \dots & PD_{0,9} \\ PD_{1,0} & PD_{1,1} & \dots & PD_{1,9} \\ \dots & \dots & \dots & \dots \\ PD_{9,0} & PD_{9,1} & \dots & PD_{9,9} \end{pmatrix}$
Emerging cooling methods
Liquid cooling
Thermoelectric coolers
Hybrid cooling
Two-phase vapor chamber
Multi-output CNN regression models
Near-optimal cooling solution
Near-optimal cooling parameters

Zihao Yuan, Ayse K. Coskun

yuan1z@bu.edu

Highlights

The paper introduces a Deep Learning-based chip cooling design optimization flow

The proposed flow is built using multi-output convolutional neural networks

The proposed flow is fast and accurate for selecting the optimal cooling design

The proposed flow is modular for various chip architectures and cooling designs

Yuan & Coskun, iScience *25*, 103582
January 21, 2022 © 2021 The Author(s).
https://doi.org/10.1016/j.isci.2021.103582

iScience

CellPress
OPEN ACCESS

Article

Neural network-based cooling design for high-performance processors

Zihao Yuan[1,2,*] and Ayse K. Coskun[1]

SUMMARY

Ultra-high chip power densities that are expected to surpass 1-2kW/cm^2 in future high-performance systems cannot be easily handled by conventional cooling methods. Various emerging cooling methods, such as liquid cooling via microchannels, thermoelectric coolers (TECs), two-phase vapor chambers, and hybrid cooling options have been designed to efficiently remove heat from high-performance processors. However, selecting the optimal cooling solution for a given chip and determining the optimal cooling parameters for that solution to achieve high efficiency are open problems. These problems are, in fact, computationally expensive because of the massive space of possible solutions. To address this design challenge, this article introduces a deep learning-based cooling design optimization flow that rapidly and accurately converges to the optimal cooling solution as well as the optimal cooling parameters for a given chip floorplan and its power profile.

INTRODUCTION

Over the last few decades, on-chip power densities have grown tremendously following the downscaling of transistors. Power densities that reach 1-2kW/cm^2 caused by the performance boost of scaling already occur in high-performance chips and result in amplified localized hot spots (Schultz et al., 2016). These localized on-chip hot spots not only degrade the performance of the chip, but also generate larger subthreshold leakage power and cause reliability challenges (Srinivasan et al., 2004; Kim et al., 2003). Conventional on-chip cooling solutions such as forced air cooling via fans or pin-fin heat sink are often not sufficient to mitigate such high-power-density hot spots and can result in over/under-cooling. Emerging cooling technologies such as liquid cooling via microchannels (Dang et al., 2010), thermoelectric coolers (TECs) (Chowdhury et al., 2009), two-phase vapor chambers (VCs) (Bulut et al., 2019), and hybrid cooling options (Yazawa et al., 2012) (e.g., of liquid cooling via microchannels and TECs) have the potential to provide better cooling performance compared to the conventional cooling solutions. However, there is no obvious winner in terms of cooling efficiency among all these emerging cooling technologies. The cooling performance and cooling power of these potential solutions vary significantly based on the cooling parameters (such as liquid flow velocity, evaporator design, TEC current, etc.) (Yuan et al., 2019a; 2019b). The selection of the cooling technologies and the cooling parameters also needs to consider the chip architecture, chip size, and floorplan, as well as the power profiles of the applications running on the given chip. To minimize the cooling power while satisfying chip thermal constraints, there is a need for an optimization flow that enables rapid and accurate selection of the optimal cooling solution and the associated cooling parameters for a given chip and application profile.

A key enabler to such a cooling design optimization flow is a set of accurate and fast models for various cooling technologies. A common approach toward this direction is using compact thermal models (CTMs) that model heat dissipation with an equivalent lumped circuit model (Pedram and Nazarian, 2006). However, given the vast solution space of possible cooling solutions (including possible hybrids) and cooling parameters, the optimal solution search time is still prohibitively time-consuming with CTMs (Yuan et al., 2019b). In addition to cooling design choice possibilities, the optimization flow needs to also account for the chip design and power profile changes. In this case, using a simple grid search to find the optimal cooling design for a small-sized chip floorplan and its typical power profile could take up to days (Yuan et al., 2020). Previous work has investigated using machine learning or black-box optimization methods to optimize or model the system with emerging cooling technologies such as liquid cooling via microchannels and TECs (Beneventi et al., 2012; Fan et al., 2018; Blackburn et al., 2020; Zhou et al.,

[1]Electrical and Computer Engineering Department from Boston University, Boston, MA 02148, USA

[2]Lead contact

*Correspondence: yuan1z@bu.edu

https://doi.org/10.1016/j.isci.2021.103582

2020; Tang et al., 2020). However, their models or techniques have not considered a wide range of emerging cooling technologies or selected the optimal cooling solution and cooling parameters based on cooling efficiency (Chen et al., 2020; Sheng et al., 2020; Juneia et al., 2019; Ramos-Alvarado et al., 2013).

This paper argues that applying a deep learning (DL) model provides an effective solution to the above challenge. A DL regression model can learn the intrinsic information among the chip designs and the cooling solutions, and then generate the optimal cooling solution as well as the cooling parameters, given a specific chip floorplan and power profile. The paper demonstrates a step toward this goal by designing a multi-output convolutional neural network (CNN) regression model to estimate the best cooling method and its cooling design and technology parameters. The cost function used to evaluate the output of the CNN is the combination of cooling power, hot spot temperatures, and temperature constraint of the chip. This CNN-based optimization flow requires the CNN regression model to be sufficiently modular for all chip floorplans and power profiles. If the input chip floorplan and power profile change, the predicted cooling solution as well as the cooling parameters should still maintain at the desired accuracy. We experiment with realistic multiprocessor system-on-chip (MPSoCs) data to evaluate the search time as well as the search accuracy of this CNN-based cooling design flow. Results confirm that, when compared to existing optimization methods, our proposed CNN architectures and DL-based optimization flow can successfully predict the optimal cooling solution and cooling parameters with a maximum error of less than 4% and a maximum speedup of 140X.

Emerging cooling technologies

Several new cooling methods have been developed to handle the high heat fluxes and address the inefficiency problem of conventional solutions. These new methods involve careful engineering of new technologies and advanced materials to reduce hot spots and thermal gradients on the chip, both of which impact power, reliability, and overall performance. We next briefly discuss several promising emerging cooling solutions.

Liquid cooling via microchannels

Liquid cooling via microchannels as shown in Figure 1A is an attractive cooling solution that uses the liquid convection effect to remove heat from processors (Sridhar et al., 2013; Dang et al., 2010). There are two main contributors to the convective heat transfer of the coolant: (1) convective heat transfer from the walls of the channel to the liquid and (2) convective heat transfer in the direction of the liquid flow into and out of the current liquid cell (Sridhar et al., 2013). These contributors, and the overall heat transfer capability of the system, are strongly impacted by the design geometries, material choices, and the active cooling power (that powers the pump connected to the system).

Thermoelectric coolers (TECs)

TEC units have gained attraction because of their abilities to effectively remove heat from high power density hot spots. A TEC unit operates based on the Peltier effect such that when an electric current passes through a TEC unit, heat is absorbed from one side (cold side) and rejected on the other side (hot side). TEC units are typically placed directly above hot spots. Existing on-chip TEC devices are composed of ultrathin (5–10 μm) Bi_2Te_3-based p-n thermocouples sandwiched between copper mini-headers and are covered with ceramic plates at the outermost surfaces to provide insulation. A typical chip stack of the TEC device is shown in Figure 1B (Kaplan et al., 2017).

Two-phase vapor chambers (VCs)

Two-phase cooling using VCs is a passive cooling method that uses a capillary-driven flow that conducts thin-film evaporation through a porous wick placed on the bottom surface of the VC to remove heat from processors (Vaartstra et al., 2019). The schematic of a VC is shown in Figure 1C. The advantages of two-phase VCs are better cooling performance and no pumping power (in contrast to liquid cooling via microchannels and microchannel-based two-phase cooling). There are two metrics that impact the cooling performance of the VCs: (i) heat transfer coefficient (HTC), and (ii) dry-out heat flux. HTC is a parameter that determines the rate of heat transfer per unit temperature difference of the evaporator. Dry-out heat flux is the thermal limit of a two-phase device. If the hot spot power density of the chip is higher than the dry-out heat flux, the coolant will no longer remain in the two-phase state and possibly cause overheating and damage to the chip (Vaartstra et al., 2019; Yuan et al., 2019a; 2019b). Therefore, a higher dry-out heat flux means

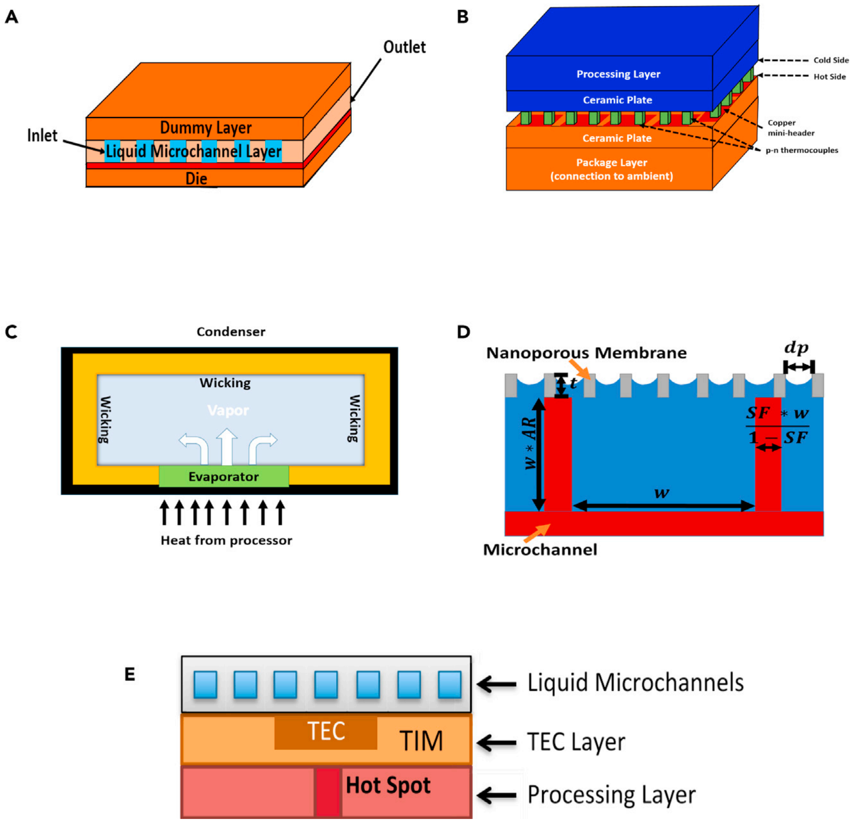


Figure 1. Emerging cooling method structure view

(A) A simple liquid cooling via microchannels chip stack.

(B) A typical chip stack of the TEC device.

(C) A vapor chamber structure view.

(D) Hybrid wick evaporator front view.

(E) A hybrid cooling method of liquid cooling via microchannels and TECs.

more heat can be removed before the coolant dries out. Micropillar wick evaporator is one type of evaporator that is particularly interesting because it uses thin-film evaporation to provide high HTC (Vaartstra. et al., 2019; Yuan et al., 2019a; 2019b). The HTC and dry-out heat flux of the micropillar wick evaporator are mainly determined by the micropillar geometry (height, diameter, and pitch).

For a micropillar wick evaporator, an evaporator with a higher HTC is desired to reduce the thermal resistance of the VCs. However, such high HTC evaporators often suffer from low critical dry-out heat flux. These two metrics are typically conflicting with each other and it is challenging to maximize HTC while enhancing dry-out heat flux. A hybrid wick evaporator of nanoporous membrane and microchannels, as shown in Figure 1D, has the potential to simultaneously improve both HTC and dry-out heat flux. Similar to two-phase VCs with micropillar wick evaporators, two-phase VCs with hybrid wick evaporators only replace the micropillar evaporator wicking structure with hybrid wicking structure of nanoporous membrane and microchannels (Lu et al., 2019; Yuan et al., 2020). The hybrid wick evaporator geometry list in Table 1 determines the HTC and dry-out heat flux. The microchannel and membrane geometries can be varied independently so as to enhance the permeability of the microchannels and the heat transfer from highly conductive solids (the substrate, microchannels, and nanoporous membrane) to the liquid-vapor interface.

Hybrid cooling

Hybrid cooling refers to incorporating two or more cooling solutions on the same platform. For example, as shown in Figure 1E, a hybrid cooling system can be designed using liquid cooling via microchannels and TECs (Kaplan et al., 2017; Yazawa et al., 2012). Liquid cooling via microchannels can effectively remove

Table 1. Hybrid wick geometry parameters and valid range

Symbol	Parameters	Valid range
t	Nanoporous membrane thickness	250–1000nm
dp	Membrane pore diameter	50–200nm
φ	Membrane porosity	0.2–0.8
AR	Microchannel aspect ratio	0.5–2
SF	Microchannel wall solid fraction	0.1–0.4
w	Microchannel width	2–8 μm

the background heat in large chips and especially in 3D-stacked architectures. TEC is favorable for handling high power densities in a small area. Hybrid cooling combines the advantages of both cooling methods and can potentially provide high cooling efficiency by manipulating the TEC current and liquid flow velocity. In hybrid cooling, TEC would be placed on select hot spots to shave the high temperatures, and then liquid cooling would remove the overall heat.

Compact thermal modeling methodology

Commercial multiphysics simulators, such as COMSOL Multiphysics (Pryor, 2009) and ANSYS (Stolarski et al., 2018), are typically used to design and simulate the thermal models of the aforementioned emerging cooling solutions. However, these tools require significant efforts to construct system-specific models. Such tools also incur long simulation times as well as large memory requirements (e.g., simulating an *mm*-scale chip takes from hours to multiple days and easily requires tens of GBs of memory) (Yuan et al., 2019a; 2019b; Kaplan et al., 2017). Compact thermal modeling has been designed to tackle the long design and simulation time problem (Pedram and Nazarian, 2006; Skadron et al., 2003). In the following sections, we briefly discuss the compact modeling methodologies and CTMs for each emerging cooling technology.

A CTM leverages the duality between electrical and thermal properties. To model a single heat source chip, the traditional heat equation can be transformed into a first-order resistor and capacitor (RC) circuit. To model a multiple heat source chip, this electrical and thermal duality can simplify the heat conductions from the neighboring nodes as a first-order thermal RC matrix equation as shown in Equation 1:

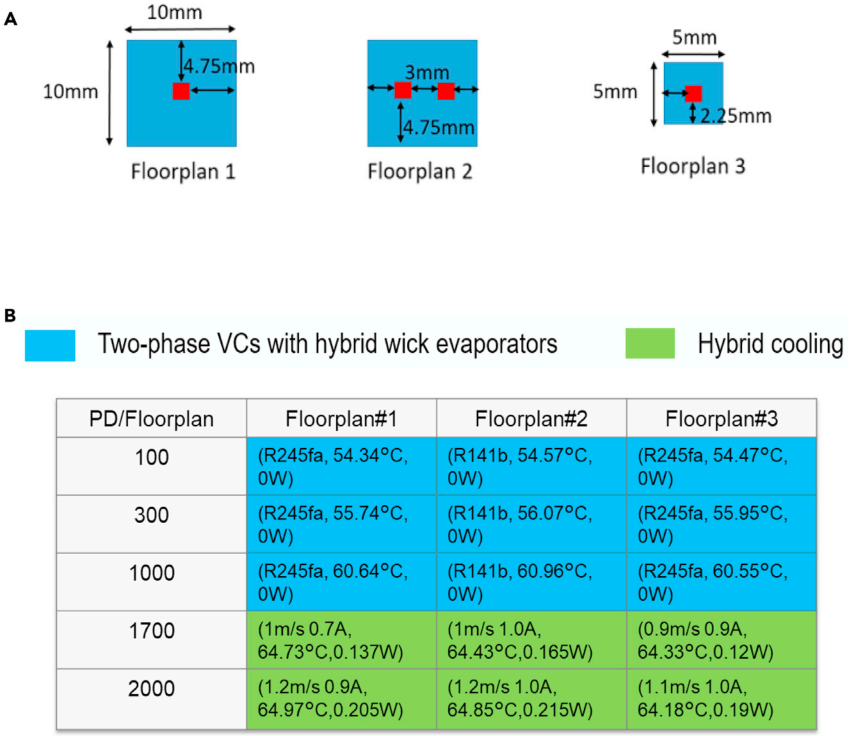
$$GT(t) + C\dot{T}(t) = U(t), \qquad \text{(Equation 1)}$$

where G, C, U are the equivalent thermal conductance, thermal capacitance, and power dissipation matrices, respectively. T is the node temperature matrix that can be solved from Equation 1. This thermal RC equation matrix can be solved using differential solvers, such as LU decomposition solvers, to obtain the steady-state temperature of each thermal node (Skadron et al., 2003; Yuan et al., 2021). Solving this first-order RC matrix is apparently faster than solving the second-order heat equations (Pedram and Nazarian, 2006).

Researchers have developed various CTMs to model the cooling behaviors of emerging cooling solutions such as liquid cooling via microchannels (Sridhar et al., 2013; Kaplan et al., 2017), TECs (Kaplan et al., 2017; Yazawa et al., 2012), two-phase VCs with micropillar wick evaporators (Yuan et al., 2019a; 2019b), two-phase VCs with hybrid wick evaporators (Yuan et al., 2020), and hybrid cooling (of liquid cooling via microchannels and TECs) (Kaplan et al., 2017; Yazawa et al., 2012).

Tradeoff of emerging cooling methods

As we discussed in the previous sections, researchers have developed CTMs for various emerging cooling technologies and among these cooling technologies, liquid cooling via microchannels, two-phase VCs, and hybrid cooling (of liquid cooling via microchannels and TECs) are particularly interesting because of their reported high heat transfer rate compared to conventional cooling methods. However, there is no comprehensive study to compare the cooling efficiency in terms of cooling performance and cooling power for these cooling technologies. We next study the cooling efficiency of the cooling methods using their corresponding CTMs. We adopt the optimization objective function as shown in Equation 2 from previous work (Yuan et al., 2019b):



🟦 Two-phase VCs with hybrid wick evaporators 🟩 Hybrid cooling

PD/Floorplan	Floorplan#1	Floorplan#2	Floorplan#3
100	(R245fa, 54.34°C, 0W)	(R141b, 54.57°C, 0W)	(R245fa, 54.47°C, 0W)
300	(R245fa, 55.74°C, 0W)	(R141b, 56.07°C, 0W)	(R245fa, 55.95°C, 0W)
1000	(R245fa, 60.64°C, 0W)	(R141b, 60.96°C, 0W)	(R245fa, 60.55°C, 0W)
1700	(1m/s 0.7A, 64.73°C,0.137W)	(1m/s 1.0A, 64.43°C,0.165W)	(0.9m/s 0.9A, 64.33°C,0.12W)
2000	(1.2m/s 0.9A, 64.97°C,0.205W)	(1.2m/s 1.0A, 64.85°C,0.215W)	(1.1m/s 1.0A, 64.18°C,0.19W)

Figure 2. Initial optimization results for hybrid cooling and two-phase VCs with hybrid wick evaporators

(A) Synthetic chip floor plans.

(B) Results for on-chip temperature constraint = 65°C. The format for two-phase VCs with hybrid wick evaporators is {coolant, hot spot temperature, cooling power}. The format for hybrid cooling is {liquid flow velocity, TEC current, hot spot temperature, cooling power}.

$$\min \alpha P_{\text{cooling,norm}} + \beta(\max(T_{\text{hs}} - T_{\text{limit}}, 0)_{,\text{norm}}). \qquad \text{(Equation 2)}$$

$P_{cooling,norm}$ is the normalized cooling power and $T_{hs} - T_{limit}$ is the temperature difference between the hot spot temperature and on-chip temperature limit. α is the user-specific weight factor with no unit and β is the penalty weight that we set to a large value to prevent violation of the temperature constraint. Given the chip floorplans, power profile, and on-chip temperature limit, we want to find out the optimal cooling solution along with the cooling parameters that result in the minimum cooling power while satisfying the temperature constraint. We select 3 floorplans with a various number of hot spots and hot spot power densities as shown in Figure 2A.

For each of the floorplan, we set the background power density to 50 W/cm^2 with hot spot power densities of $\{100, 300, 1000, 1700, 2000\} W/cm^2$. We use the aforementioned emerging cooling methods as heat sinks or inter-layer cooling methods. The on-chip maximum temperature limit (temperature constraint) is set to 65 °C. The detailed experimental setup can be found in the previous work (Yuan et al., 2019b, 2020). To select the optimal cooling solutions and cooling parameters, we use the covariance matrix adaptation evolution strategy (CMA-ES) and multi-start simulated annealing (MSA) to select the optimal cooling parameters for the aforementioned emerging cooling technologies (Yuan et al., 2019b, 2020). We summarize the results in Figure 2B.

As shown in Figure 2B, because two-phase VCs are passive cooling methods (no additional power is needed on the evaporator side), for relatively low power density (100, 300, and 1000 W/cm^2), two-phase VCs with hybrid wick evaporators completely beat other cooling methods. Note that, compared to the hybrid wick, the micropillar wick evaporator cannot provide enough HTC on the evaporator side to cool down the chip because of the low dry-out limit. Liquid cooling via microchannels cannot provide enough power to remove the high heat flow generated by the chip. Because hybrid cooling has finer control over the cooling power and cooling ability, hybrid cooling always results in lower cooling power compared to only using TEC. For high power density (1700 and 2000 W/cm^2), because hybrid cooling and TECs are targeted at removing the hot spot heat, hybrid cooling is the optimal cooling method in these cases. We also carry out experiments with a temperature constraint of 90°C, but because two-phase VCs with hybrid wick evaporators are passive cooling methods, they beat other emerging cooling technologies in terms of cooling efficiency. Because two-phase VCs with hybrid wick evaporators and hybrid cooling (of liquid cooling via microchannels and TECs) can achieve the optimal cooling efficiency among all the aforementioned cooling technologies. Therefore, we only discuss the CNN regression architectures for these two cooling technologies. We will also perform cooling optimization studies using realistic high power density chips in the optimization results section.

RESULTS

In this section, we first demonstrate the proposed DL-based cooling optimization flow and then discuss the validation results of the proposed CNN architectures. CNN architectures are discussed detail in the Star Methods. Next, we demonstrate the efficiency of using our proposed DL-based cooling optimization flow against existing cooling optimization methods on realistic multiprocessor system-on-chips from OpenROAD (Ajayi et al., 2019) and IBM Power9 processor (Sadasivam et al., 2017).

Overall CNN optimization architecture

The existing cooling optimization methods (CMA-ES and MSA) have two main issues: (i) need to run a great number of thermal simulations which results in large simulation time and (ii) there is no guarantee that the selected cooling method and its cooling parameters are optimal. The accuracy of the optimization result selected by CMA-ES and MSA is determined by the sample size and the number of iterations (Yuan et al., 2019b, 2020). Using the DL model, specifically, the CNN regression model, to predict the optimal cooling solution and its cooling parameters could be the solution to these two issues. In this section, we will elaborate on the proposed DL-based cooling optimization flow.

The overall CNN optimization architecture is shown in Figure 3A. Given an arbitrary chip power map, the optimization flow standardizes the power map into a 10 × 10 power density matrix. The power density matrix is used as the input to the hybrid cooling and two-phase VCs CNN architectures to predict the optimal cooling parameters for these two cooling technologies, respectively. The optimization flow then conducts thermal simulations for the input power map using hybrid cooling and two-phase VCs with hybrid wick evaporators as the cooling method and compares the hot spot temperatures and the cooling costs to determine the optimal cooling method and its cooling parameters.

As the solution space of this cooling optimization problem is continuous instead of discrete, the optimization results found by black-box optimization methods (e.g., exhaustive search, simulated annealing, or others) can only be near-optimal because they require discretizing the optimization inputs. Therefore, unless an accurate mathematical formula is created for such an optimization problem, the accuracy of the optimization methods will always depend on the input granularity and the outputs can only be near-optimal. It is not possible to create an accurate mathematical formula for this particular cooling optimization problem to solve it analytically; thus, we consider the output of our proposed DL-based optimization framework as optimal given the constraints. The accuracy of the proposed optimization flow depends on the granularity of the training data in the cooling parameter solution space.

Validation of the proposed CNN architectures

To validate the accuracy of the CNN architectures discussed in the previous section, we divide the 90,000 training power density maps into the training set and validation set. The total number of training power density maps is set to 72,000 and the validation set is set to 18,000. All the input power matrices are normalized with respect to the mean and standard derivation of the training data. We summarize the validation mean square error (MSE), mean absolute error (MAE), and R2 score in Table 2. For two-phase VCs with hybrid wick evaporators, because each coolant has its own CNN architecture, we average the error of two-phase VCs with hybrid wick evaporators' geometries for each coolant.

As we can see from Table 2, our proposed CNN architectures are able to properly learn patterns to predict the optimal cooling parameters for each type of cooling technology. We observe that compared to two-phase VCs with hybrid wick evaporator CNN, hybrid cooling CNN has higher mean square errors and

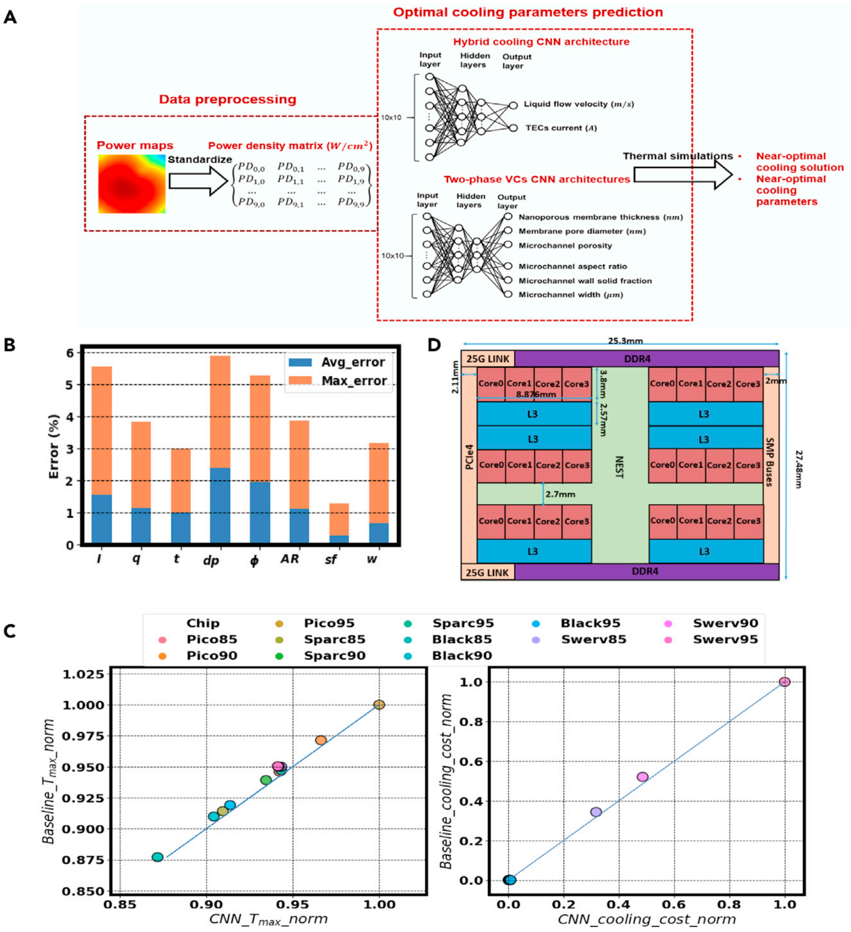


Figure 3. Deep learning-based cooling optimization flow and optimization results

(A) Deep learning-based cooling optimization flow.

(B) CNN architectures' accuracy results.

(C) The correlation plots of the maximum temperatures and cooling costs predicted using the proposed optimization flow against the results generated using the baseline methods. Baseline methods stand for the combination of MSA and CMA-ES. CNN stands for the proposed CNN architectures and optimization flow. All the data are normalized to the maximum value.

(D) IBM Power9 processor floorplan.

mean absolute errors with lower R2 scores. The reason is that hybrid cooling is more complex because of the optimization objective function. Hybrid cooling CNN also needs to take cooling power into consideration which makes the prediction more complicated and therefore less accurate. However, in two-phase VCs with hybrid wick evaporators, there is no additional cooling power at the evaporator side.

Optimization results for realistic MPSoCs

To demonstrate the predicted accuracy and search time improvements on realistic chips of our proposed optimization flow against existing cooling optimization methods (Yuan et al., 2019b, 2020), we select realistic high-power density MPSoCs from OpenROAD (Ajayi et al., 2019) with different chip sizes, floorplans, and power profiles to test the proposed DL-based cooling optimization framework. We compare the optimal results predicted using our proposed CNN architectures and optimization flow against MSA and CMA-ES from previous work with a temperature constraint of 90°C. The statistics of the realistic MPSoCs from OpenROAD are listed in Table 3. For each of the MPSoC, we first map the power profiles into 10 × 10 power density maps. We then use Equation 3 to standardize the power density maps with respect to the training power density maps:

Table 2. Validation results of the proposed CNN architectures

Metrics	Liquid flow velocity	TEC current	t	dp	φ	AR	SF	w
MSE	2.3%	2.1%	0.9%	0.6%	2.5%	0.7%	1%	2.1%
MAE	5.3%	4.7%	2%	1.5%	4%	2%	2%	4%
R2	93.2%	96%	96%	99.3%	99.2%	99.2%	99.3%	98.5%

$$PD_{\mathrm{new}} = \frac{PD_{\mathrm{original}} - \mu_{\mathrm{trainng}}}{std_{\mathrm{training}}} + b. \qquad \text{(Equation 3)}$$

$\mu_{training}$ is the mean power density of the training dataset, std_{training} is the standard derivation of the training power density dataset, and b is the bias which is defined as the ratio of the testing MPSoCs dimension over the training chip dimension.

For each cooling parameter, we calculate the average and max error for all MPSoCs and coolants and plot the percentage error in Figure 3B. The Avg_error and Max_error are defined as shown in Equation 4:

$$\begin{gathered} Avg\_error = \frac{\sum p_{\mathrm{pred}} - p_{\mathrm{base}}}{\#\ of\ MPSoCs\ \times \#\ of\ coolants}, \\ Max\_error = \max\left(\sum p_{\mathrm{pred}} - p_{\mathrm{base}}\right), \end{gathered} \qquad \text{(Equation 4)}$$

where p_{pred} is the predicted parameter by our proposed CNN architectures and p_{base} is the parameter generated using the baseline method (CMA-ES and MSA). Both CMA-ES and MSA have been validated against grid search in previous work (Yuan et al., 2019b, 2020). The reason we chose the baseline method to be CMA-ES and MSA instead of grid search is we seek to have a fast design exploration and simulation time for the baseline method, which would further show the simulation speed improvement of our proposed CNN architectures. For hybrid cooling, because the coolant is only water, $\#\ of\ coolants$ equals 1. For two-phase VCs, the $\#\ of\ coolants$ is set to 3 because there are three available coolants (water, R245fa, and R141b). As we see in Figure 3B, our proposed CNN architectures can successfully predict optimal cooling parameters for hybrid cooling and two-phase VCs with hybrid wick evaporators with a maximum error of less than 4%. Because PicoSoC with 95% utilization has the highest power density, we also show the predicted parameters using our proposed CNN architectures and baseline parameters generated using the baseline methods of PicoSoC in Table 4.

Figure 3C shows the optimization results of correlation plots for all MPSoCs. The proposed DL-based cooling optimization flow can find a similar optimal cooling solution and its cooling parameters with maximum temperature and cost difference of 0.7°C and 0.01 W compared to existing methods. Note that, the tested MPSoCs have different chip dimensions compared to the training chip size we are using, which

Table 3. Statistics of the realistic MPSoCs from OpenROAD

MPSoCs	Average power density (W/cm^2)	Utilization (%)	Dimensions (μm^2)
PicoSoC	368	85	1567 × 1567
PicoSoC	387	90	1522 × 1522
PicoSoC	409	95	1493 × 1493
Sparc	351	85	1225 × 1225
Sparc	351	90	1225 × 1225
Sparc	351	95	1225 × 225
Black_parrot	319	85	769 × 769
Black_parrot	343	90	748 × 748
Black_parrot	362	95	728 × 728
Swerv	311	85	620 × 620
Swerv	326	90	602 × 602
Swerv	338	95	595 × 595

Table 4. Predicted parameters using our proposed CNN architectures and baseline parameters generated using the baseline methods for PicoSoC

Methods	Coolant	$I(A)$	$q(m/s)$	$t(nm)$	$dp(nm)$	φ	AR	sf	$w(\mu m)$
CNN	Water	7	2.57	0.97	0.17	0.30	1.92	0.34	7.63
Baseline	Water	6.98	2.57	0.99	0.175	0.29	1.85	0.33	7.56

demonstrates that our proposed CNN architectures can be used to predict the optimal cooling parameters for any given chip sizes and power profiles. For large-size chips such as PicoSoC, Sparc, and Black_parrot, the optimal solution is always two-phase VCs with hybrid wick evaporators because it does not consume additional power on the evaporator side. For smaller chips with high power density, two-phase VCs with hybrid wick evaporators cannot efficiently spread the heat across the chip. That is the reason for Swerv MPSoCs, the optimal cooling solution is hybrid cooling. In addition, all the predicted geometries are within the valid range and all the two-phase VCs with hybrid wick evaporators' geometries satisfy the dry-out constraint. The average search time for the baseline method (MSA and CMA-ES) is 1.57 h, whereas it only takes the proposed DL-based cooling optimization flow 50 s at most to predict the optimal cooling method and its cooling parameters. Our proposed DL-based cooling optimization flow can achieve a maximum of 140X speedup when compared to using existing optimization methods. In addition, the training time for hybrid cooling CNN is 13.3 min (~21 s per epoch) and the maximum training time for two-phase VCs CNN is 18 min (~56 s per epoch). The worst-case training and inference time is calculated based on Equation 5

$$Time_{\text{worst}} = \max(Hybrid_{\text{train}} + Hybrid_{\text{infer}}, \max(VC_{\text{train}} + VC_{\text{infer}})), \qquad \text{(Equation 5)}$$

where $Hybrid_{\text{train}}$ and $Hybrid_{\text{infer}}$ are the training time and inference time for hybrid cooling, respectively. VC_{train} is the training time for two-phase VCs CNN architectures with different coolants. VC_{infer} is the inference time for two-phase VCs CNN architectures with different coolants. The worst-case training and inference time for the proposed CNN architectures is 18.83 min and the overall speedup compared to the baseline method is 5X.

Optimization results for the IBM Power9 processor

To further investigate the prediction accuracy of the proposed CNN optimization architectures, we model the IBM Power9 high-performance processor with a total chip power of 190 W (Sadasivam et al., 2017). The floorplan of the IBM Power9 processor is shown in Figure 3D and the power breakdown is shown in Table 5. We compare the optimal results predicted using our proposed CNN architectures and optimization flow against the baseline method (MSA and CMA-ES) with a temperature constraint of 90°C. We use Equation 3 to standardize the power density maps with respect to the training power density maps. The comparison results are shown in Table 6. The maximum cooling parameter difference is less than 3.8%. Because the dry-out heat flux is negatively correlated with the chip size, as the chip size increases, the dry-out heat flux decreases dramatically. In this case, both proposed CNN architectures and baseline methods cannot find optimal cooling parameters for two-phase VCs to optimize the maximum temperature under 90°C. Therefore, the optimal cooling solution is hybrid cooling. We also observe that, for a small chip with a fewer number of hot spots, compared to the power consumption of the chip, the cooling cost is not significant. However, for high-power chips with large chip sizes and more hot spots (such as IBM Power9), there will be more liquid microchannels and TEC units. Therefore, the cooling cost starts to become significant. As shown in Table 6, the cooling power is around 10% of the total chip power.

DISCUSSION

This paper introduces a DL-based cooling optimization flow for emerging cooling technologies. We designed multi-output convolutional neural network (CNN) regression models to estimate the best cooling method and its cooling design and technology parameters. We demonstrated the efficiency of using deep learning techniques on optimizing the cooling technologies against existing work. Our proposed

Table 5. IBM Power9 processor power breakdown

Components	Core (total)	Cache	Nest	I/O	DDR4
Power (W)	133	20.9	9.5	15.2	11.4

Table 6. IBM Power9 processor optimal cooling parameters, maximum temperature, and cooling power

Methods	Coolant	$I(A)$	$q(m/s)$	$T_{Max}(^{\circ}C)$	$Power_{Cooling}\ (W)$
CNN	Water	7	2.59	89.9	19.50
Baseline	Water	7	2.6	89.88	19.83

CNN architectures and DL-based cooling optimization flow can successfully predict the optimal cooling solution and cooling parameters with a maximum error of less than 4% and a maximum speedup of 140X.

Limitations of the study

Our proposed DL-based cooling optimization flow could result in local optimal results because of the training data granularity. Unless an accurate mathematical formula is created for this optimization problem, the cooling optimization method may converge to a local minimum. This is generally true for black-box optimization methods. It is not possible to create an accurate mathematical formula for this particular cooling optimization problem to solve it analytically. Therefore, we cannot guarantee that the result of our proposed cooling optimization flow is the global optimum.

An open problem is determining the optimal cooling solution and cooling parameters more broadly for new integration methods, such as 3D ICs with arbitrary layer configurations. Our current flow is applicable if the 3D IC layer configurations (i.e., which blocks are allocated on which layers) match the layer configurations available in the training data. For applying the proposed CNN optimization architecture for arbitrary 3D IC designs, the CNN regression models have to be retrained as needed to maintain the desired accuracy. We plan to provide layer partitioning configurations as inputs to the CNN regression models to tackle this limitation in our future work. Our future work also includes building CNN architectures for emerging technologies with different materials, chip thicknesses, manufacturing costs, and temperature limits. In addition, using finer granularity power density maps to train accurate CNN architectures for emerging cooling technologies is another open problem.

STAR★METHODS

Detailed methods are provided in the online version of this paper and include the following:

- KEY RESOURCES TABLE
- RESOURCE AVAILABILITY
  - Lead contact
  - Materials availability
  - Data and code availability
- METHOD DETAILS
  - Training data preparation
  - Hybrid cooling CNN architecture
  - Two-phase VCs with hybrid wick evaporators CNN architecture

SUPPLEMENTAL INFORMATION

Supplemental information can be found online at https://doi.org/10.1016/j.isci.2021.103582.

ACKNOWLEDGMENTS

We would like to thank the reviewers for their very thoughtful comments, which led to an improved version of the paper. This paper has been partially funded by the NSF CRI (CI-NEW) grant #1730316/1730003/1730389 and NSF CCF grant #1910075/1909027.

AUTHOR CONTRIBUTIONS

The program was designed by authors and all of them contributed to various research and testing phases of the project. The paper was written by all authors that contributed and commented on it.

DECLARATION OF INTERESTS

The authors declare no competing interests.

Received: October 14, 2020
Revised: September 10, 2021
Accepted: December 6, 2021
Published: January 21, 2022

REFERENCES

Ajayi, T., Chhabria, V.A., Fogaça, M., Hashemi, S., Hosny, A., Kahng, A.B., Kim, M., Lee, J., Mallappa, U., Neseem, M., and Pradipta, G. (2019). Toward an open-source digital flow: First learnings from the openroad project. In Proceedings of the 56th Annual Design Automation Conference 2019, pp. 1–4.

Beneventi, F., Bartolini, A., Tilli, A., and Benini, L. (2012). An effective gray-box identification procedure for multicore thermal modeling. IEEE Trans. Comput. *63*, 1097–1110.

Blackburn, L.D., Tuttle, J.F., and Powell, K.M. (2020). Real-time optimization of multi-cell industrial evaporative cooling towers using machine learning and particle swarm optimization. J. Clean. Prod. *271*, 122175.

Bulut, M., Kandlikar, S.G., and Sozbir, N. (2019). A review of vapor chambers. Heat Transfer Eng. *40*, 1551–1573.

Chen, H., Han, Y., Tang, G., and Zhang, X. (2020). A dynamic control system for server processor direct liquid cooling. IEEE Trans. Components, Packaging Manufacturing Tech. *10*, 786–794.

Chowdhury, I., Prasher, R., Lofgreen, K., Chrysler, G., Narasimhan, S., Mahajan, R., Koester, D., Alley, R., and Venkatasubramanian, R. (2009). On-chip cooling by superlattice-based thin-film thermoelectrics. Nat. Nanotechnol. *4*, 235–238.

Dang, B., Bakir, M.S., Sekar, D.C., King, C.R., Jr., and Meindl, J.D. (2010). Integrated microfluidic cooling and interconnects for 2D and 3D chips. IEEE Trans. Adv. Packaging *33*, 79–87.

Fan, Y., Winkel, C., Kulkarni, D., and Tian, W. (2018). Analytical design methodology for liquid based cooling solution for high TDP CPUs. In 2018 17th IEEE Intersociety Conference on Thermal and Thermomechanical Phenomena in Electronic Systems (ITherm) (IEEE), pp. 582–586.

Juneja, R., Yumnam, G., Satsangi, S., and Singh, A.K. (2019). Coupling the high-throughput property map to machine learning for predicting lattice thermal conductivity. Chem. Mater. *31*, 5145–5151.

Kaplan, F., Reda, S., and Coskun, A.K. (2017). Fast thermal modeling of liquid, thermoelectric, and hybrid cooling. In 2017 16th IEEE Intersociety Conference on Thermal and Thermomechanical Phenomena in Electronic Systems (ITherm) (IEEE), pp. 726–735.

Kaplan, F., Said, M., Reda, S., and Coskun, A.K. (2019). Locool: fighting hot spots locally for improving system energy efficiency. IEEE Trans. Comput.-Aided Des. Integr. Circuits Syst. *39*, 895–908.

Kim, N.S., Austin, T., Baauw, D., Mudge, T., Flautner, K., Hu, J.S., Irwin, M.J., Kandemir, M., and Narayanan, V. (2003). Leakage current: Moore's law meets static power. Computer *36*, 68–75.

LeCun, Y., and Bengio, Y. (1995). Convolutional networks for images, speech, and time series. Handbook Brain Theor. Neural Networks *3361*, 1995.

Lu, Z., Kinefuchi, I., Wilke, K.L., Vaartstra, G., and Wang, E.N. (2019). A unified relationship for evaporation kinetics at low Mach numbers. Nat. Commun. *10*, 1–8.

Pedram, M., and Nazarian, S. (2006). Thermal modeling, analysis, and management in VLSI circuits: Principles and methods. Proc. IEEE *94*, 1487–1501.

Pryor, R.W. (2009). Multiphysics Modeling Using COMSOL®: A First Principles Approach (Jones & Bartlett Publishers).

Ramos-Alvarado, B., Feng, B., and Peterson, G.P. (2013). Comparison and optimization of single-phase liquid cooling devices for the heat dissipation of high-power LED arrays. Appl. Therm. Eng. *59* (1-2), 648–659.

Sadasivam, S.K., Thompto, B.W., Kalla, R., and Starke, W.J. (2017). IBM Power9 processor architecture. IEEE Micro *37* (2), 40–51.

Schultz, M., Yang, F., Colgan, E., Polastre, R., Dang, B., Tsang, C., Gaynes, M., Parida, P., Knickerbocker, J., and Chainer, T. (2016). Embedded two-phase cooling of large three-dimensional compatible chips with radial channels. J. Electron. Packaging *138*, 021005.

Sheng, Y., Wu, Y., Yang, J., Lu, W., Villars, P., and Zhang, W. (2020). Active learning for the power factor prediction in diamond-like thermoelectric materials. NPJ Comput. Mater. *6*, 1–7.

Skadron, K., Stan, M.R., Huang, W., Velusamy, S., Sankaranarayanan, K., and Tarjan, D. (2003). Temperature-aware microarchitecture. ACM SIGARCH Comp. Architecture News *31*, 2–13.

Sridhar, A., Vincenzi, A., Atienza, D., and Brunschwiler, T. (2013). 3D-ICE: A compact thermal model for early-stage design of liquid-cooled ICs. IEEE Trans. Comput. *63*, 2576–2589.

Srinivasan, J., Adve, S.V., Bose, P., and Rivers, J.A. (2004). The case for lifetime reliability-aware microprocessors. ACM SIGARCH Comp. Architecture News *32*, 276.

Stolarski, T., Nakasone, Y., and Yoshimoto, S. (2018). Engineering Analysis with ANSYS Software (Butterworth-Heinemann).

Tang, L., Zhou, Y., Zheng, S., and Zhang, G. (2020). Exergy-based optimisation of a phase change materials integrated hybrid renewable system for active cooling applications using supervised machine learning method. Solar Energy *195*, 514–526.

Vaartstra, G., Lu, Z., and Wang, E.N. (2019). Simultaneous prediction of dryout heat flux and local temperature for thin film evaporation in micropillar wicks. Int. J. Heat Mass Transfer *136*, 170–177.

Yazawa, K., Ziabari, A., Koh, Y.R., Shakouri, A., Sahu, V., Fedorov, A.G., and Joshi, Y. (2012, May). Cooling power optimization for hybrid solid-state and liquid cooling in integrated circuit chips with hotspots. In 13th InterSociety Conference on Thermal and Thermomechanical Phenomena in Electronic Systems (IEEE), pp. 99–106.

Yuan, Z., Vaartstra, G., Shukla, P., Said, M., Reda, S., Wang, E., and Coskun, A.K. (2019a). Two-phase vapor chambers with micropillar evaporators: a new approach to remove heat from future high-performance chips. In 2019 18th IEEE Intersociety Conference on Thermal and Thermomechanical Phenomena in Electronic Systems (ITherm) (IEEE), pp. 456–464.

Yuan, Z., Vaartstra, G., Shukla, P., Reda, S., Wang, E., and Coskun, A.K. (2019b). Modeling and optimization of chip cooling with two-phase vapor chambers. In 2019 IEEE/ACM International Symposium on Low Power Electronics and Design (ISLPED) (IEEE), pp. 1–6.

Yuan, Z., Shukla, P., Chetoui, S., Nemtzow, S., Reda, S., and Coskun, A.K. (2021). PACT: An extensible parallel thermal simulator for emerging integration and cooling technologies. IEEE Trans. Comput. Aided Des. Integr. Circ. Syst. https://doi.org/10.1109/TCAD.2021.3079166.

Yuan, Z., Vaartstra, G., Shukla, P., Lu, Z., Wang, E., Reda, S., and Coskun, A.K. (2020). A learning-based thermal simulation framework for emerging two-phase cooling technologies. In 2020 Design, Automation & Test in Europe Conference & Exhibition (DATE) (IEEE), pp. 400–405.

Zhou, Y., Zheng, S., and Zhang, G. (2020). A review on cooling performance enhancement for phase change materials integrated systems flexible design and smart control with machine learning applications. Building Environ. *174*, 106786.

STAR★METHODS

KEY RESOURCES TABLE

REAGENT or RESOURCE	SOURCE	IDENTIFIER
Deposited Data		
Liquid cooling via microchannel model	Yuan et al., 2021	https://github.com/peaclab/PACT
IBM Power9 processor	IBM	https://ieeexplore.ieee.org/abstract/document/7924241
TEC and hybrid cooling models	Kaplan et al., 2019	https://ieeexplore.ieee.org/document/8654683
Two-phase vapor chamber model	Yuan et al., 2020	https://ieeexplore.ieee.org/document/9116480
Software and Algorithms		
HotSpot	Skadron et al., 2003	https://github.com/uvahotspot/hotspot
CNN	LeCun and Bengio, 1995	https://github.com/tensorflow/docs-l10n/blob/master/site/zh-cn/tutorials/images/cnn.ipynb

RESOURCE AVAILABILITY

Lead contact

Further information and requests for resources should be directed to the lead contact, Zihao Yuan (yuan1z@bu.edu).

Materials availability

This study did not generate new unique reagents.

Data and code availability

- This paper analyzes existing, publicly available data. These accession numbers for the datasets are listed in the key resources table.
- This paper does not report original codes.
- Any additional information required to reanalyze the data reported in this paper is available from the lead contact upon request.

METHOD DETAILS

Training data preparation

The CNN architectures shown in Figure 3A require a huge amount of data to optimize the parameters within the neural network to improve the performance of the model. One major challenge of building a CNN architecture is the preparation of the training data. Since real processors' power maps are hard to obtain, we generate a comprehensive training dataset using statistical distribution. We select a 5 $mm\times$ 5mm chip and divide the chip uniformly into 10 $\times$10 power density grids. To generate comprehensive and realistic power density maps, we choose to use Gamma distribution to randomly generate power density for each power density grid. The reason we use gamma distribution to generate random power density maps are as follows: (i) obtaining real processors power map is hard, (ii) using real processors power map may let the CNN architectures overfit to the training power maps of the chips, (iii) training power density maps may not cover corner cases, and iv) the generated power density value should be positive and most generated values should within the background power density range of 50-200 W/cm^2. The largest power density value that the selected Gamma distribution can generate is 2000 W/cm^2. We then apply data augmentation techniques to rotate and flip the power density maps to increase the training data size. The total number of power density maps we generated is 90000. We use the same gamma distribution with data augmentation techniques to generate 50000-200000 training power density maps. For each generated power density map, we need to know the optimal cooling parameters of using hybrid cooling and two-phase VCs with hybrid wick evaporators. For hybrid cooling, we apply TEC units to the power density grids that have values of more than 200 W/cm^2. The microchannel width is set to be 50 μm. We adopt

Equation 2 as the optimization objective function and run grid searches to determine the optimal liquid flow velocity and TEC current for each power density map. For two-phase VCs with hybrid wick evaporators, we select water, R245fa, and R141b as the coolants. We directly run grid searches with a finer granularity for each power density map to determine the optimal cooling parameters for each coolant.

Hybrid cooling CNN architecture

The hybrid cooling method combines the liquid microchannel layer and TEC layers into one chip stack. But since they are completely different cooling methods with different cooling performance and cooling power, the liquid flow velocity and TEC current are independent of each other. In this case, we create two branches in this CNN architecture and each branch is responsible for predicting the optimal values for either liquid flow velocity or TEC current. Both branches share the same input layer and have the same number of layers and parameters. However, since this is a multi-output CNN, the loss for each branch is different. To achieve the best regression accuracy, we build different multi-output CNN architectures with different kernel sizes, number of filters, number of convolutional layers, number of fully connected layers, and with or without batch normalization layer, and select the one that has the highest validation accuracy. Table S1 shows the details of three alternative CNN architectures of hybrid cooling.

To evaluate the accuracy of the CNN alternatives, we divide the 90000 training power density maps into the training set and validation set. The total number of training power density maps is set to 72000 and the validation set is set to 18000. All the input power matrices are normalized with respect to the mean and standard derivation of the training data. We show the accuracy results of three CNN alternatives in Table S2. To train these multi-output CNN architectures, we use Adam optimizer, and the loss function is selected to be the mean square error.

As we can see from Table S2. Model_1 is clearly overfitting with the data since the validation accuracy is at least 3.5% lower than the training accuracy. To prevent overfitting, we add additional dropout layers and increase the dropout rate to 0.5. However, the model accuracy and R2 scores start to decrease below 85%. In this case, we decide to lower the complexity of the model by using a fewer number of convolutional layers and fully connected layers, which results in model_2. The accuracy and R2 score of model_2 show that the model and the data are not closely correlated since the R2 scores are below 90%. To improve the model accuracy and R2 score, we add additional fully connected layers with additional neurons and result in model_3. As we can see from the results of model_3, the accuracy and R2 scores are higher compared to other alternatives, and the model itself is not overfitted. Therefore, we choose model_3 (as shown in Figure S1) as our hybrid cooling CNN regression model. In addition, we also experiment with different activation functions such as ReLU, Hyperbolic tangent, and Leaky ReLU. We compare the accuracy and R2 score of model_3 with ReLU, Hyperbolic tangent, and Leaky ReLU. To ensure the predicted parameter is greater than 0, we set the activation function of the last activation layer to be ReLU. We summarize the results in Table S3. As we can see from Table S3. Since ReLU achieves the highest accuracy and R2 score, we select ReLU as our activation function for all the activation layers in model_3.

Two-phase VCs with hybrid wick evaporators CNN architecture

Since two-phase VCs with hybrid wick evaporators have six different cooling parameters, there is a total of six different branches for this cooling technology. In addition, since different coolant has different cooling properties, it's not realistic to train only one CNN model to predict both the optimal cooling parameters and the coolant. To solve this problem, we train different multi-output CNNs for different coolants and conduct thermal simulations at the end to find out the optimal coolant and its cooling parameters. Compare to hybrid cooling CNN architecture, two-phase VCs with hybrid wick evaporators CNNs also need to consider the dry-out effect. In order to improve the prediction accuracy, we add additional convolutional layers in each branch, and the number of filters in each convolutional layer is doubled compared to hybrid cooling CNN architecture. We also build different CNN alternatives for each two-phase VCs with hybrid wick evaporators CNN with different coolants. We summarize 9 CNN alternatives' parameters in Table S4. For each CNN alternative, we change the number of Dropout layers from 6 to 36 and the dropout rate from 0.25 to 0.5 to prevent overfitting. After each Convolutional layer, we add batch normalization to stabilize the training process and improve the training time. We add one Max Pooling layer after all the convolutional layers to decrease the problem size. We use RMSprop as the optimizer with a learning rate of 0.001 and the loss function for each branch is set to mean square error.

To evaluate the accuracy of the CNN alternatives, we divide the 90000 training power density maps into the training set and validation set. The total number of training power density maps is set to 72000 and the validation set is set to 18000. All the input power matrices are normalized with respect to the mean and standard derivation of the training data. We show the average accuracy results of these CNN alternatives for cooling parameters in Table S5. We always start with the most complex model and our aim is to simplify the CNN by using a fewer number of convolutional layers and fully connected layers. For each of the coolants, we select model_3 as our final two-phase VCs with hybrid wick evaporators CNN model. We also select the activation functions to be ReLU, Hyperbolic tangent, and leaky ReLU. Since ReLU results in the highest accuracy, we set ReLU as our activation function for all the activation layers.