SOL ENGINEERING SERVICES

Surrogate Modeling, Multi-Fidelity Learning, and Optimization for Thermal-Fluid Design

Published surrogate accuracy and speed, calibrated uncertainty, multi-fidelity learning, multi-objective Bayesian optimization, neural operators, open-source tools and tests of the CF-JET surrogate and CFD claims

Prepared for DEVCOM Army Research Laboratory, the Ground Vehicle Systems Center, and university and industry partners of the CF-JET effort

Classification	Unclassified. Prepared from open-source information.
Date	23 November 2026

Executive Summary

Published surrogates for jet-impingement and cold plate hardware reproduce CFD to within 0.2 to 2 percent on their reported test or optimum designs after 20 to 100 simulations, so a target of ±5 percent for thermal resistance and pressure drop lies inside the reported range for smooth, low-dimensional design spaces (Table 1, Figure 2).[123] No retrieved study covers a jet array with 100 to 150 jets, interleaved return ports and a composite target plate, and none of the retrieved records reports inference time or interval coverage for a cooler. The CF-JET design point of 1.5 kW on 36 cm² and 0.02 K/W equals 72 mm²·K/W, and the errors of the surrogate and the CFD model both draw on the margin that this budget allows.[4]

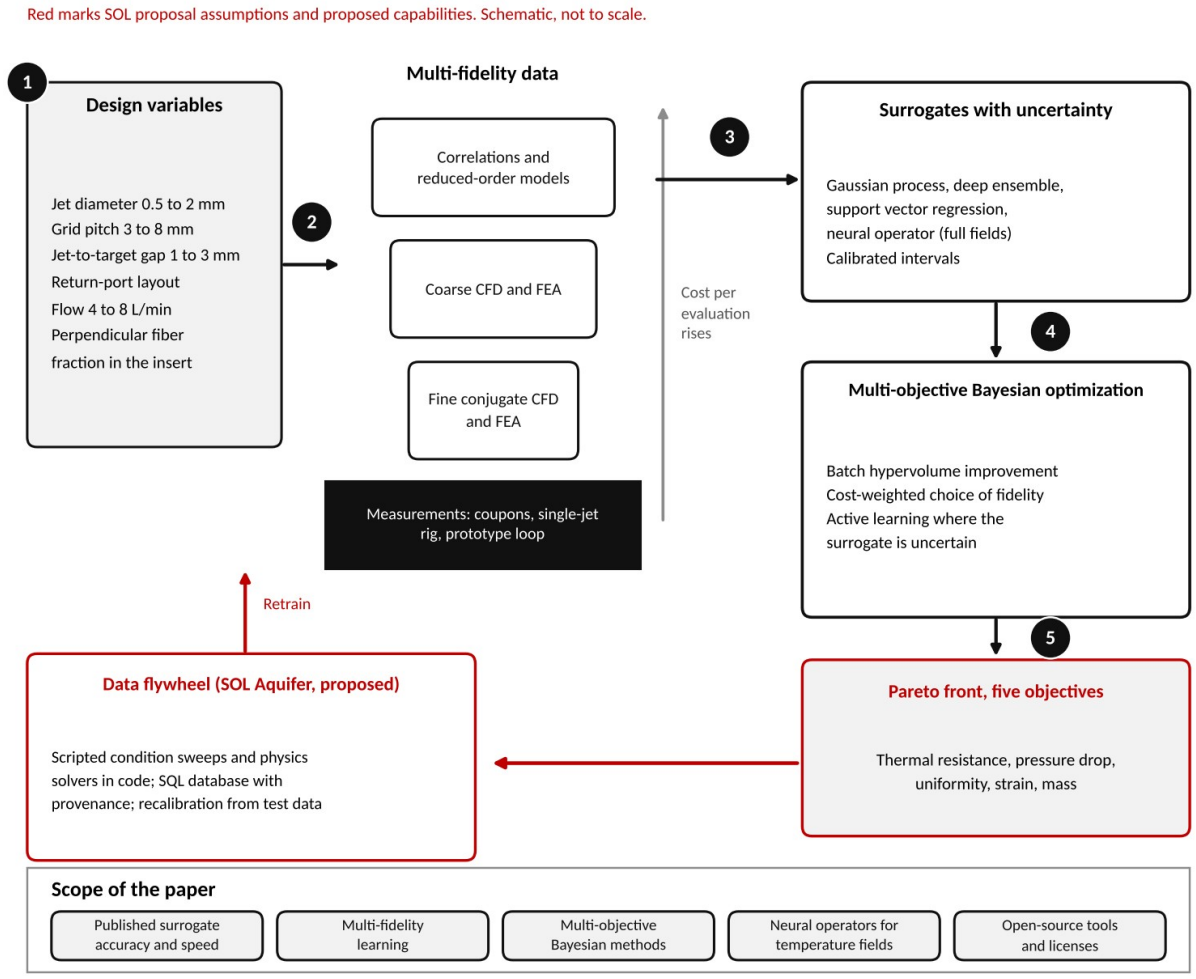
- **Accuracy and speed.** In a benchmark computed for this paper on a synthetic five-input problem, a Gaussian process placed 98.6 percent of test points within ±5 percent of the target at 200 training points and returned a prediction with its standard deviation in 0.12 ms on two CPU cores (Table 2, Figure 3). The benchmark tests the software and the sample-size requirement. It makes no statement about CF-JET hardware.
- **Calibrated uncertainty.** Gaussian process intervals with a nominal 95 percent level covered 95.8 to 99.1 percent of test points, and ensembles of five networks covered 52.8 to 75.1 percent (Table 2). Reported deep learning results show that calibration worsens outside the training data (Table 3).[5]
- **Multi-fidelity learning.** Two preprints report that 57 percent of wall time was saved in a burner design campaign and that 9.5 to 14.78 percent of evaluated airfoil designs needed the high-fidelity solver.[67] A cost model with a 126-to-1 cell-count ratio places a campaign of 120 low-fidelity and 20 to 80 high-fidelity runs at 10.5 to 40.5 percent of the cost of 200 high-fidelity runs (derived, Table 5, Figure 4).
- **Optimization.** No published cooler optimization covers all five CF-JET objectives of thermal resistance, pressure drop, uniformity, strain and mass. Uniformity appears in two studies, mass in one heat exchanger study, and strain only through a temperature non-uniformity proxy (Table 6).[8910]
- **Neural operators.** The Fourier neural operator is reported to be up to three orders of magnitude faster than traditional PDE solvers on its benchmark problems.[11] No neural operator result for a jet-impingement cold plate was found (Table 7).
- **Open-source tools.** scikit-learn and PyTorch are licensed under BSD-3-Clause, GPyTorch and BoTorch under MIT, and DVC and MLflow under Apache-2.0, so none of the six imposes copyleft terms (Table 10).[121314]
- **CFD against single-jet rig data.** The published record is split. Area-averaged results of k-ω SST in an air benchmark stay near 15 percent at most, and a 2025 laser-optical validation of a round nozzle at Re 33,250 reports an average error of 24.7 percent for SST with local errors up to 39.3 percent in the stagnation zone (Table 8, Table 11).[1516]

The implications for CF-JET are specific. A surrogate error of 5 percent combined with a CFD error of 15 percent in root-sum-square form is 15.8 percent, and a design that the surrogate predicts to meet 72 mm²·K/W needs a predicted margin of 10.2 percent when conduction and spreading use 18 mm²·K/W (derived, Table 9). The first validation step of the effort should therefore measure the CFD error on a single jet in the working fluid before the surrogate is trained, and the surrogate should report an interval whose coverage is checked on held-out simulations and on measured points.

1. Introduction and Scope

The CF-JET proposal replaces a conventional jet-impingement cold plate with a printed body that has interleaved return ports and a carbon fiber target plate. The design space for such a plate has many coupled variables, and each conjugate CFD run is expensive. The proposal pairs a physics-based workflow with learned surrogates so that thousands of candidate designs can be screened while only a few are simulated in full detail. Figure 1 shows the loop and the points where this paper tests the claims.[17]

Figure 1. Operational concept of the surrogate, multi-fidelity and optimization loop for CF-JET and scope of the paper



The design variables, operating conditions and the data flywheel are SOL proposal assumptions or proposed capabilities.[1819] *The diagram is schematic and not to scale.*

The design point is 1.5 kW on a 60 mm by 60 mm footprint, which is 36 cm² and an average heat flux of 41.7 W/cm², with 50/50 water-glycol entering at 65 °C at 4 to 8 L/min, a pressure drop of 30 kPa or less and a case-to-coolant resistance of 0.02 K/W.[20] The resistance equals 72 mm²·K/W on the footprint (derived). SOL Engineering Services describes a software pipeline, SOL Aquifer, that codifies engineering steps as code: scripted condition sweeps, physics solvers in code, a SQL database that records provenance, and surrogate training with uncertainty quantification.[21] The same documents describe uncertainty-aware neuro-symbolic inference, which combines codified physics and rules with learned surrogates, and a data flywheel that recalibrates the models from coupon, single-jet and prototype test data.[2223] The approach originated in work on building HVAC performance done with Alabama A&M University students.[24] These are proposed or demonstrated capabilities of SOL, and no independent publication of them was found.

The Army research topic on extremely sparse data names the same pairing of physics and learned components as a research need.[25]

The paper answers five questions. The first is what accuracy and speed published surrogates achieve for conjugate heat transfer and cold plate design. The second is how multi-fidelity learning reduces the number of expensive evaluations. The third is which Bayesian and multi-objective methods produce Pareto fronts across thermal resistance, pressure drop, uniformity, strain and mass. The fourth is how well neural operators predict full temperature fields. The fifth is which open-source tools are mature and what their license terms are. Two claims from the proposal are tested: that surrogates predict thermal resistance and pressure drop within ±5 percent of CFD with calibrated uncertainty in milliseconds, and that a baseline CFD model matches single-jet rig data within ±15 percent. Table 11 lists the verdicts.

2. Published Surrogate Accuracy and Speed

2.1 Published cooler surrogates reproduce CFD within 0.2 to 2 percent on 20 to 100 simulations

The studies closest to the CF-JET hardware use small neural networks or Gaussian processes trained on tens of simulations. Manaserh et al. trained a Bayesian-regularized network with one hidden layer of three neurons on a printed liquid-cooled heat sink with distributed jet inlets and guide vanes, obtained a test R^2 of 0.99, and coupled it with a genetic algorithm that lowered thermal resistance from 0.059 to 0.047 °C/W and chip temperature non-uniformity from 2.25 to 0.93 °C.[26] The record does not state the number of CFD runs behind the network, and thermal stress is not quantified, so non-uniformity serves as the proxy for strain.[27]

Fallahtafti et al. studied an impingement jet array with micro pin fins on four 2 cm by 2 cm chips with eight hotspots each. A network with one hidden layer of five neurons was trained on 89 percent of the data and reached an R^2 of 0.99, the genetic algorithm converged in about 130 generations, and the optimal designs matched CFD within 2 percent at most.[28] The training set came from 25 full-factorial CFD designs plus 8 bound cases at 8 L/min of water at 20 °C, with meshes of 19 to 168 million cells, and the reported optimum is 0.013 K/W at about 10 kPa.[29] The retrieved figures for the 8-jet design give a pressure reduction of about 90 percent against the 79 percent stated, and the difference is unresolved in the retrieved text.[30]

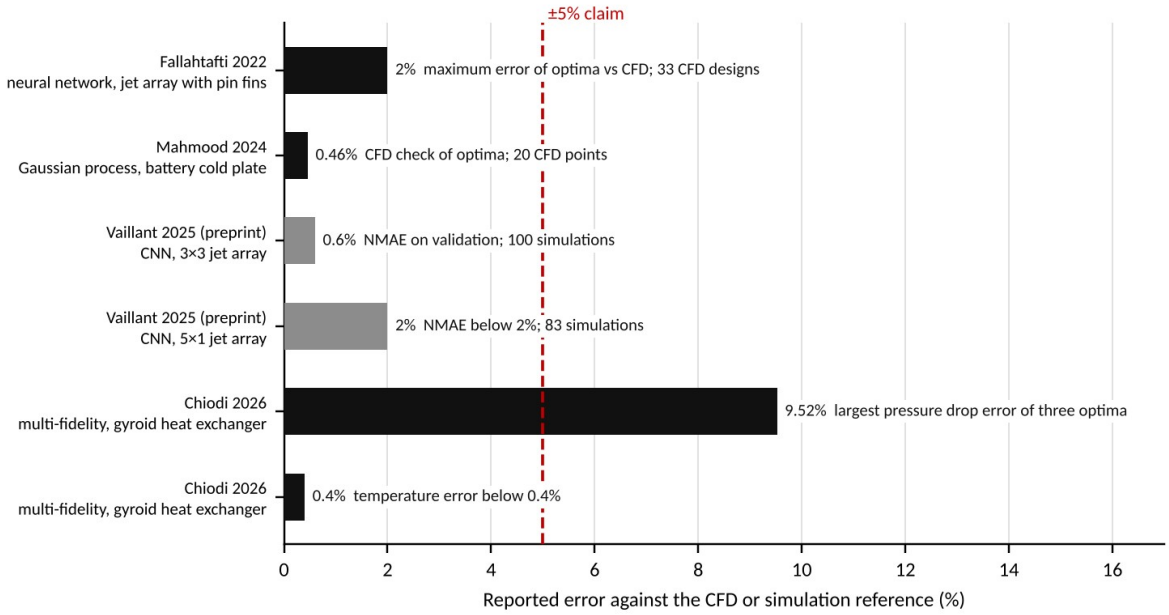
Mahmood et al. compared a Gaussian process with a radial basis function surrogate on 20 optimal Latin hypercube CFD points for a battery cooling plate. The leave-one-out mean squared error for maximum temperature was 1.40×10^{-7} for the Gaussian process and 0.043 for the radial basis function, while the pressure drop error favored the radial basis function at 0.001 against 0.004.[31] An evolutionary optimizer on the Gaussian process surrogates reduced pressure drop from 782.82 to 487.16 Pa, and CFD checks of the optima differed by up to about 0.23 and 0.46 percent.[32] Vaillant et al. report a convolutional network for impinging jet arrays trained on implicit large eddy simulations at Re under 2,000, with a normalized mean absolute error under 2 percent from 83 simulations of a 5×1 array and about 0.6 percent from 100 simulations of a 3×3 array, extended to Re under 10,000 by a correlation-based scaling (preprint).[33]

Table 1. Published surrogate studies for jet, cold plate and heat sink hardware

Study	Hardware	Surrogate	Training data	Reported accuracy and speed	Evidence
Manaserh et al. 2022[34]	Printed heat sink, jet inlets, guide vanes	Neural network, 1 hidden layer of 3 neurons	Number of CFD runs not stated	Test R^2 0.99; thermal resistance 0.059 to 0.047 °C/W; speed not reported	Peer-reviewed
Fallahtafti et al. 2022[35]	Jet array with micro pin fins, four chips	Neural network, 1 hidden layer of 5 neurons	25 full-factorial CFD designs plus 8 bound cases	R^2 0.99; optima within 2% of CFD; about 130 generations; speed not reported	Peer-reviewed
Mahmood et al. 2024[36]	Battery cooling plate	Gaussian process and radial basis function	20 optimal Latin hypercube CFD points	Leave-one-out MSE 1.40×10^{-7} (temperature); CFD check of optima 0.23% and 0.46%	Peer-reviewed
Vaillant et al. 2025[37]	Impinging jet arrays, 5×1 and 3×3	Convolutional network	83 and 100 implicit LES runs	NMAE under 2% and about 0.6%; inference time not reported	Preprint
Sikirica et al. 2022[38]	Microchannel heat sink	Random forest, gradient boosting, neural network	Not retrieved	Optimized designs equal to a CFD-based search in one-fifth of the time	Preprint
Ebbs-Picken et al. 2024[39]	Liquid-cooled cold plate, battery system	Deep encoder-decoder convolutional network	1,500 finite element simulations	Up to 65% gain in R^2 over U-Net and DenseED; absolute values not retrieved	Preprint with the DOI of a journal version

Accuracy figures are those stated in the cited records. The records for Sikirica et al. and Ebbs-Picken et al. were read at the abstract level.

Figure 2. Reported surrogate errors against the CFD or simulation reference



Black bars are peer-reviewed results and grey bars are preprints. The red dashed line marks the ±5 percent claim. The Chiodi et al. results are for a multi-fidelity model of a gyroid heat exchanger and are discussed with multi-fidelity learning.[40]

Sikirica et al. compared random forests, gradient boosting and neural networks for microchannel heat sink

optimization and report that the optimized designs matched a CFD-based search in about one-fifth of the time, with more than 10 percent lower temperature at equal pressure and more than 25 percent lower pressure drop at equal temperature than a conventional design (preprint).[41] Ebbs-Picken et al. trained deep encoder-decoder networks on 1,500 finite element simulations of a liquid-cooled cold plate and report up to a 65 percent gain in the coefficient of determination over U-Net and DenseED baselines, a result for field prediction and not for scalar outputs.[42] A Gaussian process with Bayesian optimization on an aluminum microchannel cold plate with four chips reduced entropy generation more than a response-surface method with a genetic algorithm at the same sample count, and the record gives no numbers.[43]

2.2 No retrieved record gives a surrogate inference time or a CFD run time for a conjugate cooler

Speed claims in the cooler literature are expressed as the time of a whole optimization and not as the latency of a prediction. Sikirica et al. report one-fifth of the time of a CFD-based search.[44] The records of Manaserh et al., Fallahtafti et al. and Mahmood et al. report no wall-clock time for either the CFD runs or the surrogate.[454647] The mesh sizes of 19 to 168 million cells in the Fallahtafti study indicate large runs, and a speedup ratio cannot be computed from the retrieved records.[48] The claim of prediction in milliseconds therefore has no published confirmation for a cooler, and the benchmark in Table 2 measures the latency of the software tools directly.

2.3 A synthetic five-input benchmark reaches ±5 percent at 200 training points with 0.12 ms per prediction

The benchmark in this paper uses two analytical functions of five inputs that mimic the scale of the CF-JET design variables: jet diameter 0.5 to 2 mm, pitch 3 to 8 mm, gap 1 to 3 mm, flow 4 to 8 L/min and through-thickness conductivity 20 to 300 W/m·K. One output is a thermal resistance proxy built from a jet Nusselt correlation, a conduction term and a fixed contact term, and the other is a pressure drop proxy built from jet dynamic pressure and a gap term. The functions are generic and are not CFD results. Training sets of 50 to 400 points came from a Latin hypercube with 1 percent multiplicative noise to represent mesh and convergence scatter, and 2,000 held-out points tested each model. Models came from scikit-learn 1.9.1 with fixed settings and no tuning: a Gaussian process with a Matérn 5/2 kernel and a separate length scale per input, an ensemble of five networks with two hidden layers of 64 units, and a support vector regressor.[49] The workspace had two x86-64 CPU cores.

Table 2 and Figure 3 give the results. The Gaussian process placed 98.0 percent of resistance predictions and 95.8 percent of pressure drop predictions within ±5 percent at 100 training points, and 98.6 and 99.2 percent at 200 points. The ensemble reached 96.9 and 97.2 percent at 200 points. The support vector regressor with default settings reached 75.1 and 61.4 percent at 200 points and 89.4 and 79.1 percent at 400 points, which reflects the absence of tuning. A single prediction with a standard deviation took 0.12 ms for the Gaussian process and 0.29 ms for the five-network ensemble at 200 training points, and 1,000 predictions took 5.0 and 2.6 ms.

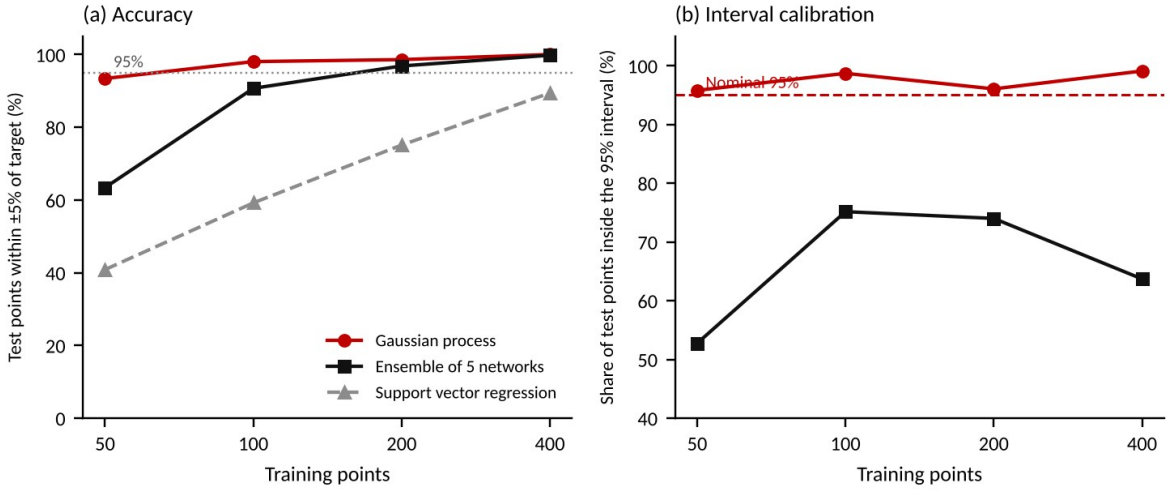
Table 2. Benchmark of surrogate tools on a synthetic five-input problem, computed in this paper

Training points	Model	Within ±5%, R / ΔP (%)	95th percentile error, R (%)	95% interval coverage, R (%)	Time per prediction (ms)	Time per 1,000 predictions (ms)	Training time (s)
50	Gaussian process	93.4 / 78.8	5.8	95.8	0.11	1.0	0.07

Training points	Model	Within ±5%, R / ΔP (%)	95th percentile error, R (%)	95% interval coverage, R (%)	Time per prediction (ms)	Time per 1,000 predictions (ms)	Training time (s)
50	Ensemble of five networks	63.4 / 68.6	16.0	52.8	0.32	3.1	0.38
50	Support vector regression	40.8 / 27.1	24.5	Not available	0.06	1.0	0.00
100	Gaussian process	98.0 / 95.8	3.3	98.7	0.11	2.5	0.18
100	Ensemble of five networks	90.7 / 94.5	7.1	75.1	0.30	2.9	0.73
100	Support vector regression	59.2 / 40.8	18.7	Not available	0.06	1.9	0.00
200	Gaussian process	98.6 / 99.2	2.8	96.0	0.12	5.0	0.57
200	Ensemble of five networks	96.9 / 97.2	3.7	74.0	0.29	2.6	1.48
200	Support vector regression	75.1 / 61.4	13.3	Not available	0.06	3.6	0.01
400	Gaussian process	100.0 / 100.0	1.6	99.1	0.14	12.3	2.81
400	Ensemble of five networks	99.8 / 100.0	1.6	63.7	0.32	3.3	1.95
400	Support vector regression	89.4 / 79.1	7.9	Not available	0.07	7.3	0.05

R is the thermal resistance proxy and ΔP is the pressure drop proxy. Errors are relative to the synthetic target on 2,000 held-out points. Times are medians of 200 single predictions on two CPU cores, with the standard deviation returned where the model provides one. The support vector regressor returns no interval. These results test the software and the sample-size requirement and are not CF-JET results.

Figure 3. Accuracy and interval calibration of three surrogates on the synthetic benchmark



Panel (a) shows the share of 2,000 test points within ±5 percent for the thermal resistance proxy. Panel (b) shows the share inside the nominal 95 percent interval. Computed in this paper.

2.4 Training-set size grows with the number of design variables and the output dimension

The benchmark has five inputs and two scalar outputs. The CF-JET design space has the five variables of the

benchmark and, in addition, the return-port layout, the fiber volume fraction of the insert and a jet pattern of 100 to 150 jets, so 200 points is a floor for planning. Exact Gaussian process training costs time that grows with the cube of the number of training points because the covariance matrix is inverted, and the measured training time rose from 0.07 s at 50 points to 2.8 s at 400 points for the resistance proxy.[50] The cost remains small beside the cost of the simulations, and it becomes the limit at several thousand points, where approximate or sparse methods apply. Field outputs need far more data than scalar outputs, as the 1,500 simulations of Ebbs-Picken et al. show.[51]

3. Calibrated Uncertainty

3.1 Gaussian process intervals met the nominal 95 percent level, and the raw spread of five networks fell short of it

A surrogate used to select designs needs an interval that says how far a prediction can be from the simulation. The Gaussian process provides a predictive standard deviation from its posterior, and the 95 percent interval built from it covered 95.8, 98.7, 96.0 and 99.1 percent of test points at 50, 100, 200 and 400 training points for the resistance proxy (Table 2). Coverage higher than the nominal level means the intervals were wider than needed. The five-network ensemble built its interval from the spread of the members, and the coverage was 52.8, 75.1, 74.0 and 63.7 percent, so the intervals were too narrow and the ensemble was overconfident at every size. Lakshminarayanan et al. propose ensembles of networks trained from random initializations as a practical way to estimate predictive uncertainty, and the benchmark shows that the raw spread of five small members does not have the nominal coverage without recalibration.[52]

The benchmark function is smooth and the test points come from the same distribution as the training points, which favors the Gaussian process. The coverage of the Gaussian process depends on a stationary kernel and on noise that matches the data, and a design in a corner of the space with sharper response can have intervals that are too narrow. Kuleshov et al. propose a recalibration step for regression models that maps predicted quantiles to observed frequencies on held-out data, and Angelopoulos and Bates describe conformal prediction, which gives finite-sample coverage under exchangeability of the calibration and test points.[5354] Both methods need held-out simulations or measurements that resemble the points where the surrogate is used.

3.2 Calibration degrades outside the training data, and the methods respond differently

A study of deep learning under dataset shift, reported through its authors' blog summary, found that accuracy fell as the shift grew, that calibration worsened, and that an ensemble of about five models degraded most gradually.[55] The study used image classification benchmarks, so the numbers do not transfer to thermal problems, and the direction of the result applies to any learned surrogate. An optimizer searches for the designs at the edge of what the surrogate knows, which is where the intervals are least reliable. Table 3 lists the methods and the conditions under which each gives valid intervals.

Table 3. Methods for calibrated prediction intervals in surrogate models

Method	What it provides	Condition for validity	Evidence
Gaussian process posterior[56]	Predictive mean and variance for each output	Kernel and noise model match the response; exact training cost grows with the cube of the training set	Textbook; computed in this paper

Method	What it provides	Condition for validity	Evidence
Deep ensemble[57]	Spread of members as an uncertainty estimate	Members differ enough to span the error; raw spread under-covered at all sizes in the benchmark	Peer-reviewed; computed in this paper
Recalibration of predicted quantiles[58]	Quantiles that match observed frequencies	Held-out data resemble the points of use	Peer-reviewed
Conformal prediction[59]	Intervals with finite-sample coverage	Calibration and test points are exchangeable; coverage is marginal and not local	Peer-reviewed monograph
Bayesian calibration with discrepancy[60]	Posterior for solver parameters and a discrepancy term	Measurements cover the region of use	Peer-reviewed
Ensemble under dataset shift[61]	Slowest degradation of calibration among methods compared	Image benchmarks; direction only	Blog summary of a conference paper

Coverage of conformal methods is marginal over the calibration distribution, and a designer who needs coverage at a specific design needs calibration points near it.

4. Multi-Fidelity Learning

4.1 Autoregressive co-kriging combines a few expensive runs with many cheap ones

Kennedy and O'Hagan describe the standard statistical form of multi-fidelity learning for computer codes. A cheap approximation of the code is evaluated many times, the expensive code is evaluated a few times, and a Gaussian process models the expensive output as a scaled copy of the cheap output plus a discrepancy function.[62] The prediction at a new design then draws on the cheap runs for the trend and on the expensive runs for the correction, and the posterior variance states where the expensive runs are still needed. For CF-JET the tiers are natural: correlations for jet arrays with interspersed extraction ports at the lowest cost, coarse CFD and FEA in the middle, fine conjugate CFD at the top, and measurements from coupons, a single-jet rig and prototypes as the final tier.[6364]

4.2 Published multi-fidelity campaigns report 43 percent of the wall time and 10 to 15 percent of the expensive evaluations

The study closest to cooler hardware is the work of Chiodi et al. on a gyroid heat exchanger and cold plate. A nonlinear autoregressive Gaussian process with a Matérn 3/2 kernel learned the closure coefficients of a porous-medium model from 120 low-fidelity cases of about 500,000 cells each, solved in OpenFOAM, and from high-fidelity cases of about 63 million cells solved with a k-ω SST model in Fluent.[65] The number of high-fidelity runs is not stated in the retrieved text. The two tiers differ by about 31 percent in the permeability coefficient and about 57 percent in the Forchheimer coefficient at the wall cell adjacent to the inlet, and a low-fidelity model alone underpredicts pressure drop.[66] With the multi-fidelity model, three optimized designs showed pressure drop errors of 2.87, 6.18 and 9.52 percent and temperature errors under 0.4 percent against high-fidelity CFD. The search was capped at 500 evaluations with four in parallel, and the Pareto front was judged converged after about 320.[67] The study reports no wall-clock saving.

Two preprints report savings in other fields. A burner design study with a multi-fidelity Bayesian optimizer that weights the acquisition by evaluation cost reports about 57 percent less total wall time than a single-fidelity campaign at comparable convergence.[68] An airfoil study that embeds active multi-fidelity learning

in the optimization reports that only 14.78 percent of evaluated individuals for the cruise condition and 9.5 percent for the take-off condition needed the high-fidelity RANS solver.[69] Lyu et al. apply transfer learning to a Fourier neural operator to transfer a low-fidelity field model to high-fidelity data and report 99 percent modeling accuracy on three problems, with sample counts and timing not retrieved (preprint).[70] Vaillant et al. extend a surrogate trained at Re under 2,000 to Re under 10,000 with a correlation-based scaling, which is a multi-fidelity step that uses a correlation as the cheap tier.[71]

Table 4. **Published multi-fidelity studies**

Study	Fidelity tiers	Reported result	Cost or time saving	Evidence
Kennedy and O'Hagan 2000[72]	Fast approximations and the full code	Autoregressive Gaussian process with a discrepancy function	Framework; no figure	Peer-reviewed
Chiodi et al. 2026[73]	120 OpenFOAM cases (about 500,000 cells) and Fluent k-ω SST (about 63 million cells)	Pressure drop errors 2.87%, 6.18%, 9.52%; temperature errors under 0.4%	Front converged at about 320 of 500 evaluations; no wall-clock figure	Peer-reviewed
Souza Lima et al. 2025[74]	Burner CFD at several fidelities	Comparable convergence to single fidelity	About 57% less wall time	Preprint
Robledo et al. 2026[75]	Fast solver and RANS	Cruise efficiency up 41.05% and take-off lift up 20.75% against the best first-generation design	RANS used for 14.78% and 9.5% of evaluated designs	Preprint
Lyu et al. 2023[76]	Low- and high-fidelity fields, Fourier neural operator	99% modeling accuracy on three problems	Not retrieved	Preprint
Vaillant et al. 2025[77]	Implicit LES at Re under 2,000 and correlation scaling	Extension to Re under 10,000	Not retrieved	Preprint

The Chiodi et al. optima include one design at 760.58 W and 1,186.15 Pa, one at 588.31 W and 501.90 Pa and one at 511.63 W and 315.30 Pa, against 363.87 W and 300.63 Pa for a uniform gyroid reference, as reported in the paper.

4.3 A 126-to-1 cost ratio puts a campaign of 20 to 80 high-fidelity runs at 10 to 42 percent of the cost of 200 runs (derived)

The cell counts in the Chiodi study give a cost ratio between the tiers. The high-fidelity case has 63 million cells against 500,000 for the low-fidelity case, a ratio of 126 (derived), and this paper takes cost proportional to cell count as a simplifying assumption. The benchmark in Table 2 shows that 200 high-fidelity points suffice for ±5 percent on a smooth five-input problem, and a campaign of 120 or 400 low-fidelity runs plus 20, 40 or 80 high-fidelity runs costs the fractions in Table 5 (derived). At a 126-to-1 ratio, 120 low-fidelity runs cost the equivalent of about one high-fidelity run, and the high-fidelity count decides the total. The model assumes that the multi-fidelity surrogate reaches the accuracy of a surrogate trained on 200 high-fidelity runs, and that assumption is the one a campaign has to test. Figure 4 places the derived scenarios beside the reported savings.

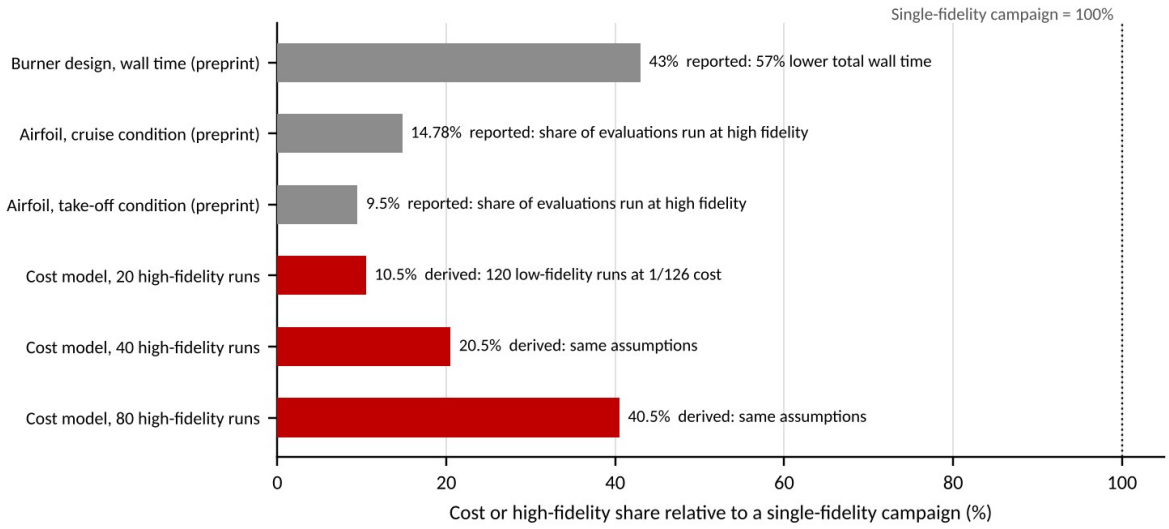
Table 5. **Cost of a multi-fidelity campaign relative to 200 high-fidelity runs (derived)**

Low-fidelity runs and high-fidelity runs	Cost ratio 1/10 (%)	Cost ratio 1/50 (%)	Cost ratio 1/126 (%)	Evidence
120 low, 20 high	16.0	11.2	10.5	Derived
120 low, 40 high	26.0	21.2	20.5	Derived

Low-fidelity runs and high-fidelity runs	Cost ratio 1/10 (%)	Cost ratio 1/50 (%)	Cost ratio 1/126 (%)	Evidence
120 low, 80 high	46.0	41.2	40.5	Derived
400 low, 20 high	30.0	14.0	11.6	Derived
400 low, 40 high	40.0	24.0	21.6	Derived
400 low, 80 high	60.0	44.0	41.6	Derived

Cost is the number of high-fidelity runs plus the number of low-fidelity runs times the cost ratio, divided by 200. The ratio of 1/126 is the cell-count ratio in the Chiodi et al. study.[78] The ratios of 1/10 and 1/50 are assumptions of this paper.

Figure 4. Reported savings from multi-fidelity learning and the derived cost model



Grey bars are reported results from preprints. Red bars are derived in this paper from 120 low-fidelity runs at 1/126 of the cost of a high-fidelity run, against a baseline of 200 high-fidelity runs.[7980]

4.4 No retrieved cooler study uses correlations, CFD and measurements in one fidelity ladder

The retrieved multi-fidelity studies use two simulation tiers. None uses a measurement tier, and none for cooler hardware uses a correlation tier together with CFD. Bayesian calibration with a discrepancy term provides the statistical route for measurements: a few field observations calibrate solver parameters and estimate the inadequacy of the solver, and the prediction includes both uncertainties.[81] The route needs measurements in the region where predictions are wanted, and a program with a few prototypes supplies few of them. A multi-fidelity ladder for CF-JET would therefore start with paired runs at two mesh levels to measure the bias between tiers, and would add the single-jet and coupon measurements as the top tier when they exist.

5. Bayesian and Multi-Objective Optimization

5.1 Published thermal-fluid optimizations cover two or three objectives and none covers all five CF-JET objectives

Table 6 lists the objectives of the retrieved optimization studies. Manaserh et al. optimize chip temperature non-uniformity, thermal resistance and pumping power with NSGA-II on a neural network,

and the optimum design they report has a pressure drop of 2.1 kPa and a non-uniformity of 0.81 °C.[82] Mahmood et al. optimize maximum temperature, temperature standard deviation and pressure drop with the GDE3 algorithm on Gaussian process surrogates at a population of 50 for 200 generations, and reduce maximum temperature from 36.38 to 35.98 °C and pressure drop from 782.82 to 487.16 Pa.[83] Fallahtafti et al. optimize thermal and flow resistance with NSGA-II.[84] Paleyes et al. optimize mass and air-side pressure drop of a heat exchanger with two inequality constraints and a binary crash constraint over nine inputs, six continuous and three discrete.[85] Strain or stress appears only as the non-uniformity proxy of Manaserh et al., and no study retrieved combines all five CF-JET objectives.

Table 6. Objectives covered by published thermal-fluid optimization studies

Study	Thermal resistance or temperature	Pressure drop or pumping power	Uniformity	Strain or stress	Mass	Algorithm
Manaserh et al. 2022[86]	Yes	Yes	Yes	Proxy only	No	NSGA-II on a neural network
Fallahtafti et al. 2022[87]	Yes	Yes	Not an objective	No	No	NSGA-II on a neural network
Mahmood et al. 2024[88]	Yes	Yes	Yes	No	No	GDE3 on Gaussian processes
Sikirica et al. 2022[89]	Yes	Yes	Not retrieved	No	No	Surrogate-based search; details not retrieved
Paleyes et al. 2022[90]	Not an objective	Yes	No	No	Yes	Batch Bayesian optimization with penalisation
Cold plate entropy study, 2025[91]	Entropy generation	Entropy generation	Not retrieved	No	No	Bayesian optimization on a Gaussian process
Chiodi et al. 2026[92]	Yes	Yes	No	No	No	Multi-fidelity surrogate with an optimizer; algorithm not retrieved

A cell reads Yes when the cited record lists the quantity as an objective. The Paleyes et al. study concerns a vehicle radiator and is the only retrieved study with mass as an objective.

5.2 Batch hypervolume methods handle noisy objectives and parallel evaluation

A multi-objective Bayesian optimizer needs an acquisition function that values a candidate by the hypervolume it adds to the current Pareto front, and the batch form needs to evaluate several candidates together because several simulations can be evaluated at the same time. Daulton et al. propose a noisy expected hypervolume improvement for parallel optimization of multiple noisy objectives and reduce the complexity of the parallel computation from exponential to polynomial in the batch size.[93] The method is available in BoTorch, which builds Bayesian optimization on PyTorch with Monte-Carlo acquisition functions.[94] Paleyes et al. use a penalisation method for batches of 20 in a nine-input heat exchanger problem with 100 random and 500 added evaluations, and they compare it only with sequential expected hypervolume improvement, with no hypervolume numbers in the retrieved text.[95] The CFD scatter that the benchmark represents with 1 percent noise is the condition that the noisy form of the acquisition addresses.

5.3 Strain and mass need modeling that no cooler study supplies

Strain in the carbon fiber target plate and the printed body depends on the temperature field, the fiber orientation and the constraint of the mounting, and a thermo-mechanical FEA gives it. The surrogate for strain therefore needs training runs of a coupled model, and its cost adds to the CFD cost. Mass follows from the geometry and the fiber volume fraction by arithmetic and needs no surrogate (assumption of this paper). A five-objective front also needs more evaluations to resolve, because the share of non-dominated points in a random sample typically rises with the number of objectives, and the hypervolume computation grows with the dimension. The retrieved studies do not report how many evaluations a front of five objectives takes, and that number is an open question for the effort.

6. Neural Operators for Temperature Fields

6.1 Neural operators learn maps between fields, and reported speedups reach three orders of magnitude on benchmark problems

A scalar surrogate predicts a summary such as thermal resistance. The uniformity objective and the hotspot behavior of the target plate need the temperature field, and neural operators predict fields. The Fourier neural operator parameterizes the integral kernel of a solution operator in Fourier space and is reported to be up to three orders of magnitude faster than traditional PDE solvers on the benchmark problems of its authors.[96] DeepONet learns nonlinear operators from data on the basis of a universal approximation theorem for operators.[97] A 2024 study of server microchips with nanofluid coolant compared five neural operators on eight design variables and reports that the Fourier neural operator was the most accurate, by an order of magnitude over the worst model, DeepONet, with a training cost 3 to 5 times lower (comparison model not retrieved) and resolution invariance, which lets a model trained on one mesh predict on another.[98]

6.2 No published neural operator result for a jet-impingement cold plate was found

Table 7 lists the evidence. The server microchip study is the closest to electronics cooling and concerns a chip with nanofluid coolant, which differs from a jet array over a composite plate in geometry, flow regime and conjugate structure.[99] The convolutional field surrogate of Ebbs-Picken et al. used 1,500 finite element simulations, and that number is the best indication of the data volume that full-field prediction needs for a cold plate.[100] A neural operator for CF-JET would need training fields from fine conjugate CFD, which makes it a second-stage model after a scalar surrogate has located the region of interest. Resolution invariance is a practical advantage for a multi-fidelity ladder, because fields from a coarse mesh and a fine mesh can train one model, and Lyu et al. pursue that idea with transfer learning.[101]

Table 7. Evidence on neural operators and field surrogates for thermal problems

Method	Problem	Reported result	Gap for CF-JET	Evidence
Fourier neural operator[102]	Parametric PDE benchmarks	Up to three orders of magnitude faster than traditional solvers	No conjugate heat transfer or jet case	Peer-reviewed conference paper
DeepONet[103]	Operator learning from data	Learns nonlinear operators on the basis of a universal approximation theorem	No cooler case	Peer-reviewed

Method	Problem	Reported result	Gap for CF-JET	Evidence
Five neural operators[104]	Server microchip with nanofluid coolant, 8 design variables	Fourier neural operator most accurate; order of magnitude over DeepONet; training cost 3 to 5 times lower; resolution invariant	Different geometry and coolant; authors and DOI not retrieved	Journal; record read at the abstract level
Fourier neural operator with transfer learning[105]	Fluid and temperature fields at two fidelities	99% modeling accuracy on three problems	Sample counts and timing not retrieved	Preprint
Deep encoder-decoder network[106]	Cold plate fields, 1,500 simulations	Up to 65% gain in R^2 over U-Net and DenseED	Convolutional network, not an operator; battery cold plate	Preprint with a journal DOI

No neural operator study for a three-dimensional jet-impingement cold plate was found in the retrieved sources.

7. Baseline CFD Against Single-Jet Rig Data

7.1 Published validations against single-jet measurements range from 1.6 to 39 percent depending on closure and quantity

The laser-optical validation by Trampe et al. compares RANS closures against measurements for single nozzles. For a round nozzle with a diameter of 25 mm at H/D of 2 and Re of 33,250, the average error was 24.7 percent for k-ω SST, 29.3 percent for GEKO, 5.9 percent for a Reynolds stress model and 20.4 percent for an optimized GEKO. At Re of 90,000 the errors were 16.0 percent for GEKO and 10 percent for the optimized GEKO.[107] For a slot nozzle at H/Dh of 5, stagnation errors were 5.5 to 12 percent at Re of 11,870 and 9.6 percent for SST at Re of 54,900, the Reynolds stress model had average errors of 36.3 and 24.3 percent, and the optimized GEKO averaged 1.6 percent.[108] Local errors up to 39.3 percent remained in the stagnation zone, and the authors' goal of 5 percent accuracy was not reached by any model across all cases.[109] The tables in the paper contain internal inconsistencies, such as 29.3 and 30.3 percent for the same GEKO case, and the values here follow the text.

The other published comparisons agree on the dependence on closure and quantity. In the air benchmark of Gurgul and Fornalik-Wajs at Re of 10,000 to 30,000 and H/D of 1 to 6, SST was within about 2 to 6 percent at stagnation, and the best SST variants had median area-averaged differences under 10 percent and maxima near 15 percent.[110] The review of Zuckerman and Lior reports local errors of 15 to 30 percent for the best k-ε variants.[111] A single-jet study at Re of 23,000 and H/d of 2 found SST the best balance of accuracy and cost, with the stagnation value changing by 11.12 percent when the mesh base size was coarsened to 2 mm.[112] A comparison of CFD with measurement in an NREL liquid cooling program found that the thermal resistance of a jet configuration matched at 0.116 °C/W for 1.67×10^{-4} m^3/s while the pressure drop differed by 14.8 percent (read from a presentation table and requiring independent verification).[113] No source retrieved reports the error of a closure for a water-glycol jet array over a layered target.

Table 8. Reported CFD errors against single-jet measurements and benchmarks

Source	Configuration	Reported error	Evidence
Trampe et al. 2025[114]	Round nozzle, D = 25 mm, H/D = 2, Re 33,250	Average: SST 24.7%; GEKO 29.3%; Reynolds stress 5.9%; optimized GEKO 20.4%	Conference paper, measurement

Source	Configuration	Reported error	Evidence
Trampe et al. 2025[115]	Same nozzle, Re 90,000	GEKO 16.0%; optimized GEKO 10%	Conference paper, measurement
Trampe et al. 2025[116]	Slot nozzle, H/Dh = 5	Stagnation 5.5% to 12% at Re 11,870; SST 9.6% at Re 54,900; Reynolds stress average 36.3% and 24.3%; optimized GEKO average 1.6%; local errors up to 39.3%	Conference paper, measurement
Gurgul and Fornalik-Wajs 2023[117]	Air, H/D 1 to 6, Re 10,000 to 30,000	SST stagnation about 2% to 6%; median average under 10% and maximum near 15% for the best SST variants	Peer-reviewed benchmark
Zuckerman and Lior 2006[118]	Review of single jets	Best k-ε variants 15% to 30% locally; k-ω up to 30% locally	Peer-reviewed review
Chitsazan et al. 2021[119]	Single jet, Re 23,000, H/d = 2	Stagnation value changes 11.12% on a 2 mm base mesh	Peer-reviewed
Narumanchi 2013[120]	Jet configuration at 1.67×10^{-4} m^3/s	Thermal resistance matched at 0.116 °C/W; pressure drop 14.8%	Government presentation

Error signs and conditions follow the cited sources. Section numbers are given for the review because page numbers were not available in the retrieved text.

7.2 A 15 percent CFD error and a 5 percent surrogate error need a predicted margin of about 10 percent (derived)

The standard for verification and validation of CFD, ASME V&V 20, compares the difference between simulation and measurement with a validation uncertainty that combines numerical, input and experimental uncertainties.[121] The comparison tells whether the model error is larger than the combined uncertainties, and the error that remains sets the margin a design needs. Suppose conduction and spreading use 18 mm²·K/W of the 72 mm²·K/W budget and convection uses 54 mm²·K/W, as an assumption of this paper. A relative error e on the convective term needs a predicted margin of 1 − (18 + 54/(1 + e))/72.[122] Table 9 gives the margins for the surrogate and CFD errors alone and in combination.

Table 9. Predicted margin needed on 72 mm²·K/W for combinations of surrogate and CFD error (derived)

Error source	Relative error (%)	Predicted margin, conduction 18 mm²·K/W (%)	Predicted margin, conduction 36 mm²·K/W (%)
Surrogate alone	5.0	3.6	2.4
CFD against single-jet rig	15.0	9.8	6.5
Root-sum-square of 5% and 15%	15.8	10.2	6.8
Linear sum of 5% and 15%	20.0	12.5	8.3
Product of factors 1.05 and 1.15	20.8	12.9	8.6

Derived in this paper. The conduction shares of 18 and 36 mm²·K/W are assumptions of this paper. The root-sum-square treats the two errors as independent, and the linear sum and product treat them as aligned.

The surrogate error is a fraction of the CFD error in this arithmetic, so the validation of the CFD model decides the margin. A CFD error of 25 percent, close to the 24.7 percent SST average at Re 33,250, needs a predicted margin of 15.0 percent at conduction of 18 mm²·K/W (derived). A design that the surrogate predicts to meet the budget with a 5 percent margin falls short of the 10.2 percent root-sum-square requirement, and such a design needs a hardware test before it is accepted.

8. Open-Source Tools and License Terms

8.1 All six tools are permissively licensed with different notice and patent terms

scikit-learn is licensed under BSD-3-Clause, and its repository page shows support for Python 3.11 or newer.[123] GPyTorch states that it is MIT licensed.[124] BoTorch states that it is MIT licensed, was at version 0.18.1 released on 8 June 2026 on its package page, requires Python 3.11 or newer, and cites Balandat et al. for the method.[125][126] PyTorch is licensed under BSD-3-Clause according to its package metadata.[127] DVC is distributed under the Apache license version 2.0 according to its repository README.[128] MLflow is Apache-2.0 licensed, is described as backed by the Linux Foundation, and provides experiment tracking and a model registry with versioning and lifecycle stages.[129] Package pages and repositories are trade and project material and not peer-reviewed sources.

The BSD-3-Clause license permits redistribution with the copyright notice and forbids using the names of the contributors to endorse derived products.[130] The MIT license requires that its notice be included in all copies.[131] The Apache license grants a patent license and conditions redistribution on providing a copy of the license, marking modified files, retaining notices and including the NOTICE file where one exists.[132] None of the three is a copyleft license, so a delivered CF-JET toolchain does not have to be released under the same terms. The licenses of the third-party dependencies of each tool apply separately, and a software bill of materials for the delivered pipeline should list them. This paper does not give legal advice, and counsel should confirm the terms for a government delivery.

Table 10. Open-source tools for the surrogate and optimization workflow and their license terms

Tool	Role in the workflow	License	Principal terms	Evidence
scikit-learn[133]	Gaussian process, support vector regression, random forest, small networks, scaling	BSD-3-Clause	Redistribution with notice; no endorsement using contributor names[134]	Repository page
GPyTorch[135]	Gaussian processes on PyTorch	MIT	Notice included in all copies[136]	Package page
BoTorch[137]	Bayesian and multi-objective optimization on PyTorch	MIT	Notice included in all copies; version 0.18.1 (8 June 2026)	Package page
PyTorch[138]	Tensors, neural networks, neural operators	BSD-3-Clause	Redistribution with notice; no endorsement using contributor names	Package metadata
DVC[139]	Versioning of data files and pipelines	Apache-2.0	Patent grant; copy of license, modification notices, retained notices and NOTICE file on redistribution[140]	Repository README
MLflow[141]	Experiment tracking and model registry	Apache-2.0	Same as DVC	Project documentation

The terms follow the license texts as summarized by the cited pages. Trade and project material is labeled as such.

Experiment tracking and data versioning address the provenance that the SOL Aquifer design requires. A run that records the solver version, mesh, closure, input design and result with the surrogate training set lets a reviewer reproduce a prediction afterward, and MLflow and DVC provide those records without a license cost.[142][143][144] The SQL database with provenance that SOL describes serves the same function, and the choice between a custom database and these tools is a design decision of the effort.

9. Implications for the CF-JET Effort

9.1 Verdicts on the two claims

Table 11. Tests of the proposal claims

Claim	Evidence	Verdict	Condition
Surrogates predict thermal resistance and pressure drop within ±5% of CFD	Published surrogates report 0.2% to 2% against CFD on 20 to 100 simulations; Gaussian process on a synthetic five-input problem: 98.6% of points within ±5% at 200 points[145][146][147]	Conditionally supported	Smooth, low-dimensional space; no cooler study with 100 to 150 jets and a composite plate
The surrogates include calibrated uncertainty	Gaussian process intervals covered 95.8% to 99.1%; ensembles of five covered 52.8% to 75.1%; no cooler study reports coverage	Conditionally supported	Needs held-out coverage tests and recalibration for ensembles
The surrogates predict in milliseconds	0.12 ms for the Gaussian process and 0.29 ms for the ensemble per prediction on two CPU cores	Supported for scalar outputs	Untested for field outputs; no cooler record reports latency
A baseline CFD model matches single-jet rig data within ±15%	SST area-averaged medians under 10% and maxima near 15% in air; 24.7% average and 39.3% local in a 2025 round-nozzle validation[148][149]	Not demonstrated	Needs measurement in water-glycol at the CF-JET jet size and a defined quantity

The verdicts apply to the evidence retrieved for this paper. The synthetic benchmark tests the software and does not test CF-JET hardware.

9.2 Risks

- **Surrogate inherits solver bias.** A surrogate trained on CFD reproduces the CFD error, which Table 8 puts at 1.6 to 39 percent depending on closure and quantity. Only measurements correct it, and the margin in Table 9 is the cost of leaving it uncorrected.
- **Overconfident intervals.** Ensembles of five networks covered 52.8 to 75.1 percent of points at a nominal 95 percent (Table 2). An optimizer that searches for the best predicted design finds the points where such intervals fail, so an uncalibrated ensemble can select designs that miss the budget.
- **Low-fidelity bias.** In the multi-fidelity study of a gyroid heat exchanger the low-fidelity closures differed by 31 to 57 percent at the wall cell and underpredicted pressure drop, and the corrected model still showed errors up to 9.52 percent (Table 4).[150] A ladder of tiers can fail where the bias between tiers varies across the design space.
- **Missing objectives.** No cooler study models strain, and the CF-JET plate depends on it (Table 6). The coupled thermo-mechanical runs raise the cost beyond the thermal-fluid cost of Table 5.
- **Data volume for fields.** Field prediction in the retrieved cold plate work used 1,500 simulations, and a jet array of 100 to 150 jets has more variables (Table 7).[151]

9.3 Design implications

- **Validate CFD first.** Measure a single jet at the CF-JET diameter in the working fluid, report the area-averaged and the stagnation values separately, and state the comparison error with the V&V 20 validation uncertainty.[152] Set the predicted margin at 10 percent or more for the conduction shares

assumed in Table 9.

- **Use a Gaussian process for scalar outputs and check coverage.** The benchmark reached ±5 percent at 200 points for five inputs, and exact training stayed under 3 s at 400 points (Table 2). Test coverage on held-out simulations and measured points before the surrogate selects designs.
- **Measure the tier bias early.** Run 10 to 20 paired designs at two mesh levels before committing to a ladder, and compute the cost fractions of Table 5 with the measured cost ratio.
- **Optimize with batch hypervolume methods.** BoTorch provides the noisy batch acquisition, and mass can be computed from geometry while strain comes from a thermo-mechanical surrogate.[153][154]
- **Record provenance.** Log the solver version, mesh, closure, inputs and results for every run, whether in the SQL database that SOL describes or in DVC and MLflow.[155]
- **Add a field model second.** A neural operator or a convolutional field surrogate follows the scalar surrogate and needs training fields from fine CFD.

9.4 Open research questions

- What is the error of k-ω SST and of the alternatives for a water-glycol jet at 0.5 to 2 mm diameter, for area-averaged heat transfer and at stagnation?
- How many high-fidelity runs does a ±5 percent surrogate need for a 100 to 150 jet array with interleaved returns, and how does that number depend on the number of design variables?
- What coverage do Gaussian process and recalibrated ensemble intervals reach on held-out CF-JET designs and on measured points?
- How large is the bias between a correlation tier, a coarse CFD tier and a fine CFD tier, and does it vary across the design space?
- How many evaluations does a Pareto front of five objectives need, and what does a thermo-mechanical surrogate add to the cost?
- Can a neural operator trained on fine CFD fields predict the plate temperature field of an unseen design within the uniformity tolerance?

10. Conclusion

The published record supports the ±5 percent accuracy target for scalar surrogates of smooth, low-dimensional thermal-fluid problems, supports prediction in fractions of a millisecond for the tools tested, and supports the use of multi-fidelity learning to cut expensive evaluations, with reported savings between 57 percent of wall time and about 85 to 90 percent of evaluations. It does not demonstrate the accuracy or the interval calibration for a 100 to 150 jet array with a composite target plate, and no study covers all five objectives or predicts the temperature field of a jet-impingement cold plate with a neural operator. The ±15 percent claim for CFD against a single-jet rig is supported by averaged SST results in an air benchmark and contradicted by a 2025 validation that reports 24.7 percent on average and 39.3 percent locally. The six open-source tools have permissive licenses. The CFD validation in the working fluid sets the margin that the surrogate and the optimizer can use, and it is the first measurement the effort needs.

Bibliography

Angelopoulos, Anastasios N., and Stephen Bates. “Conformal Prediction: A Gentle Introduction.” *Foundations and Trends in Machine Learning* 16, no. 4 (2023): 494–591. https://doi.org/10.1561/2200000101.

Apache Software Foundation. “Apache License, Version 2.0.” January 2004. https://www.apache.org/licenses/LICENSE-2.0.

U.S. Army DEVCOM Army Research Laboratory. “Artificial Intelligence and Machine Learning with Extremely Sparse Data (ARL-BAA-0038).” In *Current Research Topics for DEVCOM ARL BAA for Foundational Research*, 9 December 2024. https://files.simpler.grants.gov/opportunities/5e185327-5f25-4c09-889a-43615e1f54da/attachments/e4621f8a-b19c-40ec-8c59-6611cc4fa276/Current_Research_Topics_for_DEVCOM_ARL_BAA_9_Dec2024.pdf.

American Society of Mechanical Engineers. *Standard for Verification and Validation in Computational Fluid Dynamics and Heat Transfer*. ASME V&V 20-2009 (R2016). New York: ASME, 2009. https://webstore.ansi.org/Standards/ASME/ASME202009R2016.

Balandat, Maximilian, Brian Karrer, Daniel R. Jiang, Samuel Daulton, Benjamin Letham, Andrew Gordon Wilson, and Eytan Bakshy. “BoTorch: A Framework for Efficient Monte-Carlo Bayesian Optimization.” *Advances in Neural Information Processing Systems* 33 (2020).

BoTorch developers. *botorch*. Python Package Index. Accessed 23 November 2026. https://pypi.org/project/botorch/.

Chiodi, A., A. Alaia, E. Lombardi, M. Cisternino, K. Gkaragkounis, A. Ferrero, and S. Shahpar. “A Machine-Learning Based Approach for Multi-Scale Optimisation of Heat Exchangers with Lattice-Like Topology.” *International Journal of Heat and Mass Transfer* 256, part 2 (2026): 128035. https://doi.org/10.1016/j.ijheatmasstransfer.2025.128035.

Chitsazan, Ali, Georg Klepp, and Birgit Glasmacher. “Numerical Prediction of the Second Peak in the Nusselt Number Distribution from an Impinging Round Jet.” *International Journal of Heat and Technology* 39, no. 4 (2021): 1243–1252. https://doi.org/10.18280/ijht.390422.

Daulton, Samuel, Maximilian Balandat, and Eytan Bakshy. “Parallel Bayesian Optimization of Multiple Noisy Objectives with Expected Hypervolume Improvement.” *Advances in Neural Information Processing Systems* 34 (2021): 2187–2200.

Iterative. *DVC*. README, source repository, GitHub. Accessed 23 November 2026. https://github.com/iterative/dvc.

Ebbs-Picken, Takiah, David A. Romero, Carlos M. Da Silva, and Cristina H. Amon. “Deep Encoder-Decoder Hierarchical Convolutional Neural Networks for Conjugate Heat Transfer Surrogate Modeling.” arXiv:2311.17068. https://doi.org/10.1016/j.apenergy.2024.123723.

“Cold Plate Optimization with Entropy Minimization Using Bayesian Optimization.” In *Proceedings of the ASME 2025 International Design Engineering Technical Conferences and Computers and Information in Engineering Conference*. Authors not listed on the retrieved record. https://par.nsf.gov/biblio/10631181.

Fallahtafti, Najmeh, et al. “Shape Optimization of Hotspot Targeted Micro Pin Fins for Heterogeneous Integration Applications.” *International Journal of Heat and Mass Transfer* 192 (2022): 122897.

GPyTorch developers. *gpytorch*. Python Package Index. Accessed 23 November 2026. https://pypi.org/project/gpytorch/.

Gurgul, Sebastian, and Elżbieta Fornalik-Wajs. “On the Measure of the Heat Transfer Performance of RANS Turbulence Models in Single Round Jet Impingement.” *Energies* 16, no. 21 (2023): 7236. https://doi.org/10.3390/en16217236.

Kennedy, Marc C., and Anthony O’Hagan. “Bayesian Calibration of Computer Models.” *Journal of the Royal Statistical Society: Series B* 63, no. 3 (2001): 425–464. https://doi.org/10.1111/1467-9868.00294.

Kennedy, Marc C., and Anthony O’Hagan. “Predicting the Output from a Complex Computer Code When Fast Approximations Are Available.” *Biometrika* 87, no. 1 (2000): 1–13. https://doi.org/10.1093/biomet/87.1.1.

Kuleshov, Volodymyr, Nathan Fenner, and Stefano Ermon. “Accurate Uncertainties for Deep Learning Using Calibrated

Regression." *Proceedings of Machine Learning Research* 80 (2018): 2796–2804.

Lakshminarayanan, Balaji, Alexander Pritzel, and Charles Blundell. "Simple and Scalable Predictive Uncertainty Estimation Using Deep Ensembles." *Advances in Neural Information Processing Systems* 30 (2017): 6402–6413.

Li, Zongyi, Nikola Kovachki, Kamyar Azizzadenesheli, Burigede Liu, Kaushik Bhattacharya, Andrew Stuart, and Anima Anandkumar. "Fourier Neural Operator for Parametric Partial Differential Equations." arXiv:2010.08895. Presented at the International Conference on Learning Representations, 2021. https://arxiv.org/abs/2010.08895.

Lu, Lu, Pengzhan Jin, Guofei Pang, Zhongqiang Zhang, and George Em Karniadakis. "Learning Nonlinear Operators via DeepONet Based on the Universal Approximation Theorem of Operators." *Nature Machine Intelligence* 3, no. 3 (2021): 218–229. https://doi.org/10.1038/s42256-021-00302-5.

Lyu, Yanfang, Xiaoyu Zhao, Zhiqiang Gong, Xiao Kang, and Wen Yao. "Multi-Fidelity Prediction of Fluid Flow and Temperature Field Based on Transfer Learning Using Fourier Neural Operator." arXiv:2304.06972, preprint, submitted 14 April 2023. https://arxiv.org/abs/2304.06972.

Mahmood, Ahmed, Timothy Cockerill, Greg de Boer, Jochen Voss, and Harvey Thompson. "Heat Transfer Modeling and Optimal Thermal Management of Electric Vehicle Battery Systems." *Energies* 17, no. 18 (2024): 4575. https://doi.org/10.3390/en17184575.

Manaserh, Yaman "Mohammad Ali," Ahmad R. Gharaibeh, Mohammad I. Tradat, Srikanth Rangarajan, Bahgat G. Sammakia, and Husam A. Alissa. "Multi-Objective Optimization of 3D Printed Liquid Cooled Heat Sink with Guide Vanes for Targeting Hotspots in High Heat Flux Electronics." *International Journal of Heat and Mass Transfer* 184 (2022): 122287. https://doi.org/10.1016/j.ijheatmasstransfer.2021.122287.

MLflow project. "MLflow Documentation Overview." Accessed 23 November 2026. https://mlflow.org/llms.txt.

Narumanchi, Sreekant. "Advanced Liquid Cooling R&D." Presentation, project APE039. National Renewable Energy Laboratory, U.S. Department of Energy Vehicle Technologies Program, 14 May 2013. https://www.energy.gov/eere/vehicles/articles/advanced-liquid-cooling-rd.

"Neural Operator Models for Predicting Physical Fields in Server Electronic Microchips Doped with Water-Al2O3 Nanofluid Coolant." *Case Studies in Thermal Engineering*, March 2024. Authors, volume and DOI not retrieved.

Open Source Initiative. "The 3-Clause BSD License." Accessed 23 November 2026. https://opensource.org/license/bsd-3-clause.

Open Source Initiative. "The MIT License." Accessed 23 November 2026. https://opensource.org/license/mit.

Paleyes, Andrei, Henry B. Moss, Victor Picheny, Piotr Zulawski, and Felix Newman. "A Penalisation Method for Batch Multi-Objective Bayesian Optimisation with Application in Heat Exchanger Design." arXiv:2206.13326, preprint, submitted 27 June 2022. https://arxiv.org/abs/2206.13326.

PyTorch contributors. *torch*. Python Package Index. Accessed 23 November 2026. https://pypi.org/project/torch/.

Rasmussen, Carl Edward, and Christopher K. I. Williams. *Gaussian Processes for Machine Learning*. Cambridge, MA: MIT Press, 2006.

Rattner, Alexander S. "General Characterization of Jet Impingement Array Heat Sinks with Interspersed Fluid Extraction Ports for Uniform High-Flux Cooling." *Journal of Heat Transfer* 139, no. 8 (August 2017): 082201. https://doi.org/10.1115/1.4036090.

Robledo, Isaac, Alberto Vilariño, Arnau Miró, Oriol Lehmkuhl, Carlos Sanmiguel Vila, and Rodrigo Castellanos. "Optimization-Embedded Active Multi-Fidelity Surrogate Learning for Multi-Condition Airfoil Shape Optimization." arXiv:2603.17057, preprint, submitted 17 March 2026. https://arxiv.org/abs/2603.17057.

scikit-learn developers. *scikit-learn*. Source repository, GitHub. Accessed 23 November 2026. https://github.com/scikit-learn/scikit-learn.

Sikirica, Ante, Luka Grbčić, and Lado Kranjčević. "Machine Learning Based Surrogate Models for Microchannel Heat Sink

Optimization.” arXiv:2208.09683, preprint, version 2, 8 December 2022. https://arxiv.org/abs/2208.09683.

Snoek, Jasper, and Zachary Nado. “Can You Trust Your Model’s Uncertainty?” Google Research blog, 15 January 2020. https://research.google/blog/can-you-trust-your-models-uncertainty/.

SOL Engineering Services, LLC. *CF-JET: AI/ML-Enabled Design and Development of an Additively Manufactured, Liquid-Cooled Jet-Impingement Cold Plate with Optimized Carbon Fiber Target Plate*. Proposal, Rev Final 2, n.d.

SOL Engineering Services, LLC. *SOL Aquifer* software pipeline diagram. n.d.

SOL Engineering Services, LLC. “Modeling HVAC System Performance: Code as Architecture Approach to AI.” Work carried out with Alabama A&M University students, n.d.

Souza Lima, Patrick, Paulo Roberto Santana dos Reis, Alex Álisson Bandeira Santos, Ehecatl Antonio del Río Chanona, and Idelfonso Bessa dos Reis Nogueira. “Multi-Fidelity Bayesian Optimization Framework for CFD-Based Non-Premixed Burner Design.” arXiv:2511.23140, preprint, submitted 28 November 2025. https://arxiv.org/abs/2511.23140.

Trampe, Eileen, Jan-Henrik Rieger, Dominik Büschgens, and Christian Wuppermann. “Laser-Optical Validation and Comparative Analysis of Numerical Heat Transfer Models for Single Nozzle Impingement Flows.” In *Proceedings of the 19th International Conference on Fluid Flow Technologies (CMFF’25)*. Budapest: Budapest University of Technology and Economics, 2025. https://www.cmff.hu/papers25/CMFF25_Final_Paper_PDF_13.pdf.

Vaillant, Mikael, Victor Oliveira Ferreira, Wiebke Mainville, Jean-Michel Lamarre, Vincent Raymond, Moncef Chioua, and Bruno Blais. “Surrogate Model for Heat Transfer Prediction in Impinging Jet Arrays Using Dynamic Inlet/Outlet and Flow Rate Control.” arXiv:2507.07034, preprint, submitted 9 July 2025, revised 18 March 2026. https://arxiv.org/abs/2507.07034.

Zuckerman, Neil, and Noam Lior. “Jet Impingement Heat Transfer: Physics, Correlations, and Numerical Modeling.” *Advances in Heat Transfer* 39 (2006): 565–631. https://doi.org/10.1016/S0065-2717(06)39006-5.

Notes

[1]Najmeh Fallahtafti et al., “Shape Optimization of Hotspot Targeted Micro Pin Fins for Heterogeneous Integration Applications,” *International Journal of Heat and Mass Transfer* 192 (2022): 122897.

[2]Ahmed Mahmood, Timothy Cockerill, Greg de Boer, Jochen Voss, and Harvey Thompson, “Heat Transfer Modeling and Optimal Thermal Management of Electric Vehicle Battery Systems,” *Energies* 17, no. 18 (2024): 4575, https://doi.org/10.3390/en17184575.

[3]Mikael Vaillant et al., “Surrogate Model for Heat Transfer Prediction in Impinging Jet Arrays Using Dynamic Inlet/Outlet and Flow Rate Control,” arXiv:2507.07034 (preprint, submitted 9 July 2025; revised 18 March 2026), https://arxiv.org/abs/2507.07034.

[4]SOL Engineering Services, LLC, *CF-JET: AI/ML-Enabled Design and Development of an Additively Manufactured, Liquid-Cooled Jet-Impingement Cold Plate with Optimized Carbon Fiber Target Plate*, proposal, Rev Final 2, n.d.

[5]Jasper Snoek and Zachary Nado, “Can You Trust Your Model’s Uncertainty?,” Google Research blog, 15 January 2020, https://research.google/blog/can-you-trust-your-models-uncertainty/.

[6]Patrick Souza Lima et al., “Multi-Fidelity Bayesian Optimization Framework for CFD-Based Non-Premixed Burner Design,” arXiv:2511.23140 (preprint, submitted 28 November 2025), https://arxiv.org/abs/2511.23140.

[7]Isaac Robledo et al., “Optimization-Embedded Active Multi-Fidelity Surrogate Learning for Multi-Condition Airfoil Shape Optimization,” arXiv:2603.17057 (preprint, submitted 17 March 2026), https://arxiv.org/abs/2603.17057.

[8]Yaman “Mohammad Ali” Manaserh et al., “Multi-Objective Optimization of 3D Printed Liquid Cooled Heat Sink with Guide Vanes for Targeting Hotspots in High Heat Flux Electronics,” *International Journal of Heat and Mass Transfer* 184 (2022): 122287, https://doi.org/10.1016/j.ijheatmasstransfer.2021.122287.

[9]Mahmood et al., “Heat Transfer Modeling and Optimal Thermal Management of Electric Vehicle Battery Systems.”

[10]Andrei Paleyes et al., “A Penalisation Method for Batch Multi-Objective Bayesian Optimisation with Application in Heat Exchanger Design,” arXiv:2206.13326 (preprint, submitted 27 June 2022), https://arxiv.org/abs/2206.13326.

[11]Zongyi Li et al., “Fourier Neural Operator for Parametric Partial Differential Equations,” arXiv:2010.08895, presented at the International Conference on Learning Representations, 2021, https://arxiv.org/abs/2010.08895.

[12]scikit-learn developers, *scikit-learn*, source repository, GitHub, accessed 23 November 2026, https://github.com/scikit-learn/scikit-learn.

[13]GPyTorch developers, *gpytorch*, Python Package Index, accessed 23 November 2026, https://pypi.org/project/gpytorch/.

[14]Iterative, *DVC*, README, source repository, GitHub, accessed 23 November 2026, https://github.com/iterative/dvc.

[15]Sebastian Gurgul and Elżbieta Fornalik-Wajs, “On the Measure of the Heat Transfer Performance of RANS Turbulence Models in Single Round Jet Impingement,” *Energies* 16, no. 21 (2023): 7236, https://doi.org/10.3390/en16217236.

[16]Eileen Trampe et al., “Laser-Optical Validation and Comparative Analysis of Numerical Heat Transfer Models for Single Nozzle Impingement Flows,” in *Proceedings of the 19th International Conference on Fluid Flow Technologies (CMFF’25)* (Budapest: Budapest University of Technology and Economics, 2025), https://www.cmff.hu/papers25/CMFF25_Final_Paper_PDF_13.pdf.

[17]SOL Engineering Services, *CF-JET*, Rev Final 2.

[18]SOL Engineering Services, *CF-JET*, Rev Final 2.

[19]SOL Engineering Services, LLC, *SOL Aquifer* software pipeline diagram, n.d.

[20]SOL Engineering Services, *CF-JET*, Rev Final 2.

[21]SOL Engineering Services, *SOL Aquifer* pipeline diagram.

[22]SOL Engineering Services, *CF-JET*, Rev Final 2.

[23]SOL Engineering Services, *SOL Aquifer* pipeline diagram.

[24]SOL Engineering Services, LLC, “Modeling HVAC System Performance: Code as Architecture Approach to AI,” work carried out with Alabama A&M University students, n.d.

[25]U.S. Army DEVCOM Army Research Laboratory, “Artificial Intelligence and Machine Learning with Extremely Sparse Data (ARL-BAA-0038),” in *Current Research Topics for DEVCOM ARL BAA for Foundational Research*, 9 December 2024, https://files.simpler.grants.gov/opportunities/5e185327-5f25-4c09-889a-43615e1f54da/attachments/e4621f8a-b19c-40ec-8c59-6611cc4fa276/Current_Research_Topics_for_DEVCOM_ARL_BAA_9_Dec2024.pdf.

[26]Manaserh et al., “Multi-Objective Optimization of 3D Printed Liquid Cooled Heat Sink with Guide Vanes.”

[27]Manaserh et al., “Multi-Objective Optimization of 3D Printed Liquid Cooled Heat Sink with Guide Vanes.”

[28]Fallahtafti et al., “Shape Optimization of Hotspot Targeted Micro Pin Fins.”

[29]Fallahtafti et al., “Shape Optimization of Hotspot Targeted Micro Pin Fins.”

[30]Fallahtafti et al., “Shape Optimization of Hotspot Targeted Micro Pin Fins.”

[31]Mahmood et al., “Heat Transfer Modeling and Optimal Thermal Management of Electric Vehicle Battery Systems.”

[32]Mahmood et al., “Heat Transfer Modeling and Optimal Thermal Management of Electric Vehicle Battery Systems.”

[33]Vaillant et al., “Surrogate Model for Heat Transfer Prediction in Impinging Jet Arrays.”

[34]Manaserh et al., “Multi-Objective Optimization of 3D Printed Liquid Cooled Heat Sink with Guide Vanes.”

[35]Fallahtafti et al., “Shape Optimization of Hotspot Targeted Micro Pin Fins.”

[36]Mahmood et al., “Heat Transfer Modeling and Optimal Thermal Management of Electric Vehicle Battery Systems.”

[37]Vaillant et al., “Surrogate Model for Heat Transfer Prediction in Impinging Jet Arrays.”

[38]Ante Sikirica, Luka Grbčić, and Lado Kranjčević, “Machine Learning Based Surrogate Models for Microchannel Heat Sink Optimization,” arXiv:2208.09683 (preprint, version 2, 8 December 2022), https://arxiv.org/abs/2208.09683.

[39]Takiah Ebbs-Picken, David A. Romero, Carlos M. Da Silva, and Cristina H. Amon, “Deep Encoder-Decoder Hierarchical Convolutional Neural Networks for Conjugate Heat Transfer Surrogate Modeling,” arXiv:2311.17068, https://doi.org/10.1016/j.apenergy.2024.123723.

[40]A. Chiodi et al., “A Machine-Learning Based Approach for Multi-Scale Optimisation of Heat Exchangers with Lattice-Like Topology,” *International Journal of Heat and Mass Transfer* 256, part 2 (2026): 128035, https://doi.org/10.1016/j.ijheatmasstransfer.2025.128035.

[41]Sikirica, Grbčić, and Kranjčević, “Machine Learning Based Surrogate Models for Microchannel Heat Sink Optimization.”

[42]Ebbs-Picken et al., “Deep Encoder-Decoder Hierarchical Convolutional Neural Networks.”

[43]“Cold Plate Optimization with Entropy Minimization Using Bayesian Optimization,” in *Proceedings of the ASME 2025 International Design Engineering Technical Conferences and Computers and Information in Engineering Conference* (authors not listed on the retrieved record), https://par.nsf.gov/biblio/10631181.

[44]Sikirica, Grbčić, and Kranjčević, “Machine Learning Based Surrogate Models for Microchannel Heat Sink Optimization.”

[45]Manaserh et al., “Multi-Objective Optimization of 3D Printed Liquid Cooled Heat Sink with Guide Vanes.”

[46]Fallahtafti et al., “Shape Optimization of Hotspot Targeted Micro Pin Fins.”

[47]Mahmood et al., “Heat Transfer Modeling and Optimal Thermal Management of Electric Vehicle Battery Systems.”

[48]Fallahtafti et al., “Shape Optimization of Hotspot Targeted Micro Pin Fins.”

[49]scikit-learn developers, *scikit-learn* repository.

[50]Carl Edward Rasmussen and Christopher K. I. Williams, *Gaussian Processes for Machine Learning* (Cambridge, MA: MIT Press, 2006).

[51]Ebbs-Picken et al., “Deep Encoder-Decoder Hierarchical Convolutional Neural Networks.”

[52]Balaji Lakshminarayanan, Alexander Pritzel, and Charles Blundell, “Simple and Scalable Predictive Uncertainty Estimation Using Deep Ensembles,” *Advances in Neural Information Processing Systems* 30 (2017): 6402–6413.

[53]Volodymyr Kuleshov, Nathan Fenner, and Stefano Ermon, “Accurate Uncertainties for Deep Learning Using Calibrated Regression,” *Proceedings of Machine Learning Research* 80 (2018): 2796–2804.

[54]Anastasios N. Angelopoulos and Stephen Bates, “Conformal Prediction: A Gentle Introduction,” *Foundations and Trends in Machine Learning* 16, no. 4 (2023): 494–591, https://doi.org/10.1561/2200000101.

[55]Snoek and Nado, “Can You Trust Your Model’s Uncertainty?”

[56]Rasmussen and Williams, *Gaussian Processes for Machine Learning*.

[57]Lakshminarayanan, Pritzel, and Blundell, “Deep Ensembles.”

[58]Kuleshov, Fenner, and Ermon, “Accurate Uncertainties for Deep Learning.”

[59]Angelopoulos and Bates, “Conformal Prediction.”

[60]Marc C. Kennedy and Anthony O’Hagan, “Bayesian Calibration of Computer Models,” *Journal of the Royal Statistical Society: Series B* 63, no. 3 (2001): 425–464, https://doi.org/10.1111/1467-9868.00294.

[61]Snoek and Nado, “Can You Trust Your Model’s Uncertainty?”

[62]Marc C. Kennedy and Anthony O’Hagan, “Predicting the Output from a Complex Computer Code When Fast Approximations Are Available,” *Biometrika* 87, no. 1 (2000): 1–13, https://doi.org/10.1093/biomet/87.1.1.

[63]Alexander S. Rattner, “General Characterization of Jet Impingement Array Heat Sinks with Interspersed Fluid Extraction Ports for Uniform High-Flux Cooling,” *Journal of Heat Transfer* 139, no. 8 (August 2017): 082201, https://doi.org/10.1115/1.4036090.

[64]SOL Engineering Services, *CF-JET*, Rev Final 2.

[65]Chiodi et al., “Multi-Scale Optimisation of Heat Exchangers with Lattice-Like Topology.”

[66]Chiodi et al., “Multi-Scale Optimisation of Heat Exchangers with Lattice-Like Topology.”

[67]Chiodi et al., “Multi-Scale Optimisation of Heat Exchangers with Lattice-Like Topology.”

[68]Souza Lima et al., “Multi-Fidelity Bayesian Optimization Framework for CFD-Based Non-Premixed Burner Design.”

[69]Robledo et al., “Active Multi-Fidelity Surrogate Learning for Airfoil Shape Optimization.”

[70]Yanfang Lyu et al., “Multi-Fidelity Prediction of Fluid Flow and Temperature Field Based on Transfer Learning Using Fourier Neural Operator,” arXiv:2304.06972 (preprint, submitted 14 April 2023), https://arxiv.org/abs/2304.06972.

[71]Vaillant et al., “Surrogate Model for Heat Transfer Prediction in Impinging Jet Arrays.”

[72]Kennedy and O’Hagan, “Predicting the Output from a Complex Computer Code.”

[73]Chiodi et al., “Multi-Scale Optimisation of Heat Exchangers with Lattice-Like Topology.”

[74]Souza Lima et al., “Multi-Fidelity Bayesian Optimization Framework for CFD-Based Non-Premixed Burner Design.”

[75]Robledo et al., “Active Multi-Fidelity Surrogate Learning for Airfoil Shape Optimization.”

[76]Lyu et al., “Multi-Fidelity Prediction Using Fourier Neural Operator.”

[77]Vaillant et al., “Surrogate Model for Heat Transfer Prediction in Impinging Jet Arrays.”

[78]Chiodi et al., “Multi-Scale Optimisation of Heat Exchangers with Lattice-Like Topology.”

[79]Souza Lima et al., “Multi-Fidelity Bayesian Optimization Framework for CFD-Based Non-Premixed Burner Design.”

[80]Robledo et al., “Active Multi-Fidelity Surrogate Learning for Airfoil Shape Optimization.”

[81]Kennedy and O’Hagan, “Bayesian Calibration of Computer Models.”

[82]Manaserh et al., “Multi-Objective Optimization of 3D Printed Liquid Cooled Heat Sink with Guide Vanes.”

[83]Mahmood et al., “Heat Transfer Modeling and Optimal Thermal Management of Electric Vehicle Battery Systems.”

[84]Fallahtafti et al., “Shape Optimization of Hotspot Targeted Micro Pin Fins.”

[85]Paleyes et al., “Batch Multi-Objective Bayesian Optimisation in Heat Exchanger Design.”

[86]Manaserh et al., “Multi-Objective Optimization of 3D Printed Liquid Cooled Heat Sink with Guide Vanes.”

[87]Fallahtafti et al., “Shape Optimization of Hotspot Targeted Micro Pin Fins.”

[88]Mahmood et al., “Heat Transfer Modeling and Optimal Thermal Management of Electric Vehicle Battery Systems.”

[89]Sikirica, Grbčić, and Kranjčević, “Machine Learning Based Surrogate Models for Microchannel Heat Sink Optimization.”

[90]Paleyes et al., “Batch Multi-Objective Bayesian Optimisation in Heat Exchanger Design.”

[91]“Cold Plate Optimization with Entropy Minimization Using Bayesian Optimization.”

[92]Chiodi et al., “Multi-Scale Optimisation of Heat Exchangers with Lattice-Like Topology.”

[93]Samuel Daulton, Maximilian Balandat, and Eytan Bakshy, “Parallel Bayesian Optimization of Multiple Noisy Objectives with Expected Hypervolume Improvement,” *Advances in Neural Information Processing Systems* 34 (2021): 2187–2200.

[94]Maximilian Balandat et al., “BoTorch: A Framework for Efficient Monte-Carlo Bayesian Optimization,” *Advances in Neural Information Processing Systems* 33 (2020).

[95]Paleyes et al., “Batch Multi-Objective Bayesian Optimisation in Heat Exchanger Design.”

[96]Li et al., “Fourier Neural Operator.”

[97]Lu Lu et al., “Learning Nonlinear Operators via DeepONet Based on the Universal Approximation Theorem of Operators,” *Nature Machine Intelligence* 3, no. 3 (2021): 218–229, https://doi.org/10.1038/s42256-021-00302-5.

[98]“Neural Operator Models for Predicting Physical Fields in Server Electronic Microchips Doped with Water-Al2O3 Nanofluid Coolant,” *Case Studies in Thermal Engineering*, March 2024 (authors, volume and DOI not retrieved).

[99]“Neural Operator Models for Predicting Physical Fields in Server Electronic Microchips.”

[100]Ebbs-Picken et al., “Deep Encoder-Decoder Hierarchical Convolutional Neural Networks.”

[101]Lyu et al., “Multi-Fidelity Prediction Using Fourier Neural Operator.”

[102]Li et al., “Fourier Neural Operator.”

[103]Lu et al., “Learning Nonlinear Operators via DeepONet.”

[104]“Neural Operator Models for Predicting Physical Fields in Server Electronic Microchips.”

[105]Lyu et al., “Multi-Fidelity Prediction Using Fourier Neural Operator.”

[106]Ebbs-Picken et al., “Deep Encoder-Decoder Hierarchical Convolutional Neural Networks.”

[107]Trampe et al., “Laser-Optical Validation and Comparative Analysis of Numerical Heat Transfer Models.”

[108]Trampe et al., “Laser-Optical Validation and Comparative Analysis of Numerical Heat Transfer Models.”

[109]Trampe et al., “Laser-Optical Validation and Comparative Analysis of Numerical Heat Transfer Models.”

[110]Gurgul and Fornalik-Wajs, “Heat Transfer Performance of RANS Turbulence Models.”

[111]Neil Zuckerman and Noam Lior, “Jet Impingement Heat Transfer: Physics, Correlations, and Numerical Modeling,” *Advances in Heat Transfer* 39 (2006): 565–631, sec. III.B.6, https://doi.org/10.1016/S0065-2717(06)39006-5.

[112]Ali Chitsazan, Georg Klepp, and Birgit Glasmacher, “Numerical Prediction of the Second Peak in the Nusselt Number Distribution from an Impinging Round Jet,” *International Journal of Heat and Technology* 39, no. 4 (2021): 1243–1252, https://doi.org/10.18280/ijht.390422.

[113]Sreekant Narumanchi, “Advanced Liquid Cooling R&D,” presentation, project APE039, National Renewable Energy Laboratory, U.S. Department of Energy Vehicle Technologies Program, 14 May 2013, https://www.energy.gov/eere/vehicles/articles/advanced-liquid-cooling-rd.

[114]Trampe et al., “Laser-Optical Validation and Comparative Analysis of Numerical Heat Transfer Models.”

[115]Trampe et al., “Laser-Optical Validation and Comparative Analysis of Numerical Heat Transfer Models.”

[116]Trampe et al., “Laser-Optical Validation and Comparative Analysis of Numerical Heat Transfer Models.”

[117]Gurgul and Fornalik-Wajs, “Heat Transfer Performance of RANS Turbulence Models.”

[118]Zuckerman and Lior, “Jet Impingement Heat Transfer,” sec. III.B.8.

[119]Chitsazan, Klepp, and Glasmacher, “Second Peak in the Nusselt Number Distribution.”

[120]Narumanchi, “Advanced Liquid Cooling R&D.”

[121]American Society of Mechanical Engineers, *Standard for Verification and Validation in Computational Fluid Dynamics and Heat Transfer*, ASME V&V 20-2009 (R2016) (New York: ASME, 2009), https://webstore.ansi.org/Standards/ASME/ASME202009R2016.

[122]Derived in this paper. The predicted convective resistance is R_cfd = 54/(1 + e) when the hardware value is 54 mm²·K/W, and the predicted total is 18 + R_cfd. The margin is m = 1 − (18 + R_cfd)/72. For conduction of 36 mm²·K/W the same form gives 36 + 36/(1 + e). The calculation assumes that the error acts on the convective term alone and that the predicted conduction term is exact.

[123]scikit-learn developers, *scikit-learn* repository.

[124]GPyTorch developers, *gpytorch* package page.

[125]BoTorch developers, *botorch*, Python Package Index, accessed 23 November 2026, https://pypi.org/project/botorch/.

[126]Balandat et al., “BoTorch.”

[127]PyTorch contributors, *torch*, Python Package Index, accessed 23 November 2026, https://pypi.org/project/torch/.

[128]Iterative, *DVC* README.

[129]MLflow project, “MLflow Documentation Overview,” accessed 23 November 2026, https://mlflow.org/llms.txt.

[130]Open Source Initiative, “The 3-Clause BSD License,” accessed 23 November 2026, https://opensource.org/license/bsd-3-clause.

[131]Open Source Initiative, “The MIT License,” accessed 23 November 2026, https://opensource.org/license/mit.

[132]Apache Software Foundation, “Apache License, Version 2.0,” January 2004, https://www.apache.org/licenses/LICENSE-2.0.

[133]scikit-learn developers, *scikit-learn* repository.

[134]Open Source Initiative, “3-Clause BSD License.”

[135]GPyTorch developers, *gpytorch* package page.

[136]Open Source Initiative, “MIT License.”

[137]BoTorch developers, *botorch* package page.

[138]PyTorch contributors, *torch* package page.

[139]Iterative, *DVC* README.

[140]Apache Software Foundation, “Apache License, Version 2.0.”

[141]MLflow project, “Documentation Overview.”

[142]SOL Engineering Services, *SOL Aquifer* pipeline diagram.

[143]MLflow project, “Documentation Overview.”

[144]Iterative, *DVC* README.

[145]Fallahtafti et al., “Shape Optimization of Hotspot Targeted Micro Pin Fins.”

[146]Mahmood et al., “Heat Transfer Modeling and Optimal Thermal Management of Electric Vehicle Battery Systems.”

[147]Vaillant et al., “Surrogate Model for Heat Transfer Prediction in Impinging Jet Arrays.”

[148]Gurgul and Fornalik-Wajs, “Heat Transfer Performance of RANS Turbulence Models.”

[149]Trampe et al., “Laser-Optical Validation and Comparative Analysis of Numerical Heat Transfer Models.”

[150]Chiodi et al., “Multi-Scale Optimisation of Heat Exchangers with Lattice-Like Topology.”

[151]Ebbs-Picken et al., “Deep Encoder-Decoder Hierarchical Convolutional Neural Networks.”

[152]ASME, *V&V 20-2009*.

[153]Daulton, Balandat, and Bakshy, “Parallel Bayesian Optimization of Multiple Noisy Objectives.”

[154]Balandat et al., “BoTorch.”

[155]SOL Engineering Services, *SOL Aquifer* pipeline diagram.