Modeling Nickel-Silicidation using Physics-Informed Machine Learning

Christopher Straub
Department Modeling and AI
Fraunhofer IISB
Erlangen, Germany
christopher.straub@iisb.fraunhofer.de

Simon Mundinar
Department Modeling and AI
Fraunhofer IISB
Erlangen, Germany
simon.mundinar@iisb.fraunhofer.de

Jessica Klonnek
Department Modeling and AI
Fraunhofer IISB
Erlangen, Germany
jessica.klonnek@iisb.fraunhofer.de

Christophe Pixius
Department Modeling and AI
Fraunhofer IISB
Erlangen, Germany
christophe.pixius@iisb.fraunhofer.de

Andreas Rosskopf
Department Modeling and AI
Fraunhofer IISB
Erlangen, Germany
andreas.rosskopf@iisb.fraunhofer.de

***Abstract*—We simulate nickel silicidation in one and two space dimensions via physics-informed machine learning. Our machine learning models are solely trained on the governing physical laws in the form of a reaction-diffusion system, without requiring measurement or simulation data. In 1D, our model yields accurate predictions across a parametric temperature range. The 2D process is well approximated away from irregular domain features. Compared to classical state of the art simulations, our models achieve speedups of three orders of magnitude. We further discuss potential extensions to the approach, including the incorporation of measurement data for calibration purposes and enabling broader applicability to process optimization tasks.**

***Index Terms*—physics-informed machine learning, reaction-diffusion system, silicidation**

I. Introduction and Model

Ohmic contacts between metal and semiconductor materials are crucial for semiconductor technology, particularly for reducing contact resistance in miniaturization efforts. These contacts are typically formed through silicidation, where a thin metal layer is deposited on the silicon surface, and a silicide layer is then created during annealing. Nickel is commonly used for ohmic contacts on n-type silicon, but various nickel-silicide phases exist, some of which do not yield ohmic contacts. The phase that emerges, and consequently also the quality of the resulting ohmic contact, depends on the annealing temperature, duration, and the ratio of Ni to Si atoms [1]. Numerical modeling of this process can therefore play a significant role in optimizing the performance of the resulting device.

A continuum model based on a reaction-diffusion system for the multi-phase Ni silicidation was proposed in [2]. Since this model is implemented in commercial TCAD tools, such as the Sentaurus software from Synopsys [3], it will be used as the basis of this work. The following reactions are implemented: $2\mathrm{Ni} + \mathrm{Si} \to \mathrm{Ni_2Si}$, $\mathrm{Ni_2Si} + \mathrm{Si} \to 2\mathrm{NiSi}$, $\mathrm{NiSi} + \mathrm{Si} \to \mathrm{NiSi_2}$, with associated reaction constants k_1, k_2, k_3, respectively. The diffusing species are Si and Ni with the effective diffusion constants D_{Si} and D_{Ni}, respectively. Both the diffusion and reaction constants are given by Arrhenius laws

$$D_X(T) = D_{X,0}\mathrm{e}^{-E_X/k_\mathrm{B}T}, \tag{1}$$

$$k_i(T) = k_{i,0}\mathrm{e}^{-E_i/k_\mathrm{B}T}. \tag{2}$$

The model then takes the following form:

$$\frac{\partial C_{\mathrm{Ni}}}{\partial t} = \nabla\cdot(D_{\mathrm{Ni}}\nabla C_{\mathrm{Ni}}) - 2k_1\left(\frac{C_{\mathrm{Ni}}}{C_{0,\mathrm{Ni}}}\right)^2 C_{\mathrm{Si}}, \tag{3}$$

$$\frac{\partial C_{\mathrm{Si}}}{\partial t} = \nabla\cdot(D_{\mathrm{Si}}\nabla C_{\mathrm{Si}}) - k_1\left(\frac{C_{\mathrm{Ni}}}{C_{0,\mathrm{Ni}}}\right)^2 C_{\mathrm{Si}} - k_2\left(\frac{C_{\mathrm{Ni_2Si}}}{C_{0,\mathrm{Ni_2Si}}}\right) C_{\mathrm{Si}} - k_3\left(\frac{C_{\mathrm{NiSi}}}{C_{0,\mathrm{NiSi}}}\right) C_{\mathrm{Si}}, \tag{4}$$

$$\frac{\partial C_{\mathrm{Ni_2Si}}}{\partial t} = k_1\left(\frac{C_{\mathrm{Ni}}}{C_{0,\mathrm{Ni}}}\right)^2 C_{\mathrm{Si}} - k_2\left(\frac{C_{\mathrm{Ni_2Si}}}{C_{0,\mathrm{Ni_2Si}}}\right) C_{\mathrm{Si}}, \tag{5}$$

$$\frac{\partial C_{\mathrm{NiSi}}}{\partial t} = 2k_2\left(\frac{C_{\mathrm{Ni_2Si}}}{C_{0,\mathrm{Ni_2Si}}}\right) C_{\mathrm{Si}} - k_3\left(\frac{C_{\mathrm{NiSi}}}{C_{0,\mathrm{NiSi}}}\right) C_{\mathrm{Si}}, \tag{6}$$

$$\frac{\partial C_{\mathrm{NiSi_2}}}{\partial t} = k_3\left(\frac{C_{\mathrm{NiSi}}}{C_{0,\mathrm{NiSi}}}\right) C_{\mathrm{Si}}. \tag{7}$$

As initial conditions we impose vanishing initial concentrations for all silicide phases and set $C_{\mathrm{Ni}}(\mathbf{x}, 0) = C_{0,\mathrm{Ni}}$ within the geometry of the Ni layer and $C_{\mathrm{Si}}(\mathbf{x}, 0) = C_{0,\mathrm{Si}}$ within the Si domain. Here, $C_{0,X}$ denotes the equilibrium concentration of phase X, which is obtained from its respective density. At the spatial boundaries of the simulation domain we employ vanishing Neumann boundary conditions for the diffusing species to resemble reflecting boundaries. For the constants $D_{\mathrm{Si},0}$, $D_{\mathrm{Ni},0}$, E_{Ni}, E_{Si}, k_1, k_2, k_3, E_1, E_2, E_3, and $C_{0,X}$ we use the default values from Synopsys Sentaurus [3].

The common way of simulating such reaction-diffusion systems is by employing numerical solvers based on finite

This work was supported by the Fraunhofer Internal Programs under Grant No. PREPARE 40-08394.

element methods (FEM). These methods are well-understood and yield accurate simulations of the silicidation process provided that the discretization is sufficiently fine. However, they are often computationally expensive, in particular in the case of 2D or 3D spatial domains. This becomes a problem especially for calibration and optimization tasks, where the simulations need to be run a lot of times to match given (measurement) data and to find the ideal set of parameters for the respective application.

We, therefore, use an alternative approach to solving the reaction-diffusion system based on neural networks (NN), which will be introduced in more detail in Section II. We will apply this approach to two settings which will be presented in Section III: (i) A one-dimensional (1D) problem with a sharp initial boundary between the silicon and the nickel regions and (ii) a two-dimensional (2D) problem with a thin, rectangular nickel region placed on top of a wider silicon region, thus representing the edge of a deposited nickel layer. Both settings are used to assess the performance of the NN-based approach in comparison to the FEM solver of the Sentaurus process simulation tool [3] in Section IV. In Section V we present our conclusions.

II. Method – Physics-informed machine learning

A novel approach to simulate physical systems is provided by *physics-informed machine learning* [4]. In contrast to classical machine learning techniques, this approach directly uses the governing physical laws in the training process instead of, or in addition to, data from simulations or experiments. The training is applied to a NN that approximates the solution of the physical process [5]. If the system parameters are fixed, such NN is referred to as a *physics-informed neural network (PINN)*. In addition, the NN can take system parameters as inputs to map general configurations to the resulting solution of the system. Such NN is referred to as a *physics-informed neural operator (PINO)*. Although the training of PINNs and PINOs can be computationally rather expensive, their evaluation is typically several orders of magnitude faster compared to classical numerical approaches. Hence, a PINO can be used as a surrogate model and can, e.g., be implemented in optimization pipelines.

Physics-informed machine learning has already been successfully used across various applications [6]. In this paper, we present a PINN and a PINO that approximate the silicidation process. The core of these models are NNs with trainable weights. Similar to the system's solution, their input dimensions are the spatio-temporal variables $(\mathbf{x}, t)$ and, for the PINO, parameters determining the temperature profile. The NN outputs are the values of all concentrations at $(\mathbf{x}, t)$. For the NN architectures, we use generic multilayer perceptrons here. For an overview of alternative architectures and a comparison of their performances for a related silicidation problem we refer to [7]. The physical laws have been reformulated in a loss function whose minimization is equivalent to solving the PDE system (3)–(7), the initial conditions, and the boundary conditions [5]. This minimization problem is the models' training task. In particular, no data, e.g., from classical simulations or measurements, is used for the training process. All models presented in this work have been implemented in the Python-based library DeepXDE [8].

III. Settings

We consider the following two settings in this work.

A. 1D setting with parametric temperature

In the first setting, the initial Ni-Si interface is assumed to be a straight line at depth $200\,\mathrm{nm}$. This allows us to assume homogeneity in the other space dimensions and hence to reduce the effective spatial dimension of the system to one. The initial Si-depth is assumed to be $800\,\mathrm{nm}$, resulting in a total domain length of $1\,\mu\mathrm{m}$. This simple one-dimensional case models the silicidation process far away from the edges of a deposited Ni layer. The temperature profile is assumed to be homogeneous with a value between $300\,^\circ\mathrm{C}$ and $700\,^\circ\mathrm{C}$. We train a PINO over this temperature domain. To ensure the correct behavior at the spatial boundary and to simplify the training task, the reflecting boundary conditions are directly incorporated in the PINO's structure by suitably transforming its input as described in [9]. Additional transformations are applied to the PINO's output to ensure that all concentrations are non-negative and that the Ni and Si concentrations are bounded from above by $C_{0,\mathrm{Ni}}$ and $C_{0,\mathrm{Si}}$, respectively. The fully connected NN inside the PINO is composed of 3×50 hidden units.

B. 2D setting with fixed temperature

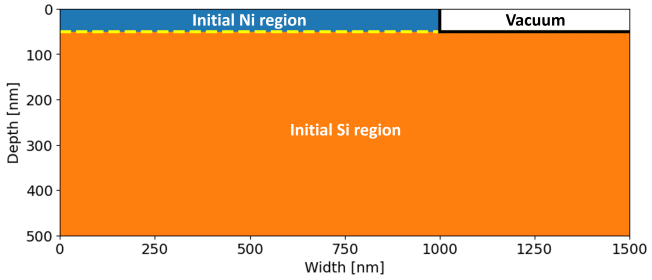

Fig. 1. Initial Ni (blue) and Si (orange) regions in the 2D setting with sharp boundary (dashed yellow line) and the time-invariant vacuum region (white).

In real-world applications, the 1D setting falls short because it is unable to model the behavior of deposited Ni layers whose width is comparable to their thickness. To take into account the influence of edge regions, we consider a thin rectangular Ni block of depth $50\,\mathrm{nm}$ and width $1\,\mu\mathrm{m}$ that is placed on top of a wider Si region of width $1.5\,\mu\mathrm{m}$ and depth $450\,\mathrm{nm}$. A visualization of this configuration is given in Fig. 1. The reduced width of the initial Ni region leads to a rectangular vacuum region. In the model, we assume this vacuum region to be invariant during the silicidation process by considering the spatial computational domain $(x, y) \in ([50\,\mathrm{nm}, 500\,\mathrm{nm}] \times [0\,\mu\mathrm{m}, 1.5\,\mu\mathrm{m}]) \cup ([0\,\mathrm{nm}, 50\,\mathrm{nm}] \times [0\,\mu\mathrm{m}, 1\,\mu\mathrm{m}])$ for all times.

To ensure that no particles can flow into the vacuum region, vanishing Neumann boundary conditions are imposed at the vacuum interface. We emphasize that this irregular, L-shaped spatial domain together with the sharp initial boundary between Ni and Si lead to a high complexity of the system from both a computational and theoretical perspective. The temporal domain is determined by the maximum annealing time for which we have selected the value $t = 2\,\mathrm{min}$; this time has been found to be sufficient for capturing the silicidation of the relatively thin Ni layer. In this 2D setting, we train a PINN for the fixed temperature value $T = 500°\mathrm{C}$, i.e., without a parametric temperature dependence. The PINN is composed of 3×75 hidden units and transformations similar to the 1D setting are applied to restrict the PINN's output range and to ensure that the reflecting boundary conditions at the outer boundaries $x = 0, 50\,\mathrm{nm}$ and $y = 0, 1.5\,\mu\mathrm{m}$ are satisfied.

IV. Results

To demonstrate the capabilities and limitations of physics-informed machine learning to approximate silicidation processes, we now analyze its performance in the two settings introduced above. To assess the accuracies of the models, we compare them to the results of Sentaurus Process, which uses a state of the art adaptive time-stepping FEM solver.

A. 1D setting with parametric temperature

The PINO in the 1D setting is trained by a combination of the Adam and the L-BFGS optimizer for about $4\,\mathrm{h}$ on an NVIDIA Quadro RTX 5000 (16GB RAM). For the reference solver, we chose a uniformly spaced grid with step size $2\,\mathrm{nm}$. For an annealing time of $5\,\mathrm{min}$, the reference solver needs about $20\,\mathrm{s}$ of simulation time on a single core.

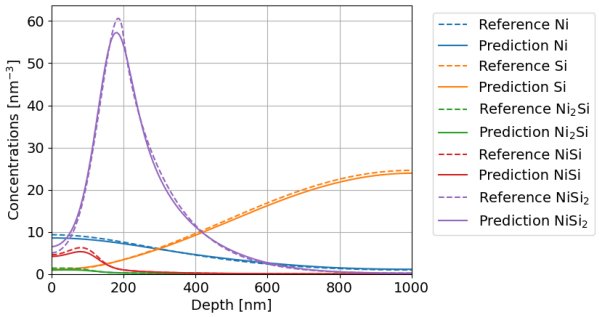

Fig. 2. Predictions of the physics-informed neural operator (PINO) at annealing time $t = 2\,\mathrm{min}$ and temperature $T = 500°\mathrm{C}$ compared to the reference simulation in the 1D setting.

TABLE I
Accuracy and speedup of the PINO approach in the 1D setting.

Annealing time	10 s	1 min	2 min	5 min
Mean L^2-error ($T = 500°\mathrm{C}$)	4.48%	5.24%	6.78%	7.31%
Speedup factor evaluation	2500	2750	2800	2875

The PINO's predictions at $t = 2\,\mathrm{min}$ for $T = 500°\mathrm{C}$ together with the simulation data from the reference solver in the 1D setting are display in Fig. 2. It can be seen that the PINO's predictions are in good agreement with the reference solution. The main difference at $\sim 200\,\mathrm{nm}$ in the Ni profile is due to the step of concentrations at the Ni-Si interface. The mean relative L^2-distances between PINO predictions and reference simulations at different times are given in Tab. I (omitting concentrations with L^2-norms $< 0.1\,\mathrm{nm}^{-3}$). We restrict the comparison to $T = 500°\mathrm{C}$; similar error values can be observed for general temperatures between $300°\mathrm{C}$ and $700°\mathrm{C}$. From an application's point of view, the achieved accuracy is sufficient because the PINO's errors are significantly smaller than the estimated discrepancy between the model and the actual physical process. In addition, the PINO respects the conservation law of total Ni and Si concentrations up to relative errors of below 1%. Lastly, we emphasize that despite the computationally expensive training, evaluating the trained model for general temperatures is computationally very cheap: Evaluating the PINO for an arbitrary temperature on a spatial grid of step size $2\,\mathrm{nm}$ for 31 timesteps (once every $10\,\mathrm{s}$ for the annealing time $t = 5\,\mathrm{min}$) takes only a few milliseconds on an ordinary laptop's CPU, resulting in the speedup factors stated in Tab. I compared to the classical solver.

B. 2D setting with fixed temperature

In the 2D setting, the PINN is trained similarly to the 1D setting for about $6\,\mathrm{h}$. For the reference solver, we use a mesh with cell sizes about $5\,\mathrm{nm} \times 5\,\mathrm{nm}$, resulting in a simulation time of about $240\,\mathrm{s}$ on a single core.

TABLE II
Accuracy and speedup of the PINN approach in the 2D setting.

Annealing time	10 s	1 min	2 min
Mean L^2-error	26.62 %	19.87 %	14.90 %
Speedup factor evaluation	1989	2320	2652

The PINN's predictions at the annealing times $t = 10\,\mathrm{s}, 2\,\mathrm{min}$ together with the simulation data from the reference solver in the 2D setting are display in Fig. 3. Significant differences can be seen close to the sharp edge of the vacuum region, already at the annealing time $t = 10\,\mathrm{s}$. These differences are due to the fact that the basic PINN used here has difficulties to simultaneously approximate the sharp initial Ni-Si boundary and the relatively fast reaction-diffusion process close to the irregular vacuum interface. Away from this region, the concentration values of the PINN and the reference solution are in better agreement. Similar to the 1D setting, the concentrations of Ni_2Si and NiSi are very low compared to $NiSi_2$, which is why we only show the latter in Fig. 3. The error values at different times are stated in Tab. II (computed similarly to the 1D setting). The conservation law for Ni and Si is satisfied up to an error of less than 1%. Lastly, we again emphasize the computational efficiency of evaluating the trained PINN: Its evaluation for 13 timesteps (once every $10\,\mathrm{s}$ for the annealing time $t = 2\,\mathrm{min}$) on the same spatial grid as used for the reference simulation takes

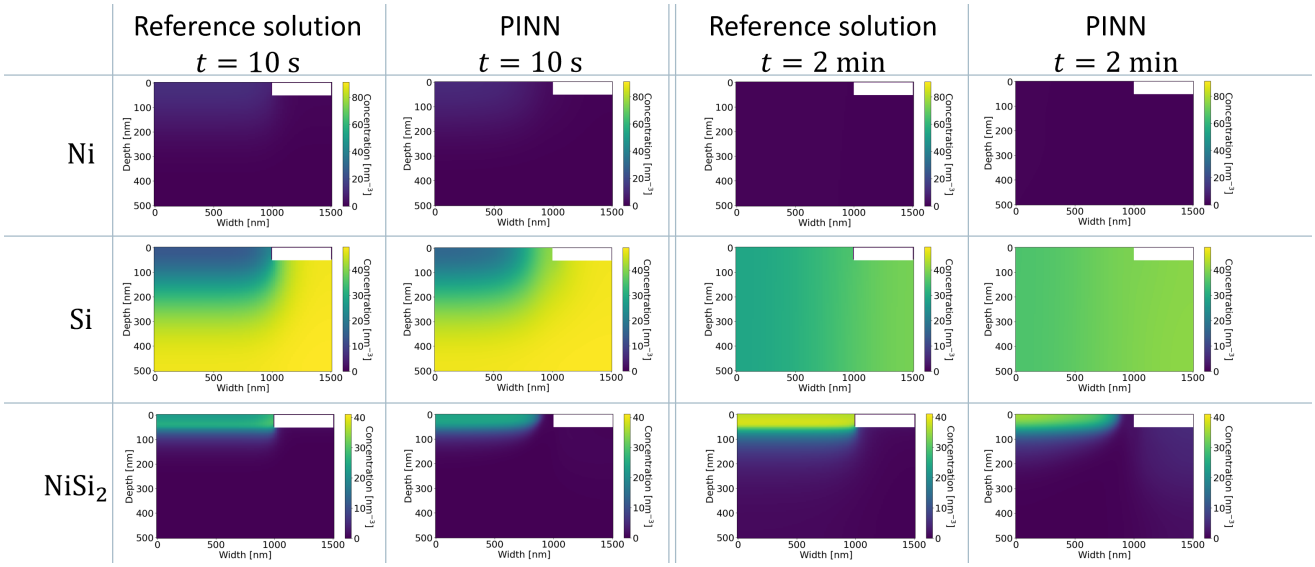

Fig. 3. Predictions of the physics-informed neural network (PINN) at $t = 10\,\mathrm{s}$ and $t = 2\,\mathrm{min}$ for the concentrations of Ni, Si, and the dominating nickel-silicide $NiSi_2$ compared to the reference simulation in the 2D setting. The color ranges are chosen according to the respective equilibrium concentrations.

about one hundred milliseconds on an ordinary laptop's CPU, leading to the speedup factors stated in Tab. II.

V. Conclusion and Outlook

We have implemented and trained neural networks (NNs) that model nickel silicidation in a 1D and 2D setting. In 1D, the comparison with classical simulations show that the NN can indeed capture the process accurately. The model has a parametric temperature dependency, allowing it to predict the silicidation process for general temperatures without retraining. In the 2D setting, the limitations of the PINN are clearly visible close to the sharp edges in the computational domain and the sharp initial interface. Away from these irregular regions, the silicidation process is captured more accurately. To increase the accuracy in this setting, further refinements need to be applied to the basic PINN used here, e.g., to enhance its training process and its ability to approximate the sharp initial Ni-Si boundary; see [6] for an extensive overview of PINN improvement techniques. This is the subject of future follow-up work. Further directions for extending the present work include the addition of a parametric temperature dependence in the 2D setting and a parametric dependence on the initial Ni geometry to arrive at a PINO that can generalize across different initial configurations. One main benefit of such models is their computationally cheap evaluation. Concretely, as indicated by the results from Section IV, speedups of about three orders of magnitude compared to classical state of the art simulations can be expected.

Lastly, we again emphasize that no measurement or simulation data is used for the training here. Instead, the training is based solely on the physical laws describing the process, for which we have used the model from [2]. It should be noted that the parameters used here have not been calibrated and should thus merely be considered as prototype parameters. A major opportunity for physics-informed machine learning is that this calibration can naturally be included in the current approach by adding the approximation of measurement data to the training process and obtain the calibrated parameters via an inverse learning task [2]. Nonetheless, we consider the currently used model a useful prototype to assess the performance of physics-informed machine learning, whose usefulness we see in the process optimization of less established materials, such as SiC. Since process parameters and material properties are less well known for such materials, methods that allow for a fast iteration of simulations can be hugely beneficial.

References

[1] T.B. Massalski, H. Okamoto (Ed.), *Binary alloy phase diagrams*, Materials Park, Ohio: ASM International (1990).

[2] R. Černý, V. Cháb, P. Přikryl, *Numerical simulation of the formation of Ni silicides induced by pulsed lasers*, Comput. Mater. Sci. **4**, 269-281 (1995).

[3] Sentaurus™ *Process User Guide*, Version W-2024.09-SP1, Synopsys Inc., 2024.

[4] G.E. Karniadakis, I.G. Kevrekidis, L. Lu, P. Perdikaris, S. Wang, and L. Yang, *Physics-informed machine learning*, Nat. Rev. Phys. **3**, 422–440 (2021).

[5] M. Raissi, P. Perdikaris, and G.E. Karniadakis, *Physics-informed neural networks: A deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations*, J. Comput. Phys. **378**, 686–707 (2019).

[6] J.D. Toscano, V. Oommen, A.J. Varghese, Z. Zou, N.A. Daryakenari, C. Wu, and G.E. Karniadakis, *From PINNs to PIKANs: Recent Advances in Physics-Informed Machine Learning*, Mach. Learn. Comput. Sci. Eng. **1**, 15 (2025).

[7] A. Rosskopf, C. Straub, and D. Tenbrinck, *Scientific Machine Learning (SciML) – How the fusion of AI and physics is giving rise to promising simulation methodologies*, SISPAD 2025.

[8] L. Lu, X. Meng, Z. Mao, and G.E. Karniadakis, *DeepXDE: A Deep Learning Library for Solving Differential Equations*, SIAM Rev. **63**, 208–228 (2021).

[9] C. Straub, P. Brendel, V. Medvedev, and A. Rosskopf, *Hard-constraining Neumann boundary conditions in physics-informed neural networks via Fourier feature embeddings*, ICLR 2025 Workshop Machine Learning Multiscale Processes (2025).