# Generalizable Prediction of Mixture Etching Rates Using Graph Neural Networks

**Tatiana Boura**[*]
Graz University of Technology
Graz, Austria

Graz Center for Machine Learning
Graz, Austria
tatiana.boura@tugraz.at

**Eneri Boniakou**
School of Chemical Engineering
National Technical University of Athens
Zografou, Greece
eneriboniakou@mail.ntua.gr

**Alex Kondi**
Institute of Nanoscience and Nanotechnology
NCSR "Demokritos"
Ag. Paraskevi, Greece
a.kondi@inn.demokritos.gr

**Vassilios Constantoudis**
Institute of Nanoscience and Nanotechnology
NCSR "Demokritos"
Ag. Paraskevi, Greece
v.constantoudis@inn.demokritos.gr

**George Kokkoris**
School of Chemical Engineering
National Technical University of Athens
Zografou, Greece
gkokkoris@chemeng.ntua.gr

## Abstract

Plasma-based etching and thin-film processing rely on high plasma densities and independent ion energy control to achieve high etching rates and anisotropy. However, accurately predicting spatially varying etching rates remains challenging. Physics-based models (PBMs) can capture the complex plasma dynamics, but they are computationally prohibitive due to the need to solve large systems of partial differential equations, especially when optimization tasks require repeated evaluations. Moreover, plasma processes depend strongly on gas chemistry, reactor configurations, and operating conditions, requiring separate PBMs for each scenario. The complexity is further amplified in the case of gas mixtures. To address this challenge, we propose an architecture that leverages pre-trained single-element neural network predictors, coupled through an inductively learned Graph Neural Network, GraphSAGE, to predict etching rates of mixtures. GraphSAGE enables inference on unseen graphs without retraining, making it possible to extend predictions to new mixtures using only the pool of pre-trained single-element models. We evaluate our approach on a two-gas argon-oxygen mixture, demonstrating promising accuracy and generalization capabilities.

[*]Corresponding author. This work was conducted while the author was affiliated with the Institute of Informatics and Telecommunications, NCSR "Demokritos", Ag. Paraskevi, Greece

39th Conference on Neural Information Processing Systems (NeurIPS 2025) Workshop: AI for Accelerated Materials Design (AI4Mat).

# 1 Introduction

Plasma, often described as the fourth state of matter, consists of charged and neutral particles that are electrically quasi-neutral on average. Low-pressure plasmas are widely used in semiconductor manufacturing for substrate etching and thin-film deposition, accounting for roughly 40–45% of process steps in device fabrication [1]. In plasma etching, electromagnetic fields generate reactive species in the reactor bulk that interact with the substrate; a key performance requirement is etching uniformity across the wafer.

Despite significant advances, predicting local etching rates remains a challenge due to the coupled effects of electromagnetic fields, plasma transport, and volumetric and surface chemistry. Accurate prediction is essential for optimizing plasma processes and understanding how operating parameters—such as power, pressure, gas composition, and substrate temperature—influence etching outcomes.

Physics-based models (PBMs) provide detailed descriptions of these interactions by solving coupled sets of partial differential equations for mass, energy, momentum, and electromagnetic fields. However, their computational cost is prohibitive, especially in optimization workflows where multiple iterations are required [2]. Surrogate models based on machine learning (ML) offer an attractive alternative, mapping operating conditions directly to substrate-scale etching rates and enabling orders-of-magnitude speedups.

In practice, plasma etching involves diverse gas chemistries, reactor designs, and process conditions. Each combination requires a separate PBM and dataset to train an ML predictor, which becomes increasingly expensive for mixtures of gases. By contrast, single-gas PBMs are significantly simpler and less costly to generate. The difference in PBM runtimes, where single elements require significantly less computation, becomes increasingly pronounced as mixtures involve more elements. This motivates our work: complex mixtures demand longer runtimes and greater modeling effort for PBMs. Since the pool of single elements is finite, we propose running single-element PBMs individually and employing a more abstract structure to decouple their interactions. In the context of ML pipelines, this translates to a framework that leverages pre-trained single-gas predictors and learns their interactions rather than treating each gas mixture as a new system from scratch.

Graph Neural Networks (GNNs) are a natural fit for this problem. Individual single-gas predictors can be represented as nodes in a graph, while edges capture their interactions and underlying chemical kinetics. The GNN then learns mixture-level embeddings that map to etching rates. Unlike transductive GNNs such as GCN [3] and GAT [4], which require all nodes to be seen during training, GraphSAGE [5] is inductive: it learns functions that generate embeddings from local neighborhoods, enabling generalization to unseen gases and mixtures.

In this work, we propose a generalizable architecture that integrates pre-trained single-gas neural predictors with GraphSAGE to estimate etching rates of mixtures. This inductive framework allows both inference on unseen mixtures and incorporation of new gases without retraining on the entire dataset. We validate our approach on a two-gas system—oxygen ($O_2$) and argon ($Ar$)—etching a Poly(methyl methacrylate) (PMMA) blanket samples in a low-pressure inductively coupled plasma reactor. Compared against a feedforward neural network (FNN) baseline, our method achieves superior accuracy and shows strong potential for scalable mixture prediction.

# 2 Experimental Setup

## 2.1 Data generation

This work investigates plasma etching of PMMA blanket samples in the Gaseous Electronics Conference (GEC) reference reactor operating in inductively coupled plasma (ICP) mode [6]. The axisymmetric physics-based model (PBM) solves species and electron mass balances, electron energy, momentum, and Poisson/Ampère equations, with coil power and chamber pressure as inputs and etching rates as outputs. Etching is modeled for pure $O_2$, pure Ar, and their mixtures. Pure $O_2$ etching involves ion-enhanced chemistry with 156 reactions and six ion species, while pure Ar etching is dominated by physical sputtering, with 40 reactions and one ion species [6, 7, 8]. In $Ar/O_2$ mixtures, etching arises from both mechanisms: oxygen adsorption followed by ion-enhanced reactions releasing volatile by-products, and argon-driven sputtering [6, 7, 9, 10]. The mixture reaction set includes 209 reactions and 17 species (7 ions). Operating conditions span pressures of 10–70 mTorr, powers

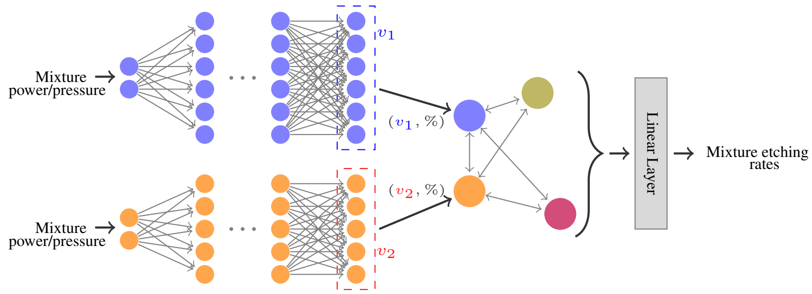


Figure 1: Proposed architecture and data pipeline. Mixture power and pressure inputs are processed by pre-trained single-chemistry networks, and their outputs are combined with gas (element) fractions to form node features in a complete graph. The graph is processed by a GraphSAGE layer followed by a linear layer to predict the etching rates of the mixture.

| *FNN features* | **GraphSAGE** | |
|---|---|---|
| | $\boldsymbol{Test}_{\text{grid}}$ | $\boldsymbol{Test}_{\text{thresh}}$ |
| Output Layer | 0.81 | 0.93 |
| Last Hidden Layer | **0.95** | **0.98** |
| Second to Last Hidden Layer | 0.94 | 0.95 |
| Only First Hidden Layer | 0.85 | 0.97 |

Table 1: Performance of GraphSAGE models on the two test set configurations, evaluated across four different sets of characteristics. The evaluation metric is $R^2$.

of 50–2000 W, and Ar fraction in the mixture of 0.01–0.96, yielding etch rates of 0.1–0.4 $\mu$m/min ($O_2$) and $10^{-4}$– 0.089 $\mu$m/min (Ar). Typical PBM runtimes are ~30 min (Ar), ~45 min ($O_2$), and ~60 min (mixtures).[2]

### 2.2 Single gas neural networks

For each plasma chemistry ($O_2$, Ar), 4000 data points were generated through PBM simulations and split 60-20-20% into training, validation, and testing sets. The test data were drawn from a distinct input subspace to ensure fair evaluation. The predictive models are simple FNNs with two inputs (coil power, pressure) and ten outputs, corresponding to etching rates at radial wafer positions. Each FNN consists of 3 hidden layers and an output layer. The $O_2$ and Ar networks achieve $R^2$ scores of 0.84 and 0.96, respectively, reflecting the higher complexity of the $O_2$ chemistry.

### 2.3 Mixture gas neural networks

#### 2.3.1 Baseline

For the Ar/$O_2$ mixture, 4758 PBM data points are used with the same training regime as the single-element NNs. A FNN with inputs power, pressure, and $x_{Ar}$, the Ar fraction in the mixture, is employed as baseline for etching-rate prediction. Two testing strategies are considered: (i) $Test_{grid}$, where the top 25% of power and pressure values are withheld for testing (4281 train, 297 test), and (ii) $Test_{thresh}$, where only cases with $x_{Ar} > 0.2$ are used for training (3963 train, 615 test). The latter emphasizes conditions dominated by the more complex $O_2$ chemistry, as described in Section 2.1. The baseline model achieves an $R^2$ of 0.93 on $Test_{grid}$ and 0.94 on $Test_{thresh}$.

#### 2.3.2 Model architecture

Figure 1 illustrates the proposed architecture. The normalized power and pressure inputs from the mixture data are first fed into the two pre-trained single-chemistry networks, which remain frozen during mixture prediction. Normalization is performed with a min-max scaler fitted on the mixture data, ensuring all inputs lie within $[0, 1]$, consistent with the expectations of the pre-trained networks.

[2]Hardware: AMD Ryzen Threadripper 3960X, 24 cores, 264GB RAM.

The outputs of each network (or intermediate layer representations, as discussed later) are concatenated with the corresponding mixture composition of each chemistry, forming feature vectors. Each feature vector is associated with a node in the graph, and serves as its node attributes. The graph is complete, since all chemistries in the mixture are expected to interact with one another. In the present case, the graph contains two nodes, corresponding to $O_2$ and Ar. Each node has 11 features: 10 from the network output and one representing the element fraction in the mixture. The nodes are connected by two directed edges, one in each direction, resulting in a two-node complete graph.

This graph is processed by a GraphSAGE layer followed by a ReLU activation. The resulting feature vectors are then passed through a linear layer to predict the 10 etching rates of the mixture. The hidden dimension of the GraphSAGE layer is chosen to keep the number of trainable parameters comparable to the baseline, ensuring a fair evaluation.

#### 2.3.3 Results and Discussion

We evaluate the proposed architecture on two testing sets and investigate different node feature representations by extracting features from intermediate layers of the pre-trained single-gas networks. While the final outputs of these networks reflect single-gas etching rates, they are not directly informative for mixture prediction. Instead, intermediate representations most probably capture richer, chemistry-specific information that the graph model can more effectively combine.

Results are summarized in Table 1, where rows correspond to feature choices: (i) network outputs, (ii) last hidden layer, (iii) second-to-last hidden layer, and (iv) first layer. Our results show that the proposed method outperforms the baseline network across both testing set configurations, with the best performance achieved when the final output layer is omitted. Overall, it outperformed the baseline in 5 out of 8 experiments. These findings support the intuition that leveraging learned features from hidden layers, rather than directly using the outputs of individual networks, leads to improved predictive performance.

## 3 Conclusion and Future Work

We introduced a generalizable architecture for predicting etching rates in plasma mixtures, designed to replace computationally expensive PBMs. The approach leverages pre-trained single-element networks and integrates them through an inductive graph learning framework. Evaluation on $Ar/O_2$ mixtures demonstrates that our method outperforms a conventional neural baseline.

A key strength of the proposed framework is its ability to generalize to unseen mixtures without re-training, requiring only pre-trained single-element predictors–a far smaller modeling effort than building full PBMs for every possible mixture. As future work, we plan to extend the study to mixtures involving additional elements to further validate the scalability and generality of our approach.


## Acknowledgments and Disclosure of Funding

This work is supported by the project “Enhancing plasma etching efficiency, repeatability, and environmental footprint via AI-based modeling and optimization (plasmAI)” of the program “AI-Aware Pathways to Sustainable Semiconductor Process and Manufacturing Technologies”, Intel Corporation & Merck KGaA.


## References


[1] Graves, D.B., Labelle, C.B., Kushner, M.J., Aydil, E.S., Donnelly, V.M., Chang, J.P., Mayer, P., Overzet, L., Shannon, S., Rauf, S. & Ruzic, D.N. (2024). Science challenges and research opportunities for plasma applications in microelectronics. In *Journal of Vacuum Science & Technology B*.

[2] Toscano, J.D., Oommen, V., Varghese, A.J., Zou, Z., Daryakenari, N.A., Wu, C. & Karniadakis, G.E. (2024). From PINNs to PIKANs: Recent Advances in Physics-Informed Machine Learning. In *Machine Learning for Computational Science and Engineering*.

[3] Kipf, T.N. & Welling, M. (2017) Semi-supervised classification with graph convolutional networks. In *ICLR*.

[4] Veličković, P., Cucurull, G., Casanova, A., Romero, A., Liò, P. & Bengio, Y. (2018) Graph attention networks. In *ICLR*.

[5] Hamilton, W., Ying, Z. & Leskovec, J. (2017) Inductive representation learning on large graphs. In *NeurIPS*.

[6] S. Mouchtouris & G. Kokkoris (2016). A hybrid model for low pressure inductively coupled plasmas combining a fluid model for electrons with a plasma-potential-dependent energy distribution and a fluid-Monte Carlo model for ions. In *Plasma Sources Science and Technology*.

[7] G. Memos, E. Lidorikis, E. Gogolides & G. Kokkoris (2021). A hybrid modeling framework for the investigation of surface roughening of polymers during oxygen plasma etching. In *Journal of Physics D: Applied Physics*.

[8] S. Spyrides, F. Alencastro, E. Guimaraes, F. Bastian & R. Simão (2019). Mechanism of oxygen and argon low pressure plasma etching on polyethylene (UHMWPE). In *Surface and Coatings Technology*.

[9] J. T. Gudmundsson & E. G. Thorsteinsson (2007). Oxygen discharges diluted with argon: dissociation processes. In *Plasma Sources Science and Technology*.

[10] D. A. Toneli, R. S. Pessoa, M. Roberto & J. T. Gudmundsson (2015). On the formation and annihilation of the singlet molecular metastables in an oxygen discharge. In *Journal of Physics D: Applied Physics*.

# A Technical Appendices and Supplementary Material

## A.1 Training and Evaluation Details

### A.1.1 Single element networks

The $O_2$ and Ar networks have approximately $580$ trainable parameters per network. Optimal hyper-parameters were selected via grid search. For the Ar network, the best configuration used a batch size of $4$, learning rate of $2.94 \times 10^{-3}$, and weight decay of $1.74 \times 10^{-6}$. For the $O_2$ network, the optimal settings were a batch size of 32, learning rate of $9.62 \times 10^{-4}$, and weight decay of $2.06 \times 10^{-5}$. Both models were trained for 700 epochs with an early stopping patience of 20.

In the selected testing set for the $O_2$ data the power falls within the range [1340 W, 2000 W] and the pressure falls within [46 mTorr, 70 mTorr]. For the Ar data, the selected power range is [330 W, 500 W], and the pressure range is [39 mTorr, 60 mTorr].

A weighted mean squared error loss was employed to address the large radial variation of etching rates across the wafer. Etching rates near the wafer center can exceed those at the edge by an order of magnitude due to higher species densities. With an unweighted loss, this imbalance prevented the networks from adequately learning the lower-valued edge outputs. By weighting the corresponding loss terms, we ensured balanced contributions from all radial positions.

Regularization was applied through early stopping and L2 weight decay. All input and output data were scaled via min-max normalization to the range $[0, 1]$, with the same procedure applied to mixture data.

### A.1.2 Baseline mixture network

The baseline network is a feedforward NN with an input layer, an output layer, and five hidden layers. Each of the hidden layers and the output layer has a width of 10. The total number of trainable parameters is 590. The selected hyperparameters where: a batch size of 32, learning rate of $9.62 \times 10^{-4}$, and weight decay of $2.06 \times 10^{-5}$.

### A.1.3 GNN mixture network

The network was trained with a learning rate of $0.001$, a batch size of 32 and a weight decay of $2.05 \times 10^{-5}$. Its trainable parameters were 571.

## A.2 Other GNN architecture results

We also conducted experiments to compare the performance of GraphSAGE with two other GNN architectures of similar capacity (GCN and GAT, with the same number of trainable parameters). The results are presented in Table 2. GCN appears unable to capture the task effectively, while

GAT performs well. Although GAT achieves slightly better results on the $Test_{grid}$ configuration, GraphSAGE demonstrates superior performance on average.

| *FNN features* | **GCN** | | **GAT** | | **GraphSAGE** | |
|---|---|---|---|---|---|---|
| | $\boldsymbol{Test}_{\text{grid}}$ | $\boldsymbol{Test}_{\text{thresh}}$ | $\boldsymbol{Test}_{\text{grid}}$ | $\boldsymbol{Test}_{\text{thresh}}$ | $\boldsymbol{Test}_{\text{grid}}$ | $\boldsymbol{Test}_{\text{thresh}}$ |
| Output Layer | 0.10 | **0.55** | 0.58 | 0.92 | 0.81 | 0.93 |
| Last Hidden Layer | **0.27** | 0.51 | 0.95 | **0.96** | **0.95** | **0.98** |
| Second to Last Hidden Layer | 0.26 | 0.46 | 0.97 | 0.91 | 0.94 | 0.95 |
| Only First Hidden Layer | 0.22 | 0.48 | **0.97** | 0.89 | 0.85 | 0.97 |

Table 2: Performance comparison of GCN, GAT, and GraphSAGE models on two test metrics, evaluated across four different sets of characteristics. The evaluation metric is $R^2$. Bold values indicate the best performance within each column.