Research Paper
Computational Methods for Differential Equations
http://cmde.tabrizu.ac.ir
Vol. *, No. *, *, pp. 1-10
DOI:10.22034/cmde.2025.62043.2708

An heuristic method for solving an inverse problem in semiconductors governed by a nonlinear coupled system

Youness El Yazidi[1,*] **and Abdellatif Ellabib**[2]

[1]Research Laboratory in Numerical Analysis and Nonlinear Analysis (LaR2A) Department of Mathematics, Faculty of Sciences, University Abdelmalek Essaadi, BP 2121 M'Hannech II 93030 Tetouan, Morocco.

[2]Laboratory of Applied Mathematics and Computer Science, Faculty of Science and Technology, Cady Ayyad University, Marrakesh, Morocco.

Abstract

In this work, we utilize a coupled system to describe the carrier density in the Metal-Semiconductor Field-Effect Transistor (MESFET) device. To identify the depletion layer in this semiconductor, we define a cost functional J, and then use it to derive the shape optimization problem, for which we prove the existence of a solution. We develop an approach to solve this optimization problem based on the finite element method with particle swarm optimization. Finally, we present several numerical examples to demonstrate the robustness of our proposed algorithm in identifying the depletion layer in the MESFET device.

Keywords. Free boundary problem; Shape optimization; Finite element method; Particle swarm optimization.

2010 Mathematics Subject Classification. 49Q10, 65L60, 90C59.

1. Introduction

Many real-world applications can be modeled as free boundary problems, including the problem of identifying cracks [17], semiconductor theory [5, 7, 8], and others [4, 13, 14, 25]. To solve these types of problems, researchers have developed various numerical methods, including gradient-based methods and non-gradient-based methods [22]. The use of a gradient-based method involves searching for the gradient of the objective function, which includes a shape gradient that can be derived from the solution of an adjoint problem. This adjoint problem is established by manipulating the sensitivity analysis [10, 19].

The MESFET is a type of semiconductor device that contains three electrodes: the source, the drain, and the gate [24]. One of the key characteristics of this type of semiconductor is the presence of a free boundary within the device that separates two regions: the conductivity region and the depletion region. By varying the applied voltages on the semiconductor, the free boundary can be moved to satisfy specific operating properties. The working of a MESFET is governed by two major modes: the depletion mode and the enhancement mode. The depletion mode is realized when a zero voltage is applied to the gate terminal, while the enhancement mode is performed when a negative voltage is applied to the gate.

The free boundary problem in the MESFET device has been studied as a shape optimization problem in previous works [1, 2]. In this paper, we revisit the coupled system that describes the MESFET device [3]. The main contribution of our work is the use of heuristic methods instead of gradient-based methods. Our motivation for using heuristic methods instead of gradient methods is due to their simplicity of implementation and the avoidance of computing the objective functional gradient, which typically involves searching for the adjoint problem via sensitivity analysis. The algorithm we use to identify the depletion layer in the MESFET is a combined finite element solver with the particle swarm optimization algorithm. In this algorithm, the depletion layer is considered as a particle in the swarm of candidate solutions to the shape optimization problem. The numerical simulations demonstrate that the algorithm

Received: 10 June 2024 ; Accepted: 20 January 2025.
∗ Corresponding author. Email: y.elyazidi@uae.ac.ma .

converges to high-quality solutions while also maintaining the functioning properties of the MESFET. Earlier works have also employed heuristic methods, such as in [6, 7] where differential evolution was used to identify the depletion region in a pn junction semiconductor. Level set method [9] has been also an efficient technique to reconstruct free boundaries. In [23], particle swarm optimization was combined with an isogeometric boundary element for structural shape optimization. Particle swarm optimization was also used in [25] for spectrum reconstruction.

This paper is structured as follows. In the next section, we introduce the MESFET device and the coupled system that describes the carrier density in this semiconductor. In section 3, we formulate the shape optimization problem, which can be derived from the coupled system by defining a cost functional. We then establish the necessary estimates to prove the existence of a solution. In section 4, we describe the proposed algorithm, which combines a finite element solver with a particle swarm optimization. Finally, we present some numerical results in section 6 to demonstrate the effectiveness of our approach in identifying the depletion layer in the MESFET, in both the depletion and enhancement modes.

2. Statement of the problem

In this part, we will try to write the shape optimization problem. For that, we first illustrate the geometry of the MESFET in Figure 1. Thereafter we seek the coupled problem that describe the behavior of the carriers density in MESFET.

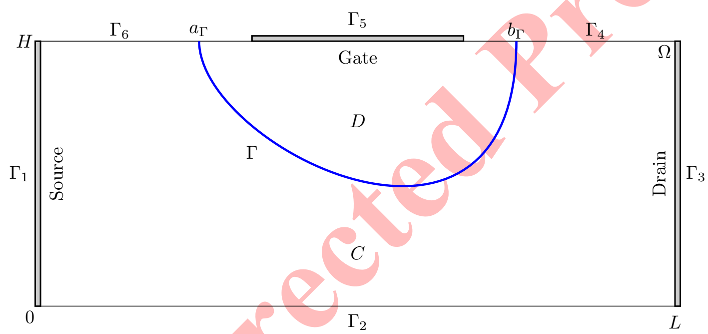

Figure 1. Geometry of MESFET semiconductor.

with Γ_1, Γ_3 and Γ_5 are the drain, the source and the gate terminals respectively. we set $\Gamma_D = \Gamma_1 \cup \Gamma_3 \cup \Gamma_5$ and $\Gamma_N = \Gamma_2 \cup \Gamma_4 \cup \Gamma_6$.

The electrostatic potential in MESFET satisfies the famous drift diffusion model [15, 24]. Basing on this drift diffusion model, we have a coupled system of equations that describes the carriers density in the MESFET, we write the following [15]:

$$\begin{cases} -\Delta u = 0, u = w, & \text{in } C, \\ -\Delta u = \zeta, u < w, & \text{in } D, \\ \dfrac{\partial u}{\partial n} = 0, & \text{on } \Gamma_N, \\ u = u_d, & \text{on } \Gamma_D. \end{cases} \tag{2.1}$$

$$\begin{cases} \nabla(e^{\beta(u-w)}\nabla w) = 0, & \text{in } \Omega, \\ \dfrac{\partial w}{\partial n} = 0, & \text{on } \Gamma_N, \\ w = w_d, & \text{on } \Gamma_D. \end{cases} \tag{2.2}$$

where

$$\begin{cases} u_d = V^+ \text{ on } \Gamma_3,\ V^- \text{ on } \Gamma_5 \text{ and } 0 \text{ on } \Gamma_1, \\ w_d = V^+ \text{ on } \Gamma_3 \cup \Gamma_5 \text{ and } 0 \text{ on } \Gamma_1. \end{cases} \tag{2.3}$$

3. Shape identification problem

First we give some standard notations that will be useful throughout this paper.

Let be the functional space: $H^1_{\Gamma_D}(\Omega) = \{u \in H^1(\Omega),\ \ u = 0 \text{ on } \Gamma_D\}$, where $H^1(\Omega)$ is the standard Sobolev space. We equip $H^1(\Omega)$ and $H^1_{\Gamma_D}(\Omega)$ by their usual norms, $\|\cdot\|_{1,\Omega}$ and $|\cdot|_{1,\Omega}$ respectively.

In [15], Lemma 3.3.14 we have the following:

Remark 3.1. For any (u, w) solution of the coupled system, we get

$$\underline{u_d} \le u \le \overline{u_d} \text{ a.e in } \Omega \tag{3.1}$$

$$\underline{w_d} \le w \le \overline{w_d} \text{ a.e in } \Omega \tag{3.2}$$

where $\underline{v} = \inf_{\Gamma_D} v$ and $\overline{v} = \sup_{\Gamma_D} v$.

Now we establish the variational problems, since the map of $H^1(\Omega)$ in $H^{\frac{1}{2}}(\Gamma_D)$ is surjective, then there exists $\tilde{u}, \tilde{\eta} \in H^1(\Omega)$ such that, $\tilde{u} = u_d$ and $\tilde{w} = w_d$ on Γ_D, thereafter similar to [2], the variational problems are as follows

$$\begin{cases} \text{Find } u \in H^1_{\Gamma_D}(\Omega) \text{ such:} \\ \displaystyle\int_\Omega \nabla u \nabla v dx = \int_D \zeta v dx - \int_\Omega \nabla \tilde{u} \nabla v dx,\ \text{For all } v \in H^1_{\Gamma_D}(\Omega). \end{cases} \tag{3.3}$$

$$\begin{cases} \text{Find } w \in H^1_{\Gamma_D}(\Omega) \text{ such:} \\ \displaystyle\int_\Omega e^{\beta(u-w)} \nabla w \nabla v dx = -\int_\Omega e^{\beta(u-w)} \nabla \tilde{w} \nabla v dx,\ \text{For all } v \in H^1_{\Gamma_D}(\Omega). \end{cases} \tag{3.4}$$

To simplify the second variational form (3.4), we introduce the Slotboom variable [24] $\eta = e^{-\beta w}$, then it become:

$$\begin{cases} \text{Find } \eta \in H^1_{\Gamma_D}(\Omega) \text{ such:} \\ \displaystyle\int_\Omega e^{\beta u} \nabla \eta \nabla v dx = -\int_\Omega e^{\beta u} \nabla \tilde{\eta} \nabla v dx,\ \text{For all } v \in H^1_{\Gamma_D}(\Omega). \end{cases} \tag{3.5}$$

The use of the Slotboom variable $\eta = e^{-\beta w}$, implies the need to show that η lives in $H^1(\Omega)$. From Remark 3.1 implies that $e^{-\beta w(x)} \in L^\infty(\Omega)$, then $\eta \in L^2(\Omega)$, we have $\nabla e^{-\beta w(x)} = -\beta \nabla w(x) e^{-\beta w(x)}$ yields $\nabla \eta \in L^2(\Omega)$, therefore $\eta \in H^1(\Omega)$. We mention that finding η or w is equivalent. For linearity and simplicity reasons, in the numerical simulation we solve problem (3.5) instead of (3.4).

As in [1], we have the next result

Proposition 3.2. *The variational problems* (3.3) *and* (3.4) *have a unique solution in* $H^1_{\Gamma_D}(\Omega)$

Let us recall [2, 3] some estimates on the solutions sequence u_n and w_n of problems (3.3) and (3.4).

Lemma 3.3. *Let u_n be a sequence of solution of problem* (3.3)*, there exists $M_1 > 0$ such that $|u_n|_{1,D} \le M_1$.*

Lemma 3.4. *Let w_n be a solution sequence of problem* (3.4)*, there exists $M_2 > 0$ such that $|w_n|_{1,\Omega} \le M_2$.*

Now we introduce the following cost functional,

$$J(\Gamma) = \frac{1}{2} \int_D \Big[(w(\Gamma) - u(\Gamma))^+\Big]^2 dx + \frac{1}{2} \int_C \Big[u(\Gamma) - w(\Gamma)\Big]^2 dx, \tag{3.6}$$

the first part in J consist to keep the constraint $u < w$ on D hold, the other part is for the constraint $u = w$ on C hold. The fact that Γ is a part of the boundary of D and C (also $\Gamma = \partial D \cap \partial C$), the solutions $u(\Gamma)$ and $w(\Gamma)$ will depend on the geometry of Γ. Let θ_{ad} be the set of the admissible boundaries defined by:

$$\theta_{ad} = \Big\{ \Gamma(\varphi)/\varphi \in \mathcal{C}([0,L]), \exists L_0 > 0, |\ \varphi(x) - \varphi(y)\ | \le L_0\ |\ x - y\ |, 0 \le \varphi(x) \le H \ \ \forall x, y \in [0,L]$$
$$\text{and } \varphi = 0 \text{ on } [0, a_\Gamma] \cup [b_\Gamma, L] \Big\}.$$

This choice of θ_{ad} will guaranty that Γ does not exceed the hight of the semiconductor. It is better to define Γ on a fixed interval $[0, L]$ not the variable one $[a_\Gamma, b_\Gamma]$, hence the condition that Γ must vanish on $[0, a_\Gamma] \cup [b_\Gamma, L]$. In addition the free boundary should be smooth to help with the mathematical analysis of the problem.

The space θ_{ad} must be equipped by a topology, for that we have

Definition 3.5. we supply the space θ_{ad} with the next topology:

$$\Gamma_n \underset{n \longrightarrow \infty}{\longrightarrow} \Gamma, \tag{3.7}$$

the convergence of Γ_n to Γ is in the sense of their characteristic functions [11, 19]. We have also the convergence of D_n and C_n to D and C respectively.

Finally the shape optimization problem can be given by the following:

$$\begin{cases} \text{Find } \Gamma^\star \in \theta_{ad} \text{ such:} \\ \Gamma^\star = \underset{\Gamma \in \theta_{ad}}{argmin} J(\Gamma) \end{cases} \tag{3.8}$$

To this end, we announce the next existence result:

Theorem 3.6. *The problem (3.8) admit at least one solution.*

The proof of this theorem is based on the compactness of the space θ_{ad}, and to prove the continuity of functional J. We will prove those two results in the next lemmas, thereafter the proof of Theorem 3.6 can be skipped.

We state the first result as follows

Lemma 3.7. *The space θ_{ad} is compact.*

The proof is based on the Ascoli-Arzela theorem [21], for the detailed proof we refer to [6, 7].

Now, let us prove the second result:

Lemma 3.8. *The functional J is continuous on θ_{ad}.*

Proof. Let Γ_n be a sequence in θ_{ad} that converge to Γ, $u_n = u(\Gamma_n)$ and $w_n = w(\Gamma_n)$ are the associated solution, we shall prove that:

$$\lim_{n \to \infty} J(\Gamma_n) = J(\Gamma).$$

We have

$$\begin{aligned} J(\Gamma_n) &= \frac{1}{2} \int_{D_n} [(u_n - w_n)^+]^2 dx + \frac{1}{2} \int_{C_n} (u_n - w_n)^2 dx, \\ &= \frac{1}{2} \int_\Omega \chi_{D_n} [(u_n - w_n)^+]^2 dx + \frac{1}{2} \int_\Omega \chi_{C_n} (u_n - w_n)^2 dx. \end{aligned}$$

As results of Lemmas 3.3 and 3.4, u_n and w_n are bounded in $H^1_{\Gamma_D}(\Omega)$, then they converge weakly in $H^1_{\Gamma_D}(\Omega)$, the fact that the impending $L^2(\Omega) \hookrightarrow H^1_{\Gamma_D}(\Omega)$ is compact, then the convergence is strong in $L^2(\Omega)$, hence

$$\lim_{n \to \infty} \int_{C_n} (u_n - w_n)^2 dx = \int_C (u - w)^2 dx.$$

For the same reason we have also:

$$\lim_{n\to\infty}\int_{D_n}[(u_n-w_n)^+]^2dx=\int_D[(u-w)^+]^2dx.$$

In conclusion the functional J is continuous on $\mathcal{F}$. □

To solve this shape optimization problem, we use a particle swarm optimization method combined with a finite element method. The problems (3.3) and (3.5) are resolved when the depletion layer is assumed known.

4. Particle swarm optimization

Particle Swarm Optimization(PSO) [16] is one the efficient heuristic methods, it consist to search the optimal solution of an optimization problem by adjusting a candidate solution iteratively according to the model inspired from the birds behaviour's in swarm. Like the famous Genetic algorithm (GA) [12], PSO works on a swarm S (population in GA) of particles x (candidate solutions), which can move into the swarm with a velocity v. There is a second swarm P where we store the best particles in each position over iteration. The best global particle is obtained from the swarm of best position based on its fitness score. At the iteration n we update the velocity and the particle with respect to the Equations (4.1) and (4.2):

$$v_i^{n+1} = \omega v_i^n + c_1 r_1 (p_i^n - x_i^n) + c_2 r_2 (p_g^n - x_i^n), \tag{4.1}$$

$$x_i^{n+1} = x_i^n + v_i^{n+1}. \tag{4.2}$$

Where c_1 is the cognitive parameter, c_2 is the social parameter, r_1 and r_2 are two vectors of random floats in the range $[0, 1]$. ω is the inertia weight.

Algorithm 1:

Input: Choose a precision tol (or maximum iteration), c_1, c_2

```
Generate a random S in the range [0, L_0] × [0, H] and set P = S
For each particle in the swarm S solve the problems (3.3) and (3.5)
Find the best particle Γ_best,
while | J(Γ_best) | > tol do
    Update the swarm P,
    Update the swarm S using Equations (4.1) and (4.2),
    For each particle in the swarm S solve the problems (3.3) and (3.5)
    Find the best particle Γ_best,
```

Output: The optimal solution (Γ_{best}, u, w)

5. Algorithm validation

Before we turn the proposed algorithm to solve the free boundary problem in the MESFET, we shall show its validity to solve similar problem when the free boundary is supposed known. We keep the same geometry given in Figure 1. We test our algorithm for two cases, the first is when the free boundary is given by control points, the other one is when it is given by a explicit function.

For the PSO parameters we take as the swarm size 30 particles, the cognitive and the social parameters are 1.5 and 2 respectively. The precision tol in Algorithm 1 is replaced by a maximum iteration number, which we choose equal to 50.

Example 5.1. We assume that the free boundary is given by the control points

$$p_1 = (1.7, 2),\ \ p_2 = (2.35, 1.3),\ \ p_3 = (3, 1.2),\ \ p_4 = (3.65, 1.3),\ \ \text{and}\ \ p_5 = (4.3, 2).$$

In Figures 2 and 3 we remark that, after 40 iterations the calculated boundary is closer enough to the exact one. The proposed algorithm converges fast to the exact boundary, after only 50 iterations, the cost is of order 10^{-2}. We conclude that the developed algorithm is efficient to solve the above free boundary problem.

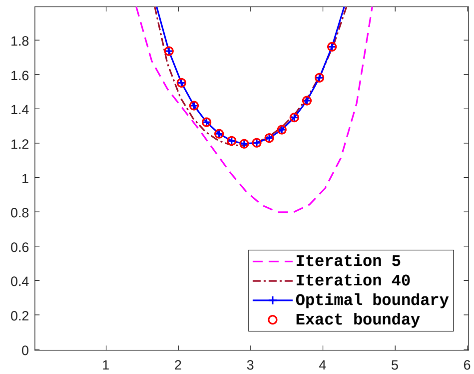

FIGURE 2. Example 5.1: comparison of obtained results.

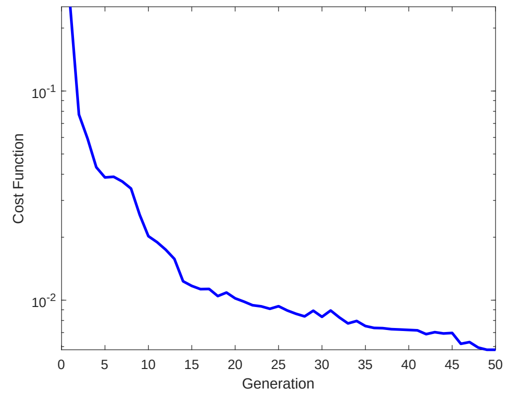

FIGURE 3. Example 5.1: The log scale of the cost.

Example 5.2. In this example we suppose that the free boundary is given by

$$\Gamma = \{(x, y) : \ x \in [1.3, 4.5] \text{ and } y = 2 + 0.5(1.3 - x)(4.5 - x)\}.$$

Figures 4 and 5, after 40 iterations the computed boundary become closer the exact boundary. At the end of maximum

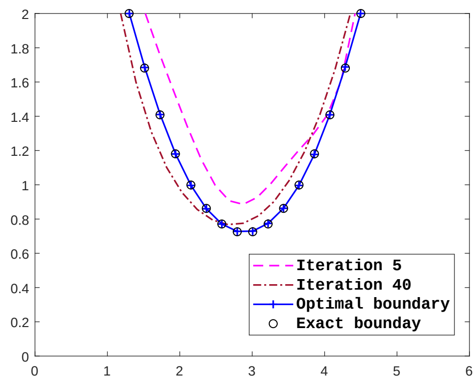

FIGURE 4. Example 5.2: comparison of obtained results.

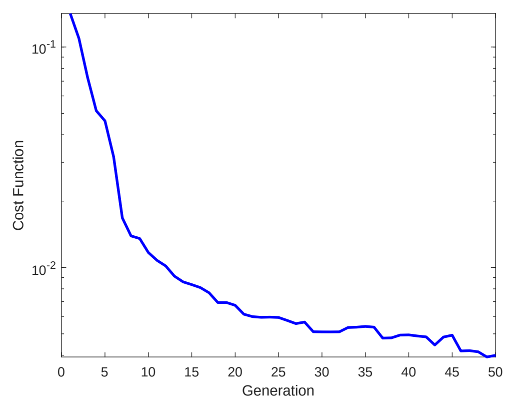

FIGURE 5. Example 5.2: The log scale of the cost.

iteration number the optimal boundary and the exact one are indistinguishable. Again we conclude the success of the proposed algorithm. At this stage we can say that Algorithm 1 is valid to solve this kind of shape optimization problem.

In order to investigate the sensitivity of the obtained results with respect to the presence of noise, we follow [6, 8] and add some noise to the boundary data u_d and w_d in the following way

$$\|u_d^\delta - u_d\| + \|w_d^\delta - w_d\| \le \delta$$

where u_d^δ and w_d^δ are the noise data, and δ is the noise level.

In Table 1 we read the optimal cost for examples 1 and 2 with respect to different noise levels.

One can observe that as the noise increases, the quality of the approximation based on the obtained error decreases, which is natural and expected. However, despite the increased errors, the approximation still sufficiently accurate.

Table 1. Comparison of errors.

Noise level	0%	1%	5%	10%
Example 1	0.0218	0.0287	0.0376	0.0877
Example 2	0.0093	0.0104	0.0481	0.0792

6. MESFET model simulation

We consider $\Omega = [0,6] \times [0,2]$ as the MESFET domain. We call the finite element $\mathbb{P}_1$ to solve the variational formulations (3.3) and (3.5). The free boundary Γ is given by a Bézier curve [20], which means we give a number of control points then we call the Casteljau [20] algorithm to determine the coordinates of Γ. In this application the mesh is not fix for the reason that we compute the state problem solution's in each region, we have to avoid the fix mesh used in the paper [3] for the simple reason that the solutions of problems (3.3) and (3.5) depend on the geometry of C and D, which change from iteration to another due to the change of the depletion layer while iteration, hence a new mesh is required. To mesh each region we call algorithm proposed in [18].

The process consist to solve the problems (3.3) and (3.5) with a given candidate Γ (a particle of the swarm), then we compute the value of J, which was constructed to keep the constraints $u = w$ in C and $u < w$ in D hold, the value of J is the fitness of the particle. We run the algorithm for 100 times, the optimal boundary for each example is the mean of the obtained boundaries in the 100 runs for the same example.

The parameters used for PSOFEM algorithm are given in Table 2.

Table 2. PSO parameters

Swarm size	Max iterations	cognitive parameter	social parameter
30	700	2	1.5

6.1. **Depletion mode.** The depletion mode is realized when we fix a zero voltage on the gate terminal, and vary the voltage on the drain terminal. In Figure 6 we show the obtained depletion layers for different voltages under the depletion mode, where we fix $0V$ on the gate and we vary the drain voltage from $0.1V$, $0.4V$ to $0.7V$. It is obvious that the depth of the depletion region increase while we vary the drain voltage's.

In Figure 7 we plot the potential u for the depletion mode, when the applied voltage on the drain is $V^+ = 0.4$. We can see that the potential u is flowing, its lower values are near to the gate and does not exceed the middle of the domain.

6.2. **Enhancement mode.** In this mode the voltage gate's is not fix. In Figure 8 we show the obtained depletion layers for different voltages under this mode. We fix the applied voltage on the drain at $+0.4V$, then we vary the voltage on the gate from $-0.1V$, $-0.4V$ to $-0.7V$. We can remark that the depletion region is more wider if we compare it to the last mode. We show in Figure 9 the potential u for this mode when $V^- = -0.4$. It is seen that u in the middle of the MESFET start vanishing.

6.3. **High potential on the gate.** We also simulate the case when we apply higher voltage on the gate, we set 0.7 at the drain terminal, and we vary the gate voltage. As we can see in Figure 10 the depletion zone is too wide, which can understand as it tries to stop the current flow. Also it is demonstrated in Figure 11 where the potential u vanishes in the bottom-center of the MESFET.

In the table we show the cost for each case of functioning.

7. Conclusion

In this paper, we have investigated the existence of a solution to a coupled system with a free boundary arising from the MESFET semiconductor device. To solve this inverse problem, we proposed a combined finite element and particle swarm optimization approach. We demonstrated the validity of our method through numerical simulations

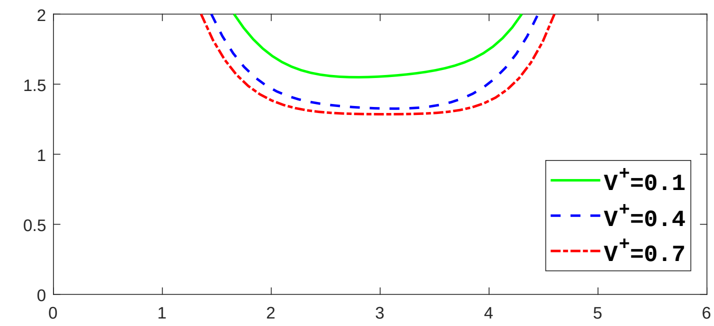

FIGURE 6. Optimal layers under depletion mode.

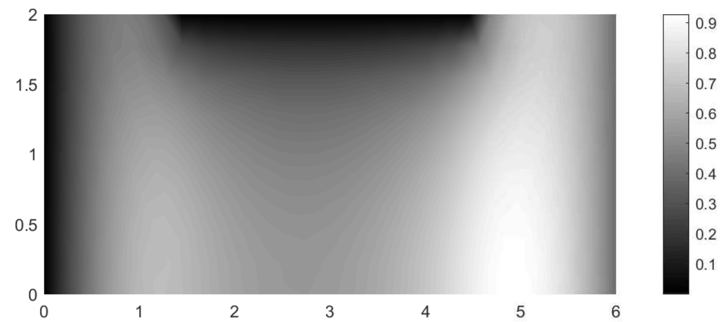

FIGURE 7. The potential u when $V^+ = 0.4$ and $V^- = 0$.

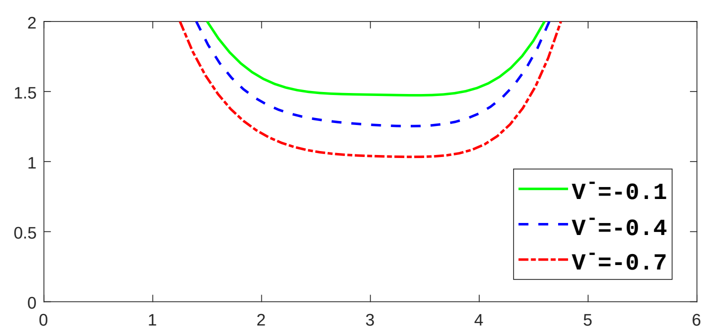

FIGURE 8. Optimal layers under enhancement mode.

TABLE 3. The cost for some examples.

	$V^- = 0$ and $V^+ = 0.4$	$V^- = -0.4$ and $V^+ = 0.4$	$V^- = -3$ and $V^+ = 0.7$
Cost on C	0.001	0.022	0.083
Cost on D	0.115	0.431	0.952

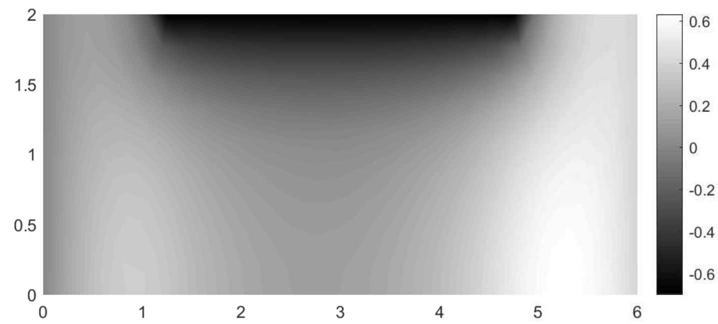

FIGURE 9. The potential u when $V^+ = 0.4$ and $V^- = -0.4$.

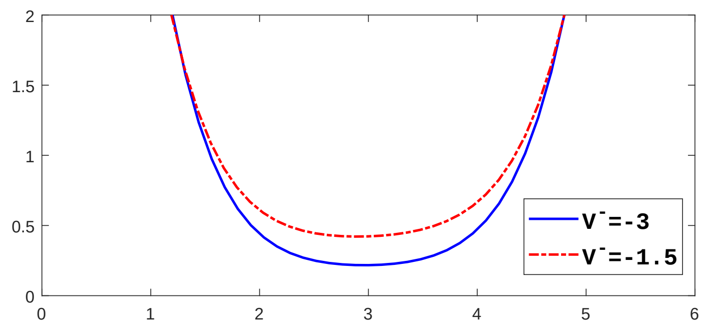

FIGURE 10. Optimal boundaries under high potential mode.

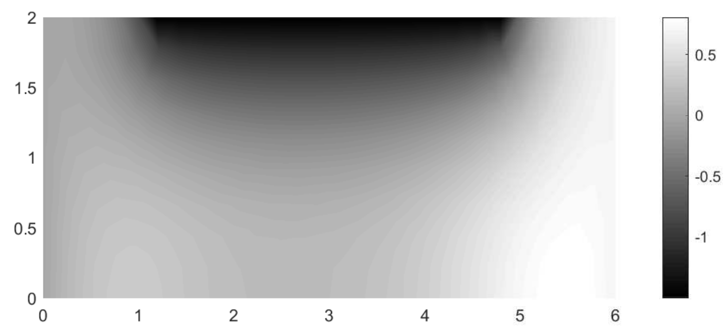

FIGURE 11. The potential u when $V^+ = 0.7$ and $V^- = -1.5$.

for two cases of MESFET functioning, namely the depletion and enhancement modes, as well as a case where a higher voltage is applied to the gate. Our results, shown in the six figures and Table 3, are consistent with the expected properties of MESFET functioning. For future work, we suggest exploring the identification and reconstruction of the free surface of depletion in a three-dimensional MESFET device, which is a challenging inverse problem that may require a robust optimization approach.

References

[1] J. Abouchabaka, R. Aboulaich, and A. Souissi, *Numerical approach of a free boundary in the junction field effect transistor – MESFET*, Math. Comput. Simul., *47* (1998), 531–539.

[2] J. Abouchabaka, R. Aboulaich, O. Guennoun, A. Nachaoui, and A. Souissii, *Shape optimization for a simulation of a semiconductor problem*, Math. Comput. Simul., *56* (2001), 1–16.

[3] J. Abouchabaka, R. Aboulaich, A. Nachaoui, and A. Souissi, *A decoupled algorithm for a drift-diffusion model*, Math. Method. Appl. Sci., *28* (2005), 1291–1313.

[4] A. Boulkhemair and A. Chakib, *On the Uniform Poincaré Inequality*, Commun. Partial Differ. Equ., *32*(9) (2007), 1439-1447.

[5] Y. El Yazidi and A. Ellabib, *Augmented Lagrangian approach for a bilateral free boundary problem*. J. Appl. Math. Comput., *67*, (2021), 69-88.

[6] Y. El Yazidi and A. Ellabib, *An iterative method for optimal control of bilateral free boundaries problem*, Math. Methods Appl. Sci., *44*(14), (2021), 11664-11683.

[7] Y. El Yazidi and A. Ellabib, *A new hybrid method for shape optimization with application to semiconductor equations*, Numer. Algebra Control Optim., *12*(4), (2021), 763-781.

[8] Y. El Yazidi and A. Ellabib, *A fuzzy particle swarm optimization method with application to shape design problem*, RAIRO Oper. Res., *57*(5) (2023), 2819-2832.

[9] A. Charkaoui and Y. El Yazidi, *Level Set and Optimal Control for the Inverse Inclusion Reconstruction in Electrical Impedance Tomography Modeling*, Int. J. Comput. Methods, *21*(01) (2024), 2350020

[10] J. Haslinger and R. A. E. Mäkinen, *Introduction to Shape Optimization*, SIAM, 2003.

[11] A. Henrot and M. Pierre, *Variation et optimisation de formes*, Springer-Verlag Berlin Heidelberg, 2005.

[12] D. E. Goldberg, *Genetic Algorithm in Search, Optimisation, and Machine Learning*, Addison-Wesley Longman Publishing Co, 1989.

[13] M. Marin, *An uniqueness result for body with voids in linear thermoelasticity*, Rend. Mat. Appl., *17*(7) (1997), 103–113.

[14] M. Marin and G. Stan, *Some Basic Results in Nonlinear Theory of Dipolar Porous Materials*, Journal of Porous Media, *16*(11) (2013).

[15] P. A. Markowich, *The Stationary Semiconductor Device Equations*, Springer Vienna, 2013.

[16] C. Maurice, *Particle Swarm Optimization*, ISTE, 2006.

[17] G. Nakamura, G. Uhlmann, and J.-N. Wang, *Reconstruction of cracks in an inhomogeneous anisotropic medium using point sources*, Adv. in Appl. Math., *34*(3) (2005), 591–615.

[18] P. O. Persson and G.Strang, *A Simple Mesh Generator in MATLAB*, SIAM Review, *46* (2004).

[19] O. Pironneau, *Optimal shape design for elliptic systems*, Springer Berlin Heidelberg, 1982.

[20] H. Prautzsch, W. Boehm, and M. Paluszny, *Bézier and B-Spline Techniques*, Springer, Berlin, Heidelberg, 2002.

[21] W. Rudin, *Functional Analysis*, McGraw-Hill, 1991.

[22] S. Salhi, *Heuristic Search: The Emerging Science of Problem Solving*, Palgrave Macmillan, 2017.

[23] S. H. Sun, T. T. Yu, T. T. Nguyen, E. Atroshchenko, and T. Q. Bui, *Structural shape optimization by IGABEM and particle swarm optimization algorithm*, Eng. Anal. Bound. Elem., *88* (2018), 26–40.

[24] S. M. Sze, *Semiconductor Devices: Physics and Technology*, John Wiley & Sons Singapore Pte. Limited, 2012.

[25] Y. Tian, S. L. Xu, P. Chen, and Y.-P. Xu, *Spectrum Reconstruction from Multiple Band Spectrum Data using an Inversion Technique*, Int. J. Nonlinear Sci. Numer. Simul., *10* (2009), 1281–1290.