Simulation Tool for Optimization of Digital Active Gate Drive Sequence Using Genetic Algorithm

Hajime Takayama*, Shuhei Fukunaga†, and Takashi Hikihara*

∗ Department of Electrical Engineering,
Kyoto University
A1-412, Kyoto-university-Katsura, Nishikyo
Kyoto 615-8246, Japan
Phone: +81 75-383-2241
Email: h-takayama@dove.kuee.kyoto-u.ac.jp

† Division of Electrical, Electronic,
and Infocommunication Engineering,
Osaka University
Yamadaoka 2-1, Suita
Osaka 565-0871, Japan

Acknowledgments

This work was partially supported by the Program on Open Innovation Platform with Enterprises, Research Institute and Academia (OPERA) of Japan Science and Technology Agency, the Cross-ministerial Strategic Innovation Promotion Program (SIP), "Energy systems of an Internet of Energy (IoE) society"(Funding agency: JST), and JSPS KAKENHI Grant numbers 20H02151 and 21J22448.

Keywords

≪Digital control≫, ≪Hard switching≫, ≪Genetic algorithm≫, ≪Optimization method≫, ≪Silicon Carbide (SiC)≫, ≪Simulation≫, ≪Smart Gate Drivers≫.

Abstract

We demonstrate a simulation tool to optimize the operation of the digital active gate driver (DAGD) for SiC MOSFETs. The binary nature of DAGD's operating principle makes the genetic algorithm a preferable method. The optimization tool is developed by combining a Python-based program and SPICE simulation. Optimized solutions exhibit improved switching characteristics in wide operating conditions. The effect of conditions on obtained solutions is also analyzed.

Introduction

Active gate drive (AGD) has been gathering much attention in the hard switching of power devices. In particular, various types of AGDs for wide band-gap power devices have been studied because they face increased overshoots of the device voltage and current, which can lead to serious reliability problems such as electromagnetic interference and crosstalks [1, 2]. Various types of AGDs have been studied so far, such as gate-resistance-controlled methods [1], gate-voltage-controlled methods [3–7], and others [2,8]. The digitization of AGDs has also been attempted [4,5,8]. Unlike analog AGDs and other methods to suppress the overshoots, such as snubber circuits, they can adjust their operation without changing the circuit itself, which adds to the flexibility in their control strategy.

We particularly put the focus on the gate-voltage-controlled AGDs, which have been investigated in several research [3–7]. They directly designate a certain state of MOSFETs during the transient state of switching by setting particular gate-voltage level. This makes it easier to understand the relation between the control strategy and the device characteristics, which differs from device to device and, in addition, depends on the operating conditions.

For the pursuit of the potential of digitized AGDs, the digital control scheme to optimize their operation needs to be developed. Gate-voltage selection based on the modeling approach is investigated in [3].

However, they require an accurate parameter estimation or identification in advance, which is challenging for some applications. This work explores the potential of applying the genetic algorithm [9], one of the well-known metaheuristics, to obtain the optimized operation of AGD in a wide operating range. Since it takes a combinatorial-optimization approach, it can be applied to any device characteristics and operating conditions, even if they are unknown. A simulation tool using Python software and SPICE simulation is originally developed for this purpose. The obtained solutions are analyzed from the viewpoints of the device characteristics and the operating conditions.

GA-based operation optimization of digital active gate driver

This section introduces the digital active gate driver (DAGD) studied in [5] and the simulation tool using Python software and SPICE simulation. Due to the binary nature of the operation of DAGD, the application of the genetic algorithm is found preferable for optimizing its operation. The details of the algorithm and simulation settings are also described.

Configuration of 4-bit DAGD and simulated circuit

DAGD is designed from the architecture of the binary-weighted resistor digital-to-analog converter. The circuit configuration of 4-bit DAGD is shown in Fig. 1. The gate-source voltage (V_{GS}) waveform is shaped by the successive alteration of V_{GS} using a multi-bit gate signal sequence; V_{GS} is set to the value given by (1) every time the gate signals (b_0 to b_3) are changed at the clock rate of the outer controller such as FPGAs.

$$V_{\mathrm{GS}} = \frac{\sum_{j=0}^{3} b_j 2^j}{\sum_{j=0}^{3} 2^j} V_{\mathrm{drv}} \tag{1}$$

The operation of DAGD is determined by means of this gate signal sequence, which proposes a great controllability of the V_{GS} waveform. The voltage-related characteristics of the MOSFET, such as the threshold voltage and the Miller plateau voltage, can be effectively taken into consideration in shaping the V_{GS} waveform. This advantage, however, requires proper control methods as the adjustment of the gate driver's operation is totally handled by the digital controller, not by adjusting the circuit itself.

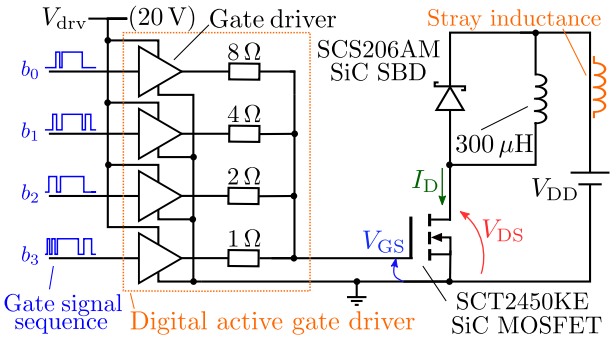


Fig. 1: Configuration of 4 bit DAGD. Double pulse testing circuit is assumed as the load.

The right part of Fig. 1 is an inductive-load test circuit known as the double-pulse testing, composed of a SiC MOSFET, a SiC Schottky barrier diode, a load inductor, and a power source (V_{DD}). It can set arbitrary operating conditions, a set of drain-source voltage (V_{DS}) and drain current (I_{D}), which needs to be taken into consideration in the AGD strategy. The test circuit also includes stray inductance of 80 nH at the wire from V_{DD} and 5 nH at each terminal of power devices.

For the model of the SiC MOSFET, we use the surface-potential-based model developed in [10]. Since it reflects the physical structure of the MOSFET using the surface equation, it can reproduce the accurate I–V and C–V characteristics even in the high-power region, where the actual switching operations in power converters take place. This helps the development of AGD strategy of DAGD as it is strongly related to these attributes.

Optimization tool based on GA

To find the optimum gate signal sequence, we here adopt a multi-objective genetic algorithm known as NSGA-II [9]. The optimization process is described in Fig. 2. The NSGA-II is implemented using a Python framework called DEAP [11], and the SPICE simulation is performed using SIMetrix software (Ver. 7.20j) [12].

Firstly, 60 genes are randomly selected to create an initial population. The genes are encoded into the gate signal sequences, which are the 4-bit time-series signal data saved in text-file format to be read by the netlist file. The Python program then sends a command to run the SIMetrix script, which simulates the double-pulse testing for all genes. After the simulation finishes, the program reads the waveform data to calculate the two fitness. Here, we use the switching loss and the amount of overshoot (peak of V_{DS} for turn-off and that of I_{D} for turn-on) as the two fitness, according to which the individuals are selected. It is noted that all the individuals pass the selection in the first loop as there are just 60 individuals in the population. The offspring population is generated from the current population through two-point crossover and one-bit-flip mutation, whose probabilities are set at 0.9 and 0.1, respectively. Their fitness are evaluated in the same way as described above. From the second loop, 60 individuals are selected from a set of the parent population and the offspring population, whose size is 120. By doing so, the Pareto front solutions are kept and updated. The number of generations is fixed at 30.

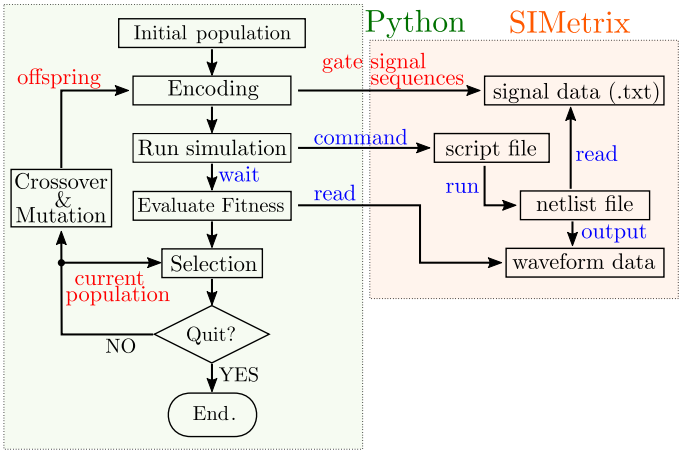


Fig. 2: Flowchart of optimization based on NSGA-II.

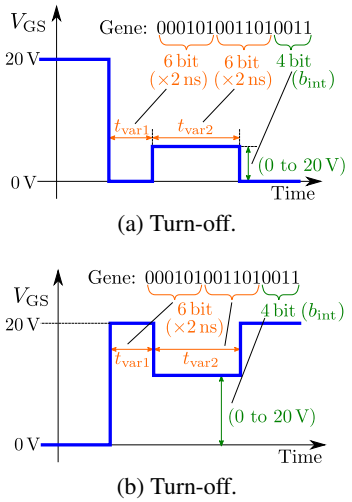


Fig. 3: Genotype of gate signal sequences.

Genotype of gate signal sequences

The multi-bit gate signal sequence of DAGD is naturally a binary sequence, which can be directly used as the genes and also can be easily converted to other genotypes. We adopt the genotype as shown in Fig. 3 to limit the degree of freedom into three, making the analysis simple and effective. The sequence for turn-off/on is described as follows (hereinafter, the prefix "0x" means the following number is hexadecimal): firstly, the gate signal is set at 0x0/0xF for a duration of t_{var1}; then, it is changed to b_{int} and kept for t_{var2}; finally, it is set at 0x0/0xF again to complete the turn-off/on operation. The first and following 6 bits of the gene represent t_{var1} and t_{var2}, respectively, while the last 4 bits represent b_{int}.

It is worthwhile to mention that this kind of V_{GS} waveform is sometimes referred to as the three-level (3-L) waveform in the research of voltage-controlled AGDs [3]. It is ideal for analyzing the relationship between a particular voltage level, which corresponds to b_{int}, and the resulting switching characteristics.

Verification of optimization tool

The proposed optimization tool is performed in simulation. Firstly, the tool is verified in a single operating condition, and the obtained solutions are analyzed in detail. A comparison with a standard gate

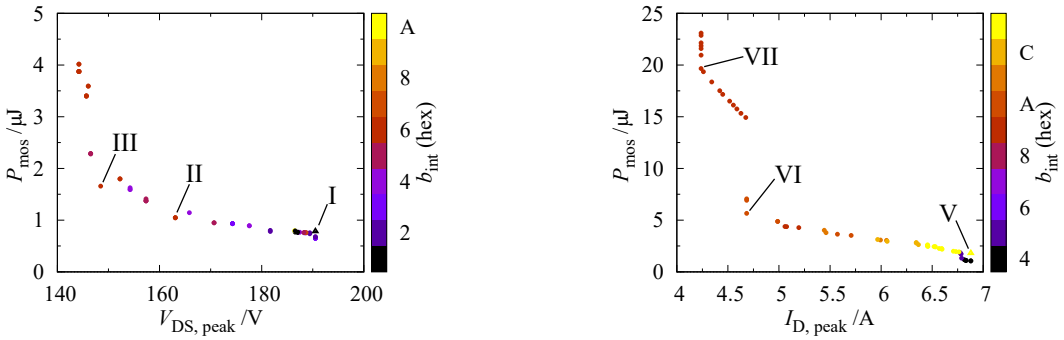
driving and optimization results in other operating conditions are given afterward.

Optimization in a single operating condition

For the system verification, the operating condition is set at $(V_{\mathrm{DS}},\ I_{\mathrm{D}}) = (120\,\mathrm{V}, 4\,\mathrm{A})$. Figs. 4 (a) and (b) show the fitness maps of solutions in the final generation population at turn-off and turn-on, respectively. The selected values of b_{int} are also expressed by the color bar. These figures confirm that the Pareto-front solutions are successfully obtained by the optimization both at turn-off and turn-on. Here, we select some of the Pareto-front solutions marked with Roman numbers to see the transient behavior. The switching waveforms of these solutions are shown in Fig. 5. Solutions I and V, which are shown with triangles in Fig. 4, refer to the case without AGD, where $t_{\mathrm{var1}} = t_{\mathrm{var2}} = 0$ and $b_{\mathrm{int}} = 0\mathrm{x}0/0\mathrm{xF}$ for turn-off/on.

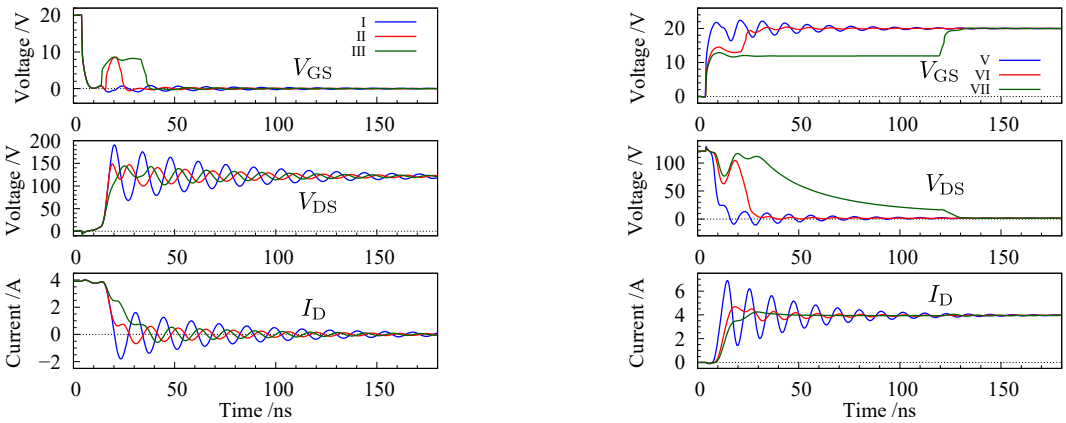
In turn-off, 37 solutions out of 60 belong to the Pareto front. The solution I is located at the right edge of the front, meaning that it has the most significant surge voltage. Solutions with the value of b_{int} typically around 4 to 6 are scattered along the front. Solutions II and III have the same b_{int} but different t_{var1} and t_{var2}, which produces the difference in their fitness. It is an interesting result that several levels of V_{GS} can be used to obtain similar values of fitness.

In turn-on, 57 solutions belong to the Pareto front. Unlike at turn-off, the Pareto front can be divided into groups of solutions with the same value of b_{int} located closely to each other. For example, solutions around VII have $b_{\mathrm{int}} = 0\mathrm{x}9$, and solutions around VI have $b_{\mathrm{int}} = 0\mathrm{x}8$. The value of b_{int} increases as the solution locates further to the bottom-right.



(a) Turn-off solutions. (b) Turn-on solutions.

Fig. 4: Fitness map of obtained solutions in final generation.



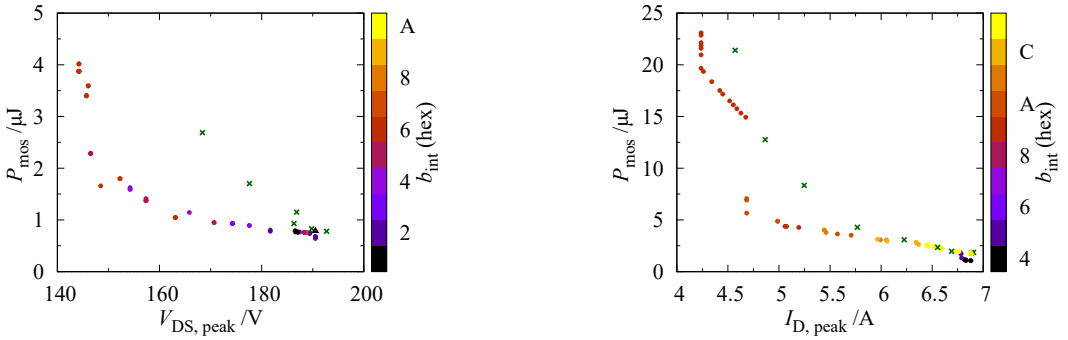
(a) Turn-off solutions I, II, and III. (b) Turn-on solutions V, VI, and VII.

Fig. 5: Selected switching waveforms of obtained solutions.

Comparison with gate-resistance-incremental method

The obtained Pareto front solutions are compared with those of standard gate-driving (SGD), where only 1-bit of the DAGD is used, and the gate resistance is set at $2^k\,\Omega$ ($k = 1, 2, 3\ldots$). Figs. 6 (a) and (b)

show the comparison at turn-off and turn-on, respectively, where SGD solutions are plotted with crosses. It is clear that the optimized DAGD solutions show a better trade-off between the two fitness in both cases. This is because the optimized solution of DAGD slows the switching only during the critical period, while in SGD the switching is stalled for the entire duration of the switching. These results show that DAGD outperforms SGD with a properly-optimized gate signal sequence, which can be selected depending on the design constraints.



(a) Turn-off solutions.

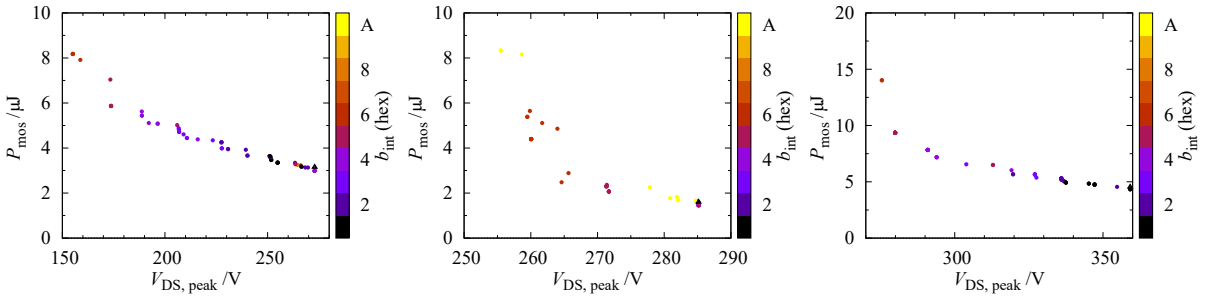
(b) Turn-on solutions.

Fig. 6: Comparison of fitness between DAGD optimized by GA (dots) and SGD (crosses).

Optimization in different operating conditions

So far, the optimization tool of DAGD is verified in a single operating condition. We now apply it to a variety of operating conditions to see how the optimized solution changes accordingly. The operating conditions of $(V_{DS}, I_D) = (120\,\mathrm{V}, 8\,\mathrm{A})$, $(240\,\mathrm{V}, 4\,\mathrm{A})$, and $(240\,\mathrm{V}, 8\,\mathrm{A})$ are selected additionally. The focus is put on what value of b_{int} is selected, so compare the fitness maps of the optimized solutions.

Figure 7 shows the optimization results at turn-off in the three operating conditions. By comparing the three figures and also Fig. 4 (a), it is found that similar values of b_{int} of 0x4 to 0x6 are selected in every operating condition. This result is understood with the threshold voltage of MOSFETs, which does not change drastically when operating condition changes.



(a) 120V, 8A.

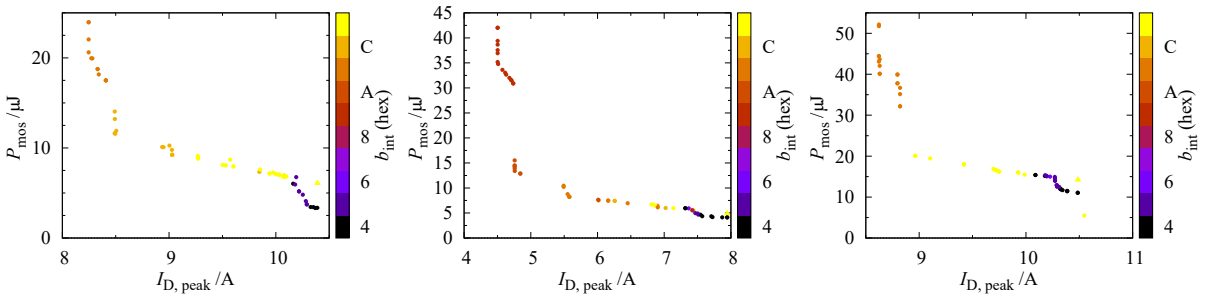
(b) 240V, 4A.

(c) 240V, 8A.

Fig. 7: Optimization results at turn-off in varied operating conditions.

Figure 8 shows the optimization results at turn-on. We clearly recognize a difference in the selection of b_{int}. For example, if we compare the solutions with the least $I_{D,peak}$, the selected bit is 0xB in Fig. 8 (a) and (c), while it is 0x9 in Fig. 4 (a) and Fig. 8 (b). This suggests that the effective value of b_{int} is higher for an operating condition with larger current. It complies with the fact that the Miller plateau voltage of the device increases as the operating current becomes larger.

These results confirm that the operating condition affects the AGD strategy mainly at turn-on. The proposed tool selects the appropriate gate signal sequences in every condition, dealing with such changes.



(a) 120V, 8A. (b) 240V, 4A. (c) 240V, 8A.

Fig. 8: Optimization results at turn-on in varied operating conditions.

Conclusion

This paper provided the Python-based simulation tool to optimize the gate signal sequence of DAGD using GA. The gate signal sequence is converted into a genotype with three variables to realize a simple and effective analysis of the resulting switching characteristics. It was shown that the Pareto-front solutions were successfully obtained, which outperformed the basic gate-driving method of increasing the gate resistance. It was also confirmed that the operating conditions play a critical role in the selection of effective V_{GS} level at turn-on in response to the shift of the Miller plateau voltage.

By utilizing the results obtained in simulation, the presented tool can be used for predicting the proper gate signal sequence in the experiment, according to the approved rate of overshoots and losses. Practical limitations, such as the time restrictions of the controllers and buffers, need to be taken into account. Also, the effect of the temperature and the resulting thermal drift are not considered in the simulation. A combination with a local search algorithm using closed-loop feedback will help optimize the operation furtherly according to such dispersion of device characteristics. The experimental verification of the optimization tool is now under development.

References

[1] A. P. Camacho, V. Sala, H. Ghorbani, and J. L. R. Martinez, "A Novel Active Gate Driver for Improving SiC MOSFET Switching Trajectory," *IEEE Trans. Ind. Electron.*, vol. 64, no. 11, pp. 9032–9042, 2017.

[2] Z. Zhang, J. Dix, F. F. Wang, B. J. Blalock, D. Costinett, and L. M. Tolbert, "Intelligent Gate Drive for Fast Switching and Crosstalk Suppression of SiC Devices," *IEEE Transactions on Power Electronics*, vol. 32, no. 12, pp. 9319–9332, 2017.

[3] S. Zhao, A. Dearien, Y. Wu, C. Farnell, A. U. Rashid, F. Luo, and H. A. Mantooth, "Adaptive Multi-Level Active Gate Drivers for SiC Power Devices," *IEEE Transactions on Power Electronics*, vol. 35, no. 2, pp. 1882–1898, 2020.

[4] D. J. Rogers and B. Murmann, "Digital active gate drives using sequential optimization," in *2016 IEEE Appl. Power Electron. Conf. and Expo. (APEC)*, ser. 2016 IEEE Appl. Power Electron. Conf. and Expo. (APEC), 2016, pp. 1650–1656.

[5] H. Takayama, T. Okuda, and T. Hikihara, "Digital active gate drive of SiC MOSFETs for controlling switching behavior—Preparation toward universal digitization of power switching," *International Journal of Circuit Theory and Applications*, no. July, p. cta.3136, 9 2021. [Online]. Available: https://onlinelibrary.wiley.com/doi/10.1002/cta.3136

[6] N. Idir, R. Bausière, and J. J. Franchaud, "Active Gate Voltage Control of Turn-on dv/dt and Turn-off dv/dt in Insulated Gate Transistors," *IEEE Trans. Power Electron.*, vol. 21, no. 4, pp. 849–855, 2006.

[7] P. D. Judge, R. Mathieson, and S. Finney, "A Six Level Gate-Driver Topology with 2 . 5 ns Resolution for Silicon Carbide MOSFET Active Gate Drive Development," in *12th Energy Conversion Congress and Exposition - ASIA (ECCE ASIA)*, 2021, pp. 2118–2123.

[8] K. Miyazaki, S. Abe, M. Tsukuda, I. Omura, K. Wada, M. Takamiya, and T. Sakurai, "General-Purpose Clocked Gate Driver IC With Programmable 63-Level Drivability to Optimize Overshoot and Energy Loss in Switching by a Simulated Annealing Algorithm," *IEEE Transactions on Industry Applications*, vol. 53, no. 3, pp. 2350–2357, 2017.

[9] K. Deb, A. Pratap, S. Agarwal, and T. Meyarivan, "A fast and elitist multiobjective genetic algorithm: NSGA-II," *IEEE Transactions on Evolutionary Computation*, vol. 6, no. 2, pp. 182–197, 2002.

[10] Y. Nakamura, M. Shintani, K. Oishi, T. Sato, and T. Hikihara, "A simulation model for SiC power MOSFET based on surface potential," in *International Conference on Simulation of Semiconductor Processes and Devices (SISPAD)*. Institute of Electrical and Electronics Engineers Inc., 10 2016, pp. 121–124.

[11] DEAP Project, "DEAP documentation," https://deap.readthedocs.io/en/master/, (accessed on 9th May 2022).

[12] SIMetrix Technologies Ltd., "SIMetrix Simulator," https://www.simetrix.co.uk, (accessed on 9th May 2022).