Linear regression model for screening SiC MOSFETs for paralleling to minimize transient current imbalance

James Abuogo¹, Zhibin Zao¹ and Junji Ke¹

¹State Key Laboratory of Alternate Electrical Power Systems with Renewable Energy Sources, North China Electric Power University, 102206 Changping District, Beijing, China
E-mail: zhibinzhao@126.com

Abstract. This paper describes the development of a machine learning model which can be used to screen SiC MOSFETs for paralleling with minimized transient current imbalance. The spread of device parameters is determined to isolate the device parameters which are likely to significantly influence transient current distribution. A linear regression model is then developed and trained using device parameters’ data measured from forty devices of the same production lot. The resulting model is an expression for current imbalance as a function of device parameters, with weight of each parameter determined. The performance of the trained model on a set of twenty testing devices is then determined and verified via a double pulse test experiment incorporating paralleled devices. The model is found to perform satisfactorily and can effectively be deployed in screening devices for paralleling.

1. Introduction
Over the recent years, Silicon Carbide (SiC) Metal Oxide Semiconductor Field Effect Transistor (MOSFET) has received a lot of attention due to its fast switching speed, high breakdown voltage, good thermal conductivity and low switching loss [1-3]. Despite having superior material properties over conventional Silicon Insulate Gate Bipolar Transistor (Si IGBT) and Silicon MOSFET, one significant drawback of SiC MOSFET is its low current carrying capacity [1-3]. This is a hindrance to deployment of SiC MOSFETs in high power and high current applications. To navigate this problem, parallel connection of discrete SiC MOSFET devices or building a multichip module have been the most common solutions [1-4]. However, studies of paralleled SiC MOSFET devices or chips reveal that mismatches in device and circuit parameters lead to an imbalanced current distribution among the paralleled units. The mismatch in device parameters is significant in SiC MOSFETs due to the relatively immature fabrication technology of SiC MOSFETs compared to other conventional power electronic devices like Si IGBTs and Si MOSFETs [5]. This leads to overcurrent in some units which further causes excessive heating with the final result being device degradation or eventually failure. According to [5, 6], device parameters which affect current sharing among paralleled SiC MOSFET devices include the on-resistance, pinch-off voltage, reverse breakdown voltage of gate, and static transfer characteristics. In [6-8], distribution of transient current is attributed to gate threshold voltage ($V_{th}$) while that of steady state current is dependent on on-resistance. In some works like [9, 10], only $V_{th}$ is considered for balancing transient current distribution while in other works like [5, 6] other parameters are found to contribute to the current imbalance as well. Therefore, in this paper, balancing of transient current for paralleled devices by considering multiple parameters simultaneously is sought.
Some parameters which can potentially affect transient current distribution are considered. First, the variation of each of these parameters among a group of devices picked from a single production batch is investigated. The parameters which show significant variation are then subjected to screening using a machine learning approach. Linear regression method of machine learning is used to determine the extent to which each of these parameters contribute to current sharing and express this contribution in terms of weights. The resulting model is then used to screen twenty testing devices to assess the model’s performance.

The organization of the paper is as follows. Section I introduces the paper. Section II describes supervised machine learning and outlines how it has been applied for screening SiC MOSFETs in this paper. Section III presents device parameters’ test results and shows the statistical description of the batch of devices used in developing the screening model. Section IV describes the training of the linear regression model while Section V assesses the model’s performance through experiment. Section VI concludes the paper.

2. Supervised Learning Algorithm

The screening approach adopted in this paper takes advantage of the supervised machine learning algorithms to screen SiC MOSFET devices. A number of different static and dynamic device parameters affect the current distribution of SiC MOSFETs. However, the relationship between the individual parameters with the current distribution is quite complex and one single function that can relate the current distribution with all the parameters has not been derived. Even though many researchers have identified the individual parameters that affect the current distribution, the extent to which each parameter affects current distribution has not been established. Screening has therefore been limited to consideration of a single parameter at a time.

Machine learning is the process of giving computers the ability to learn from data without being explicitly programmed [11, 12]. Application of machine learning to SiC MOSFETs screening involves training a model with data of device parameters and corresponding current imbalances to get a trained model. This is a supervised learning problem since device parameters are given as well as the label of corresponding current imbalances. From the trained model, the weights for individual parameters are obtained. These represent the contribution of each parameter to the current imbalance. The resulting model can be used to screen a new set of devices provided their parameters are measured and fed into the model. A linear regression method using gradient descent is deployed for this exercise. The flow of tasks is as shown in Figure 1.

3. Spread of SiC MOSFET Parameters

Due to relatively immature manufacturing technology of SiC MOSFET chips, the parameters of the devices vary significantly. The important device parameters include threshold gate voltage (V_{th}), transconductance (g_{fs}), MOSFET’s on-resistance (R_{on}), gate resistance (R_{g}), gate-drain capacitance (C_{gd}), gate-source capacitance (C_{gs}) and drain-source capacitance (C_{ds}). The spreads of these parameters among sixty devices drawn from a single manufacturing batch were determined using Agilent Curve Tracer. These devices were produced by one of the global leaders in semiconductor device manufacturing. Figure 2 to Figure 8 show the spread of each parameter among the sixty devices. C_{gs} is ignored in this paper since its influence on transient current imbalance is dependent on R_{g} [13]. The two parameters determine the switching delay time (\tau=R_{g}C_{gs}), where R_{g} is more dominant.
**Figure 1.** Overview of supervised learning screening approach

- **Training devices**
  - Curve Tracer Measurement
  - Parallel with a common device
  - Process data
  - Train

- **Testing devices**
  - Vth, Gfs, Rds, Rg

CI is percentage current imbalance

**Figure 2.** Spread of gate threshold voltage

**Figure 3.** Spread of device transfer curves

**Figure 4.** Spread of maximum transconductance of the devices

**Figure 5.** Spread of on-resistances of the devices

**Figure 6.** Spread of internal gate resistances of the devices
The output capacitance of SiC MOSFET package ($C_{gd}$ and $C_{ds}$) is dependent on the drain-source voltage. The Agilent Curve Tracer measures these capacitances as functions of $V_{ds}$. $C_{gd}$ curves for all the sixty devices are plotted on the same axes to display the variation in the parameter as shown in Figure 7. Likewise, $C_{ds}$ curves are plotted together in Figure 8.

From the spread of parameters plotted above, $V_{th}$, $g_{fsmax}$, $R_{ds}$ and $R_{g}$ are the ones which show large variations. $C_{gd}$ and $C_{ds}$ are approximately the same for the whole production batch. Hence current imbalance is mainly as a result of the former four parameters. The statistical description of the four parameters for the batch of sixty devices is summarized in Table 1 below where s.d. abbreviates standard deviation.

### Table 1. Statistics of the four selected device parameters

| Parameter | Mean | s.d. | Min  | Max  | Range  | Median |
|-----------|------|------|------|------|--------|--------|
| $V_{th}$  | 2.607 | 0.117 | 2.448 | 2.994 | 0.546  | 2.5795 |
| $g_{fsmax}$ | 10.26 | 0.434 | 9.361 | 10.89 | 1.529  | 10.304 |
| $R_{ds}$  | 0.084 | 0.0027 | 0.080 | 0.094 | 0.0141 | 0.0837 |
| $R_{g}$   | 5.738 | 0.408 | 4.842 | 6.273 | 1.4315 |

### 4. Training Algorithm

Machine learning algorithm was developed in MATLAB and devices’ data, already presented in the previous section, used to train a linear regression model. The test devices’ data was applied to the resulting model for prediction of their current imbalances. The remainder of this section describes the components of the learning algorithm used in this paper for developing a model for screening SiC MOSFETs for paralleling.

#### 4.1. Data Processing

The training data for any supervised machine learning project consists of the input data (presented as an m by n matrix, $X$) and label (which is an m dimension vector, $y$). Here m represents the number of training examples while n is the number of features. In this exercise, sixty devices are used. Out of the sixty devices, forty are used to train the model while twenty are used to verify the performance of the trained model. For the forty training devices, their parameters are measured to get the X matrix and later each of them is paralleled with a common device (device 0) which is not a part of this batch. The current imbalance (CI) is determined for parallel connection of each device with device 0. CI is the ratio of difference between $I_d$ values to their average expressed as a percentage. This gives the label vector $y$. Since the input data has features with considerably different numerical values, for instance, the average value of $R_{ds}$ is 0.084Ω while that of $g_{fsmax}$ is 10.26S, it is important to carry out feature normalization (or feature scaling) on the data before using it in training. This will make the gradient descent algorithm converge faster. Z-score feature normalization, with mean of zero and standard deviation of one, is used in this exercise. Equation (1) is the formula used for this normalization method.
Here, \( x_{j, \text{norm}} \) is the normalized value of the \( i^{th} \) training example of feature \( j \). \( \mu_j \) and \( \sigma_j \) are mean and standard deviation for feature \( j \) respectively.

4.2. **Gradient Descent Algorithm**

In linear regression method, the objective is to minimize the cost function expressed as (2) [11, 14].

\[
J(\theta) = \frac{1}{2m} \sum_{i=1}^{m} (h_\theta(x^{(i)}) - y^{(i)})^2
\]  

Where \( m \) is the total number of training examples and \( h\theta \) is a linear hypothesis given as (3), where \( n \) is the number of training features.

\[
h_\theta(x) = \theta^T x = \theta_0 x_0 + \theta_1 x_1 + \theta_2 x_2 + \ldots + \theta_n x_n
\]  

Batch gradient descent algorithm adjusts the values of \( \theta_j \) in order to minimize the cost. Each iteration of batch gradient descent simultaneously updates the weight parameter \( \theta_j \) for all features \( j \). The update function can be expressed as (4) where \( \alpha \) is the learning rate.

\[
\theta_j := \theta_j - \alpha \frac{\partial J(\theta)}{\partial \theta_j} = \theta_j - \alpha \frac{1}{m} \sum_{i=1}^{m} (h_\theta(x^{(i)}) - y^{(i)})x_j^{(i)}
\]  

To vectorize the implementation of this algorithm, a feature \( x_0 \) is introduced whose value is a vector of ones to multiply the intercept term \( \theta_0 \) of (3).

4.3. **Visualizing the Cost Function**

In order know whether the algorithm is working well or not, it is important to plot values of the cost after every iteration. The cost should monotonically reduce as the number of iterations increase. Plotting cost values (training, validation and test errors) enables debugging of the code, determining if the learning rate is too large or too small, and identifying presence of problems of over-fitting and under-fitting.

4.4. **Normal Equation Method**

Besides the gradient descent algorithm, normal equation method can also be used to solve the linear regression machine learning problem. This method uses the closed form solution of linear regression which is represented as (5). Normal equation method does not require feature scaling. It is also implemented without writing loops as the problem is completely vectorized and is achieved conveniently through matrix multiplication. This method gives exactly the same answer as gradient descent algorithm and is therefore used in this work to verify the results of gradient descent training algorithm.

\[
\theta = (X^T X)^{-1} X^T y
\]  

Here, \( X \) is the matrix of input training values, \( y \) is the vector of input training labels and \( \theta \) is the vector of trained model parameters (weights).

5. **Testing the Trained Model**

5.1. **Prediction using the Trained Model**

After training with the forty devices, a model is obtained with parameters (weights) corresponding to the contribution of each feature (device parameter) on current imbalance. This model is used to predict the current imbalance for each of the twenty remaining (testing) devices. This is similar to obtaining the current imbalance which could have been measured had each of them been paralleled with device 0. Since device 0 which these test devices are tested against is common, the devices with close current
imbalance values must have minimum current imbalance between them. Table II below shows the predicted current imbalances for the twenty testing devices.

**Table 2. Current Imbalances of test devices as obtained from the model’s prediction**

| Device | CI (%) | Device | CI (%) |
|--------|--------|--------|--------|
| 41     | 2.4885 | 51     | 16.4045|
| 42     | 10.7199| 52     | 11.8968|
| 43     | 4.1265 | 53     | 10.2524|
| 44     | 2.3723 | 54     | 14.4785|
| 45     | 2.0956 | 55     | 1.1138 |
| 46     | 13.0290| 56     | 5.9505 |
| 47     | 1.4588 | 57     | 2.6892 |
| 48     | 10.3182| 58     | 20.2235|
| 49     | 23.4038| 59     | 9.5586 |
| 50     | 16.8281| 60     | 18.4721|

5.2. Experimental Verification

From the predictions shown in Table II, selection for paralleling of devices whose CI values are close results in minimized CI. To test this, a double pulse test (DPT) experiment was conducted with paralleled devices. The test bench for DPT was optimized to minimize circuit parameter mismatches. This ensures that CI is mainly as a result of device parameter mismatches. The test bench for DPT comprised gate driver, dc bus capacitors, two paralleled SiC MOSFET devices (DUT), SiC Schottky Barrier Diode (SBD) as a freewheeling diode, load inductance and discharge resistor. Figure 9 below shows the DPT set-up.

From the Table 2, device 48 and device 53 have close CI values. Paralleling them is expected result in a very low CI. Another set of devices with close CI values are device 50 and device 51. These two sets of devices were tested to verify this assertion. Figure 10 and Figure 11 below are the waveforms of drain current for both pairs of devices during turn-on transient. The transient drain currents for device 48 and device 53 are very close. This implies that when the two devices are paralleled, a good transient current distribution is achieved. Likewise, device 50 and device 51 also have a good transient current distribution when paralleled. The model developed through machine learning technique can therefore aid in selecting devices for paralleling such that transient current distribution is nearly uniform.

Still, according to the model’s prediction, device 47 and device 58 have CI values which are quite far apart. Also, device 49 and device 55 have CI values which are significantly different. Paralleling these sets of devices is expected to result in large transient current imbalances. A DPT experiment was conducted to ascertain this and the results tally with the assertion. Figure 12 and Figure 13 show the turn-on transient current waveforms obtained from paralleling each pair. Again, the transient drain currents are significantly different in this case as expected, from the model prediction. This further validates the effectiveness of the trained model in screening SiC MOSFETs for paralleling.

In order to get a quantitative measure of accuracy of the model, the actual current imbalances for these paralleled pairs of devices were determined. The values were compared with the difference in CI for each device as obtained from the model. It is expected that when two devices are paralleled then the current imbalance should be equal to the difference in their CI values predicted by the model. This is because the model assumes paralleling each of the devices with a common device. Since the model represents a linear relationship between device parameters and current imbalance, with the device parameters remaining constant, this additive nature of current imbalances is feasible. Model’s accuracy is then calculated from test CI and model’s CI as shown in Table 3. In this paper, CI is calculated using (6).
From the experimental results, the developed model can effectively screen devices for paralleling. The predicted CI values can be used to determine which devices to parallel to achieve low transient current imbalance. The devices with small difference in predicted CI values have small actual current imbalance when paralleled. Those with large predicted CI values on the other hand have large actual CI when paralleled. The accuracy of prediction of the model is also satisfactorily high.
6. Conclusion
Using machine learning algorithms to learn the level of influence of device parameters on current imbalance can help screen SiC MOSFET devices for paralleling. This paper has demonstrated implementation of such an approach by using linear regression method to train a model which is later used to screen devices for paralleling. The screened devices are tested using DPT experiment and the performance of the model is found to be satisfactory. Further work will be done to develop a model which can screen devices to achieve both minimum transient and steady state current imbalance when paralleled.

Acknowledgement
The authors of this paper wish to send their gratitude to National R&D Program of China for supporting this research through the project numbered 2018YFB0905703.

References
[1] H. Li et al 2016 Influences of device and circuit mismatches on paralleling silicon carbide MOSFETs, in IEEE Transactions on Power Electronics, vol. 31, no. 1, pp. 621-634.
[2] H. Li, S. Munk-Nielsen, C. Pham and S. Bęczkowski, 2014 Circuit mismatch influence on performance of paralleling silicon carbide MOSFETs, 2014 16th European Conference on Power Electronics and Applications, Lappeenranta, pp. 1-8.
[3] J. Fabre and P. Ladoux, 2016 Parallel connection of 1200-V/100-A SiC-MOSFET half-bridge modules, in IEEE Transactions on Industry Applications, vol. 52, no. 2, pp. 1669-1676.
[4] D. Peftitis et al 2013 Challenges regarding parallel connection of SiC JFETs, in IEEE Transactions on Power Electronics, vol. 28, no. 3, pp. 1449-1463.
[5] J. K. Lim et al 2014, Analysis and experimental verification of the influence of fabrication process tolerances and circuit parasitics on transient current sharing of parallel-connected SiC JFETs, in IEEE Transactions on Power Electronics, vol. 29, no. 5, pp. 2180-2191.
[6] G. Wang, J. Mookken, J. Rice and M. Schupbach, 2014 Dynamic and static behavior of packaged silicon carbide MOSFETs in paralleled applications, 2014 IEEE Applied Power Electronics Conference and Exposition - APEC 2014, Fort Worth, TX, pp. 1478-1483.
[7] J. Fabre and P. Ladoux 2015 Parallel connection of SiC MOSFET modules for future use in traction converters, 2015 International Conference on Electrical Systems for Aircraft, Railway, Ship Propulsion and Road Vehicles (ESARS), Aachen, pp. 1-6.
[8] Y. Mao et al 2017 Balancing of peak currents between paralleled SiC MOSFETs by source impedances, 2017 IEEE Applied Power Electronics Conference and Exposition (APEC), Tampa, FL, pp. 800-803.
[9] Y. Xue et al Active current balancing for parallel-connected silicon carbide MOSFETs, 2013 IEEE Energy Conversion Congress and Exposition, Denver, CO, 2013, pp. 1563-1569 (2013).
[10] S. Bęczkowski et al 2017 Switching current imbalance mitigation in power modules with parallel connected SiC MOSFETs, 2017 19th European Conference on Power Electronics and Applications (EPE'17 ECCE Europe), Warsaw, 2017, pp. P.1-P.8.
[11] I. Goodfellow and Y. Bengio 2016 Deep learning, MIT Press.
[12] N. M. Nasrabadi 2007 Pattern recognition and machine learning, Journal of Electronic Imaging.
[13] J. Ke et al 2018 Influence of parasitic capacitances on transient current distribution of paralleled SiC MOSFETs, IEEE Workshop on Wide Bandgap Power Devices and Applications in Asia - WiPDA Asia 2018, Xi’an, Shaanxi, China.
[14] R. A. Farrugia 2012 Improving motion vector prediction using linear regression, 2012 5th International Symposium on Communications, Control and Signal Processing, Rome, pp. 1-4.