A Gradient Descent Based Efficiency Calculation Method With Learning Rate Adaption

Felix Oberdorf¹, Kevin McFall², Joachim Kempkes³

¹) Felix.Oberdorf@fhws.de, University of Applied Sciences Würzburg-Schweinfurt, Ignaz-Schön-Straße 11, 97421 Schweinfurt, Germany
²) KMcFall@Kennesaw.edu, Kennesaw State University, 1000 Chastain Road, 30144 Kennesaw, USA
³) Joachim.Kempkes@fhws.de, University of Applied Sciences Würzburg-Schweinfurt, Ignaz-Schön-Straße 11, 97421 Schweinfurt, Germany

This paper presents a gradient descent based calculation method with an adaptive learning rate for the maximum efficiency calculation of an electrical machine. The method is applied to a permanent magnet synchronous machine, thus the machine equations are described. Of course the method could be applied to other machine types as well. The presented method reduces some problems, which occur while the usage of a gradient descent based method without learning rate adaption. The results are presented and conspicuous features are discussed.

Keywords: Efficiency, gradient descent, learning rate adaption, optimization

1 Introduction

Because of the increasing integration and thus the limited possibilities for heat dissipation the importance of efficiency calculation of electrical machines is growing. Therefore the power losses and the resulting efficiency have to be studied. The principle calculation of efficiency maps is a well known topic [1], [2], [3], even with saturation effects [4]. The calculations are mainly based on curve- and surface fitting methods to obtain an algebraic equation describing e.g. the dependency of flux linkage and current in direct and quadrature axis \( \psi(I_d, I_q) \).

As an alternative approach in comparison to an exhaustive search a method is presented in [5]. It is based on a gradient descent (GD) optimization function. The
main objective is a tuning of the input parameters torque \( T \) and rotational speed \( n \) of an electrical machine to find the maximum efficiency. The presented method shows good results, with little problems in the optimization process. Because of the chosen learning rate an overtuning for \( T_0 = 150 \text{ Nm} \) and \( n_0 = 1000 \text{ rpm} \) (Figure 1 yellow dotted ellipse) can be observed. The yellow arrows mark the evolution of the predicted efficiency. There was stated, that a fixed learning rate of \( \alpha_\ell = 5 \cdot 10^4 \) was chosen, because learning rates of \( \alpha \geq 8 \cdot 10^4 \) led to convergence problems.

![Figure 1](image)

Efficiency map with a comparison of the evolution of \( \eta \) during the optimization process for \( T_0 = 150 \text{ Nm} \) and different rotational speed starting values \( n_0 \) [5]. The overtuning for \( n_0 = 1000 \text{ rpm} \) is marked with an yellow dotted ellipse.

As a solution this paper presents an improved GD method with a learning rate adaption, which is further called the GDa (\( \alpha \) for alpha – learning rate) method. Therefore in section 2 the machine and loss models are presented. Section 3 includes an overview of the optimization process and the learning rate adaption. In section 4 the results are shown and conspicuous features are discussed.
2 Machine Model

2.1 Main equations

The Torque $T$ of a permanent-magnet synchronous-machine (PMSM) can be described using (1), based on flux linkage and currents in direct and quadrature axis directions as well as the number of pole pairs $p$ [5].

$$ T = \frac{1}{2} \cdot p \cdot (\psi_d \cdot I_q + \psi_q \cdot I_d) $$

(1)

The electromagnetic power $P_{em}$ is further calculated by $T$ and the mechanical angular speed $\omega_{Mech}$. 

$$ P_{em} = T \cdot \omega_{Mech} $$

(2)

2.2 Power loss calculation

An electrical machines power losses can be divided into copper, iron, eddy current, windage and bearing losses. The copper losses are calculated by (3) with $R_{ph}$ as the phase resistance and $I_S$ as the stator phase current.

$$ P_{l,Cu} = 3 \cdot I_S^2 \cdot R_{ph} $$

(3)

The iron losses are calculated with default settings in ANSYS Maxwell® and described as a 3D surface depending on the torque $T$ and the rotational speed $n$. The mechanical losses are adapted from [1] and [6], where measurements with a dummy rotor were performed, including bearing and windage losses. As a simplification the eddy current losses in magnets are neglected. The total losses $P_l$ can be calculated by (4).

$$ P_l = P_{l,Cu} + P_{l,Fe} + P_{l,Mech} $$

(4)

The efficiency $\eta$ is calculated by (5) from the electromagnetic power $P_{em}$ and the total power loss $P_l$.

$$ \eta = \frac{P_1}{P_{em} + P_1} $$

(5)
Based on (5) the efficiency for a given operating point can be calculated. In Figure 2 the efficiency map of the machine is shown with an ellipse highlighting an area with $\eta \approx 90.8\%$. This area is further called the area of maximum efficiency. The target of the optimization (section 3) is to find results within this area. This simplification is assumed on the basis of the calculation effort and further described in section 3.

3 Optimization

3.1 Gradient Descent Method

The gradient descent optimization function is described in [7]. The main purpose of this function is the reaching of a minimum. To apply this to the problem of finding the maximum efficiency a (optimization) loss function is necessary. A common loss function is the Mean Squared Error (MSE) function (6) [7].

$$l(w) = \frac{1}{N} \cdot \sum_{j=1}^{N}(\eta_{\text{pred}}(w, x_j) - \eta_{\text{tar},j})^2$$  \hspace{1cm} (6)

The calculated loss is used for the weight update process. Therefore the update rule (7) [8] is applied.

$$w_t = w_{t-1} - \alpha \cdot \nabla l(w_{t-1})$$  \hspace{1cm} (7)
Equation 7 includes the learning rate $\alpha$ with the purpose of an adaption of the gradient’s slope. Like mentioned in [5] there is a large effect on the behaviour of the optimization process. Hence there are more details in section 3.2.

Within the iterative process the (optimization) loss is calculated and the weights are updated. To avoid protracted calculation times, a specific stop criteria has to be defined. It is calculated by the change of loss (8) and compared to the stop criteria of $\Delta l < 1e^{-12}$. This realizes an applicable trade-off between accuracy and optimization time.

$$\Delta l = |l_{t-1} - l_t|$$ (8)

### 3.2 Learning Rate Adaption

During some research about the GD method there was pointed out, that the stability problems as well as the so called overtuning strongly depends on the learning rate. Especially for learning rate of $\alpha \geq 8 \cdot 10^4$ there occurred problems e.g. no convergence of the calculated efficiency. Further there could be detected the mentioned overtuning. It was assumed that this behaviour of the GD method is mainly depending on the chosen learning rate, especially within the first epochs. Thus the GDa method with a learning rate adaption was implemented.

To explain the GDa method a comparison with the GD method is presented. Therefore the GD method with a fixed learning rate $\alpha_f = 15 \cdot 10^4$ is applied. The results can be seen in Figure 3 a) and Figure 4 a) for different start values of torque $T_0$ and rotational speed $n_0$. During the optimization the learning rate $\alpha_f$ is constant.

While the optimization process with the GDa method the learning rate $\alpha_a$ is adapted and defined by (9). For the example $\alpha_a$ depends on the number of the actual epoch while optimization process $N_{epoch}$.

$$\alpha_a(N_{epoch}) = \begin{cases} 5000, & N_{epoch} < 200 \\ 150000, & N_{epoch} \geq 200 \end{cases}$$ (9)

As defined by (9) the learning rate for the first 199 epochs is $\alpha_a = 5 \cdot 10^3$, within epoch 200 the learning rate is adapted to $\alpha_a = 15 \cdot 10^4$. Beside the chosen definition of $\alpha_a$ there could be applied functional dependencies like linear, quadratic or exponential equations. Further an adaptive learning rate method like ADADELTA [9] could be implemented. The results of the GDa method with the discrete learning rate change (9) are presented in section 4 in comparison to the original GD method.
4 Results

In Figure 3 and Figure 4 the evolutions of $\eta$ are shown for different starting values. The results are shown for a) a fixed learning rate (GD method) and b) an adaptive learning rate (GDa method) respectively.

![Efficiency map with a comparison of the evolution of $\eta$ during the optimization process for $T_0 = 50$ Nm and different rotational speed starting values $n_0$. With a) GD and b) GDa method defined by (9).](image1)

![Efficiency map with a comparison of the evolution of $\eta$ during the optimization process for $T_0 = 150$ Nm and different rotational speed starting values $n_0$. With a) GD and b) GDa defined by (9).](image2)
In Table 1 the optimization results are compared for different starting values and both methods. There are shown the maximum predicted efficiency $\eta_{\text{pred, max}}$, the torque $T_{\text{max}}$ and rotational speed $n_{\text{max}}$ at maximum predicted efficiency as well as the number of epochs $N_{\text{epochs}}$, until fulfilment of the stop criteria.

<table>
<thead>
<tr>
<th>$T_0$</th>
<th>$n_0$</th>
<th>GD method</th>
<th>GDa method</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>$\eta_{\text{pred, max}}$</td>
<td>$T_{\text{max}}$</td>
</tr>
<tr>
<td>50</td>
<td>1000</td>
<td>0.9091</td>
<td>98.8</td>
</tr>
<tr>
<td>150</td>
<td>1000</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>50</td>
<td>2000</td>
<td>0.9091</td>
<td>98.7</td>
</tr>
<tr>
<td>150</td>
<td>2000</td>
<td>0.9090</td>
<td>98.5</td>
</tr>
<tr>
<td>50</td>
<td>3000</td>
<td>0.9094</td>
<td>103.8</td>
</tr>
<tr>
<td>150</td>
<td>3000</td>
<td>0.9094</td>
<td>103.9</td>
</tr>
<tr>
<td>50</td>
<td>4000</td>
<td>0.9086</td>
<td>107.8</td>
</tr>
<tr>
<td>150</td>
<td>4000</td>
<td>0.9087</td>
<td>107.7</td>
</tr>
<tr>
<td>50</td>
<td>5000</td>
<td>0.9086</td>
<td>107.9</td>
</tr>
<tr>
<td>150</td>
<td>5000</td>
<td>0.9087</td>
<td>107.5</td>
</tr>
</tbody>
</table>

As shown in Figure 3 a) the GD method with constant $\alpha_l = 15\cdot10^4$ leads to an overtuning of the evolution of $\eta_{\text{pred}}$. Further in Figure 4 a) can be seen that for $T_0 = 150$ Nm not just an overtuning exist, but also for $n_0 = 1000$ rpm $\eta_{\text{pred}}$ does not convergence. This approves the statement in [5] and the limitation of $\alpha_l < 8\cdot10^4$.

The results of the GDa method are presented in Figure 3 b) and Figure 4 b) as well as in Table 1. At first it is obvious that all the evolutions of $\eta_{\text{pred}}$ result in the area of maximum efficiency $\eta_{\text{max}} > 0.9086$. Thus the GDa method is approved. Further it can be seen that the overtuning is reduced.

A comparison of the resulting values for $T_{\text{max}}$ and $n_{\text{max}}$ (Table 1) shows that the values vary in a little range, which is presumed admissible. The number of epochs until fulfillment of the stop criteria $N_{\text{epochs}}$ is increasing with the GDa method, which is the consequence of the reduced learning rate for the first 200 epochs. Nevertheless is has to be mentioned that the difference of $N_{\text{epochs}}$ GD and GDa method is not exactly the chosen number of 200 epochs. This results from the optimization
process, which leads to results with a small variation. Thus the calculations were repeated multiple times and the presented results could be proved.

5 Discussion

Usually while an optimization process at first a high learning rate is chosen and at a certain point the learning rate is reduced to fine tune the results. The application within the GDa method is different. The GDa method starts with a small learning rate which further is increased. This mainly depends on the equations (7) and (8) and the chosen stop criteria. The loss between the actual epoch and the previous one is calculated by (8). The result is compared with the stop criteria. If the absolute difference in the loss between two epochs is too small, the stop criteria is fulfilled and the optimization process is finished. Actually this should happen in the area of maximum efficiency, since the gradient in this area is small.

Especially around the path of maximum gradient [5], where the evolutions of $\eta_{\text{pred}}$ are located on, the gradient is high. Based on the gradient calculation in (7) the weights are updated strongly. This actually leads to the described overtuning or even no convergence. The reduced learning rate for the first 199 epochs compensates the high gradient calculation and prevents overtuning as well as no convergence. Afterwards the high learning rate allows a computational efficient reaching of the area of maximum efficiency, without a premature finish of the evolution because of the chosen stop criteria.

Conclusions

In this paper the application of the GDa method for calculating the maximum efficiency area $\eta_{\text{max}}$ of an electrical machine was presented. It is a further development of the GD method and has the main benefit of an adaptive learning rate. The GDa method is applied to a PMSM, nevertheless it could be applied to other machine types as well. The power and power loss calculations are based on the equations in section 2. The gradient descent optimization function, the (optimization) loss function and the learning rate adaption are presented in section 3. The GDa optimization results show, that the area of maximum efficiency $\eta_{\text{max}} > 0.9086$ can be found. The described overtuning or not converging evolutions are reduced. Further the results are discussed and features of the GDa method are explained.

Within further research the presented method could be applied to other machine types. Further additional power losses e. g. eddy current losses in magnets should be considered. The method could also be applied to electrical drive systems including power electronics.
References


