Neural Network for Optical Performance Estimation and Advanced Lens Combination

Robert Brüning[1,*], *Michael Verhoek*[1], and *Uwe Lippmann*[1]

[1]Fraunhofer Institute of Applied Optics and Precision Engineering IOF, Albert-Einstein-Str. 7, 07745 Jena, Germany

Abstract. We developed an algorithm to estimate the performance of an optical system based on the errors of its individual components. After a short training period with classical simulated systems, the performance evaluation for tolerancing could be accelerated by a factor of about three million. Additionally, we propose a probability-based sorting algorithm to combine individual, erroneous components in order to compensate for the tolerance budget within the system and increase the overall yield.

1 Introduction

In the last years machine learning approaches have started to be investigated for optical design tasks. Especially, for the generation of good initial systems [1] or starting points for freeform optimization [2] deep learning algorithm have shown promising results.
In this work we explore the potential of neural network learning for system performance evaluation. In addition, an automated sorting approach was investigated, which combines individual, erroneous lenses to compensate for each other's manufacturing deviations to compensate for individual errors and maximize tolerance budgets.
First, we trained a four-layer neural network to estimate the optical performance of the final system based on the individual deviations of each optical surface with respect to the ideal one. Afterwards, we used this algorithm to evaluate the mutual matching performance of the single lenses with each other to estimate the potential yield within a given tolerance budget.
In a second step, we aim to maximize the yield of performing systems within increased tolerance budgets by matching individual lenses whose tolerances are compensating each other. Starting from a heuristic approach, we developed a probability base algorithm for finite lens batch sizes, which enables its applicability even on a continuous and limited lens flow during manufacturing.

2 Neural network for performance estimation

The modulation transfer function (MTF) value for a system of two lenses L_1 and L_2 was learned using a Multi-Layer Perceptron (MLP) neural network [3]. MLPs can efficiently approximate non-linear functions [4]. The MLP consisted of four layers: an input layer with 6 nodes, two hidden layers with 64 and 32 nodes respectively, and an output node. A ReLU (Rectified Linear Unit) activation function was employed after each hidden layer. The MLP was implemented in PyTorch. The Least Absolute Deviations loss function was used. A diagram is shown in Fig. 1. The input layer nodes denote the values of the 6 lens parameters of the two-lens system: the two radii and the thickness for each of the two lenses. The goal is to teach the MLP to output the MTF for the system in the output node. The training dataset consisted of 200+200 instances of L_1 and 200+200 instances of L_2, combining to $200^2 + 200^2 = 80{,}000$ systems. The instances were created by randomly generating L_1 and L_2 lens parameters according to a uniform distribution within a fixed range and computing the MTFs of the combined L_1+L_2 systems using Zemax ray-tracing software. 85% of the dataset, randomly chosen, was used for training, with the remaining 15% used for validation. Various values of the hyperparameters were tested, eventually settling on 500 epochs with a batch size of 50, and an initial learning rate of 10^{-3}, changing to 10^{-4} after 250 epochs.

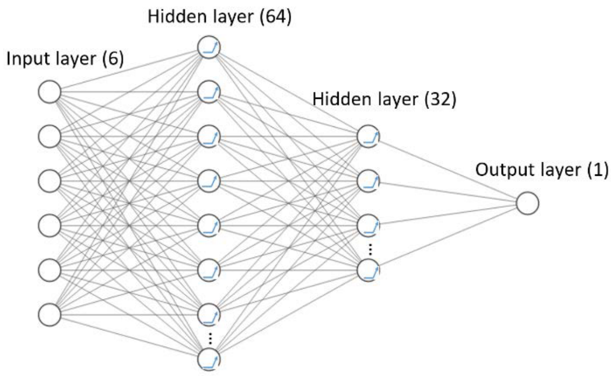

Fig. 1. Diagram of the used four-layer Multi-Layer Perceptron neural network

* Corresponding author: robert.bruening@iof.fraunhofer.de

Evaluating the trained neural network with the remaining 15% of the initial dataset gives a mean relative deviation of about 0.5 percent between the estimated and the simulated MTF value. At the same time calculation of the MTF could be accelerated by a factor of about 3.6 million compared to the classical raytraced based simulation procedure at the same hardware.

3 Lens Sorting Algorithm

The MLP model was used to provide for a method of optimally combining L_1s and L_2s. Suppose these lenses are manufactured with given tolerances of their lens parameters and that these parameters follow a normal distribution independently of each other. The goal is to match up a given L_1 from a pool of L_2s (or vice versa) without explicit knowledge of L_2s produced in the future. To this end, the known probability distributions were utilised. The three-dimensional lens parameter spaces of L_1 and of L_2 were divided into 33 × 33 × 33 bins each. The probability of each bin occurring was calculated using the CDF (cumulative distribution function) for the normal distribution. For each specific L_1, the MTF was calculated using the learned MLP, between the L_1 lens parameters and the central parameters of each L_2 bin. For a given MTF threshold, the bins were selected that satisfied the threshold, and their probabilities were added up. This gave the sum probability that a sufficiently good L_2 can be found for the specific L_1. The algorithm can be repeated with L_1 and L_2 swapped.

Two further datasets were used: one Zemax dataset exactly like the earlier 80,000-system dataset with the exception that the lens parameters were drawn randomly according to a normal distribution (a new MLP was trained on this dataset); and one MLP-generated dataset, in which 10,000 L_1 parameter sets and 10,000 L_2 parameter sets were randomly generated according to the same normal distribution, but the MTF values of the combined 100,000,000 systems were calculated using the trained MLP rather than Zemax.

The number of successful systems, defined by an MTF threshold, that can be combined from the set of L_1s and L_2s was produced according to the probability method as follows: a pool is a subset of the full dataset with N_p L_1 lenses and N_p L_2 lenses. Within this pool, for each L_1, the probability $P(L_1)$ of finding a successful L_2 was calculated according to the above probability algorithm, and vice versa for $P(L_2)$. For each of the N_p^2 combinations, the L_1 and L_2 probability values were multiplied, $P_{12} = P(L_1)P(L_2)$. Also, for each of the N_p^2 combinations, the MTF value was calculated using the trained MLP. The combinations where the MTF values were above the threshold, the P_{12} were compared and the combination with the smallest P_{12} was selected. The rationale is that we want to find matches with good MTF values from unlikely lenses, since such matches are the least likely to be needed for future lenses. The newly matched lenses were removed from the pool, which was refilled with the next in line from the larger dataset. One danger was that the pool fills up with lenses that would never find a match. To avoid this, a removal limit was set: when the number of L_1s or L_2s in the pool that satisfies the removal criteria exceeded this limit, one L_1 or L_2 was discarded from the pool, which was then refilled from the larger dataset. An L_1 satisfies the removal criteria if its MTF is below the MTF threshold for all L_2 in the pool or its P_{12} is zero for all L_2 in the pool (and vice versa).

This method was performed for a range of values for N_p and the removal limit, and the number of successful systems that was yielded, was compared to the number of successful systems according to a heuristic combination algorithm that did have advance knowledge of all the lenses in the dataset. Depending on the pool size N_p, which varies between 0.1% and 1% of complete lens set, our proposed probability method was able to find between 92% and 96% of the successful systems combinations, respectively.

4 Conclusion

We have demonstrated that MLP neural networks can be successfully used to estimate optical performance parameters for erroneous lens simulation as typically during a tolerance analysis. Additionally, the speed advantage for system evaluation algorithm enables to train and prove a probability-based sorting algorithm, which enables the combination of erroneous lenses to a successful optical system in real time.

References

1. G. Cote, J. La Londe, S. Thibault, Opt. Express, Vol. 27, No. 20, 28279-28292 (2019)
2. T. Yang, D. Cheng, Y. Wang, Opt. Express, Vol. 27, No. 12, 17228-17237 (2019)
3. T. Hastie, R. Tibshirani, J. Friedman. The Elements of Statistical Learning: Data Mining, Inference, and Prediction. Springer, New York, NY (2009)
4. G. Cybenko. Math. Control Signals Syst., Vol. 2, No. 4, 303-314 (1989)