Particle swarm optimization hyperparameters tuning for physical-model fitting of VCSEL measurements

(Article begins on next page)

Particle Swarm Optimization Hyperparameters Tuning for Physical-Model Fitting of VCSEL Measurements

Andrea Marchisio[a], Enrico Ghillino[b], Vittorio Curri[a], Andrea Carena[a], and Paolo Bardella[a]

[a]Dipartimento di Elettronica e Telecomunicazioni, Politecnico di Torino, Corso Duca degli Abruzzi 24, 10129 Torino, Italia

[b]Synopsys Photonic Solutions, 400 Executive Blvd, Ossining, NY 10562, United States

ABSTRACT

We propose the use of the Particle Swarm Optimization (PSO) algorithm for the direct extraction of Vertical-Cavity Surface-Emitting Laser (VCSEL) parameters compatible with a rate equation-based model that takes into account the thermal effects. PSO is an evolutionary algorithm that drastically reduces the computational cost and time with respect to traditional brute-force approaches, thanks to the "swarm intelligence" of the optimization agents (called "particles"). With an optimal choice of the initial hyperparameters of the algorithm, the method is shown to predict parameters that accurately reproduce the nonlinear behavior of the device, as well as its complicated thermal effects.

Keywords: Vertical-cavity surface-emitting lasers, parameter extraction, evolutionary algorithms, particle swarm optimization

1. INTRODUCTION

Vertical-cavity surface-emitting lasers (VCSELs) offer multiple advantages such as a small footprint, high efficiency, and simple manufacturing, which make them a prime choice as coherent sources in a number of different applications, including data communications, sensing, and photonic integrated circuits.[1]

Among these beneficial practical aspects, extremely complex quantum and thermal effects regulate the physical behavior of such devices. Due to the presence of inherent nonlinearities of various nature, in order to predict the behavior of VCSELs, it is possible to use very accurate multiphysics models[2] or simple rate-equation-based models devised for circuit-level simulations;[3] in both cases, a large number of unknown parameters have to be considered to reproduce the experimental findings. That is why it is important to possess a reliable method for extracting such parameters from readily-available data, such as power and frequency response measurements. This task could be solved by employing brute-force techniques, where the entire solution space is scanned to find a suitable match. Clearly, this approach is extremely time-consuming and computationally expensive, especially in the context of VCSEL parameter extraction, since the number of involved parameters (and thus the solution space itself) is large. For this reason, it is possible to employ two different approaches that enable a more efficient search of the solution space: Machine Learning (ML) or optimization techniques (e.g., evolutionary algorithms).

Since the effectiveness of training an artificial neural network (ANN) for ML is linked to the ability to generate large data sets, requiring either a large amount of measurements and devices or substantial computational power to follow a simulation approach, in this work, we will further expand the idea of directly extracting the parameters of the VCSEL model from curves using an evolutionary algorithm called Particle Swarm Optimization (PSO).[4] In particular, employing a more advanced implementation of PSO, called Adaptive PSO (APSO),[5] we want to extract a set of parameters compatible with the rate equation-based model implemented in Synopsys OptSim™,[6] thus creating a workflow that could enable simpler system-level simulations after the characterization of the unknown laser source.

After the analysis of the impact of the initial values of the APSO hyperparameters on its performance, we apply the proposed method to predict a set of physical parameters able to accurately reproduce the device behaviour. The performance of the APSO is also compared to that of the traditional PSO,[7] to underline its benefits, despite the more complex implementation and rules.

Further author information: Andrea Marchisio
E-mail: andrea_marchisio@polito.it, Telephone: +39 011 090 4030

Vertical-Cavity Surface-Emitting Lasers XXVIII, edited by Chun Lei, Kent D. Choquette, Proc. of SPIE Vol. 12904, 129040N · © 2024 SPIE · 0277-786X · doi: 10.1117/12.3002576

2. THE VCSEL MODEL

In this work, a well-established rate equation-based model[3] is employed to describe the physics of VCSELs. It contains rate equations for the evolution of carriers and photons over time, as well as empirical equations to capture the thermal effects.

In this model, the distribution of carriers in the radial direction r is expressed through its expansion in a two-term Bessel series, which allows us to neglect the spatial dependency of carrier and photon numbers:[3]

$$N(r,t) = N_0(t) - N_1(t) J_0(\sigma_1 r/R) \tag{1}$$

with σ_1 first nonzero root of J_1, J_0 and J_1 Bessel functions of the first kind, and R effective radius of the active layer. The temporal evolution of $N_0(t)$, $N_1(t)$, and the photon number is given by the following spatially independent rate equations:

$$\frac{\mathrm{d}N_0}{\mathrm{d}t} = +\frac{\eta_\mathrm{i} I}{q} - \frac{N_0}{\tau_\mathrm{n}} - \frac{I_\mathrm{l}(N_0,T)}{q} - \frac{G(T)\left[\gamma_{00}(N_0 - N_\mathrm{tr}(T)) - \gamma_{01}N_1\right]}{1+\epsilon S} S \tag{2}$$

$$\frac{\mathrm{d}N_1}{\mathrm{d}t} = -\frac{N_1}{\tau_\mathrm{n}}(1+h_\mathrm{diff}) + \frac{G(T)\left[\phi_{100}(N_0 - N_\mathrm{tr}(T)) - \phi_{101}N_1\right]}{1+\epsilon S} S \tag{3}$$

$$\frac{\mathrm{d}S}{\mathrm{d}t} = -\frac{S}{\tau_\mathrm{p}} + \frac{\beta_\mathrm{sp} N_0}{\tau_\mathrm{n}} + \frac{G(T)\left[\gamma_{00}(N_0 - N_\mathrm{tr}(T)) - \gamma_{01}N_1\right]}{1+\epsilon S} \tag{4}$$

with η_i injection efficiency, I injected current, q electron charge, τ_n carrier lifetime, T temperature, $G(T)$ gain coefficient, $N_\mathrm{tr}(T)$ transparency carrier number, $I_\mathrm{l}(N_0,T)$ leakage current, ϵ gain compression factor, h_diff diffusion coefficient, τ_p photon lifetime and β_sp spontaneous emission coefficient. The coefficients γ_{00}, γ_{01}, ϕ_{100}, ϕ_{101} quantify the overlap between the fundamental transverse mode and the active region. The output power P_out is proportional to S through a suitable coupling coefficient k_f.

Concerning the temperature dependency, we exploited the modified empirical equations for gain, transparency number, and leakage current of 4:

$$G(T) = G'_0 \frac{1 + a'_{\mathrm{g}_1} T + a'_{\mathrm{g}_2} T^2}{1 + b'_{\mathrm{g}_1} T + b'_{\mathrm{g}_2} T^2} \tag{5}$$

$$N_\mathrm{tr}(T) = N'_{\mathrm{tr}_0}\left(1 + C'_{\mathrm{n}_1} T + C'_{\mathrm{n}_2} T^2\right) \tag{6}$$

$$I_\mathrm{l}(N_0,T) = I_{\mathrm{l}_0} \exp\left(\frac{-a_0 + a_1 N_0 + a_2 N_0 T - a_3/N_0}{T}\right) \tag{7}$$

3. THE PSO ALGORITHM

PSO is an evolutionary algorithm where the optimization is performed by a swarm of N_p agents called "particles" that move in a N-dimensional solution space. For the current problem, each dimension of the solution space represents one of the unknown parameters to extract, and it is bounded in the ranges reported in Tab. 1. Convergence to the target is made possible according to motion rules depending on the best position found by each particle and on the global best position found by the swarm. For each j-th particle, at the k-th iteration, the velocity and the positions are computed as follows:[7]

$$\mathbf{v}_j^{k+1} = c_\mathrm{i}\mathbf{v}_j^k + c_\mathrm{c} r_1(\mathbf{p}_j^k - \mathbf{x}_j^k) + c_\mathrm{s} r_2(\mathbf{p}_\mathrm{gl}^k - \mathbf{x}_j^k) \tag{8}$$

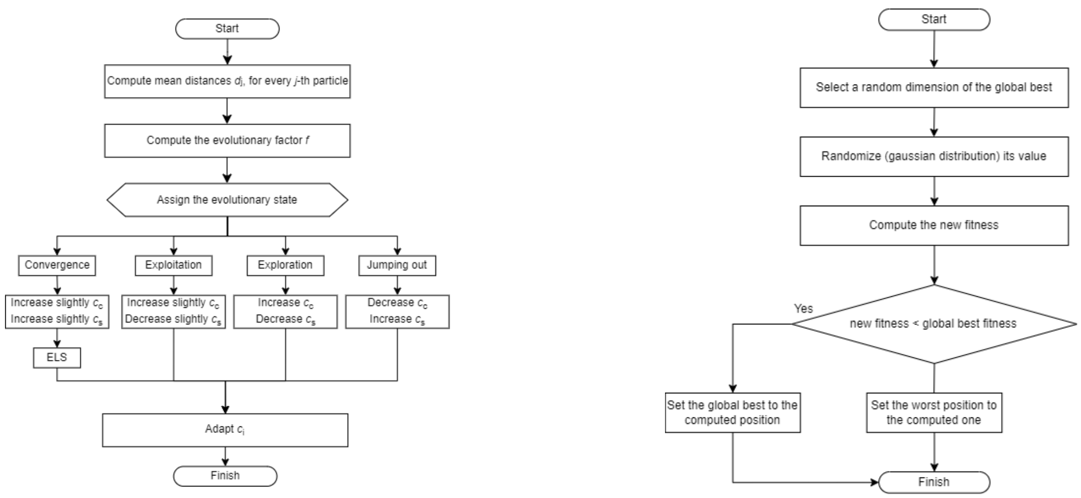
$$\mathbf{x}_j^{k+1} = \mathbf{x}_j^k + \mathbf{v}_j^{k+1} \tag{9}$$

with c_i inertia coefficient, c_c cognitive acceleration coefficient, c_s social acceleration coefficient, r_1 and r_2 random scaling factors, $\mathbf{p}_j^k$ personal best position for the j-th particle, and $\mathbf{p}_\mathrm{gl}^k$ global best position.

One of the most problematic aspects of PSO is related to the choice of the hyperparameters. Indeed, there are not fixed values for the velocity coefficients that ensure convergence and the optimum can change depending on the problem. Moreover, a higher number of particles and iterations could lead to better results, but, of course,

Table 1: Investigated VCSEL parameters, with variation ranges and values of the target device.

Parameters	**Range**	**Parameters**	**Range**
Injection efficiency η_i	0.70 to 1.00	Transp. num. N'_{tr_0}	2.00×10^6 to 1.00×10^8
Power coeff. k_f (nW)	10.00 to 60.00	Transp. num. coeff. C'_{n_1} (kK^{-1})	-100.00 to -1.00
Carrier lifetime τ_n (ns)	0.50 to 5.00	Transp. num. coeff. C'_{n_2} (kK^{-2})	0.00 to 100.00
Photon lifetime τ_p (ps)	1.50 to 3.50	Leakage current factor I_{l_0} (A)	1.00 to 2.00
Gain coeff. G'_0 (ms^{-1})	-360.0 to -11.1	Leakage current coeff. a_0 (K)	2.00×10^3 to 1.00×10^4
Gain coeff. a'_{g_1} (kK^{-1})	-5.00 to -0.50	Leakage current coeff. a_1 (K)	0.00 to 3.00×10^{-4}
Gain coeff. a'_{g_2} (kK^{-2})	-50.00 to -2.00	Leakage current coeff. a_2	1.00×10^{-9} to 4.00×10^{-8}
Gain coeff. b'_{g_1} (kK^{-1})	-100 to 0	Diffusion parameter h_diff	1.00 to 20.00
Gain coeff. b'_{g_2} (kK^{-2})	5.56 to 900.0	Thermal impedance R_th (K/W)	5.00×10^2 to 8.00×10^3
Gain saturation factor ϵ	1×10^{-6} to 3×10^{-6}	Spont. emission coeff. β_sp	1×10^{-5} to 9×10^{-6}



(a) (b)

Figure 1: Flowcharts for the adaptation of the velocity coefficients (a) and for the ELS (b) as prescribed by 5.

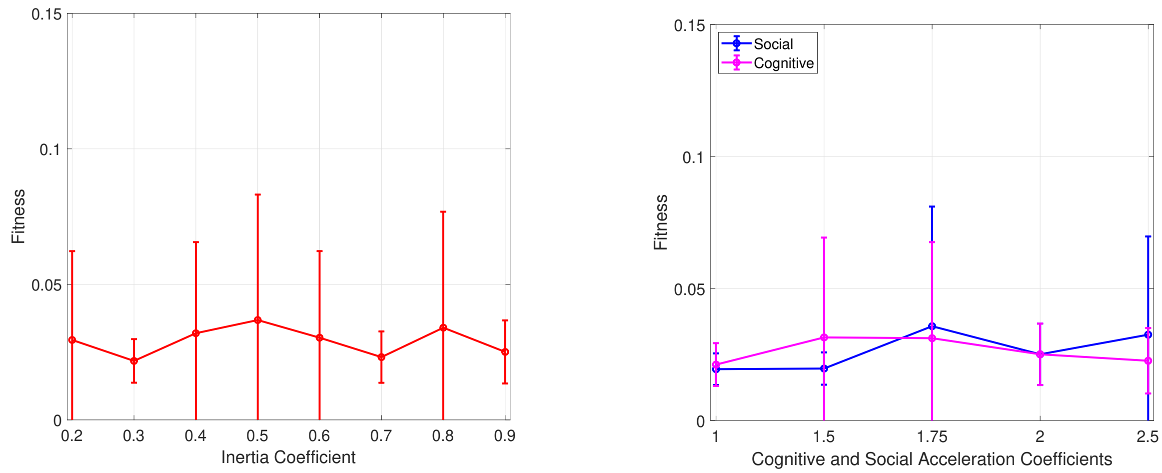
the optimization will require more time and computational power to complete. Therefore, alternative techniques such as APSO can be exploited for more consistent performances, in order to avoid suboptimal choices that could easily result in an optimizer that converges prematurely in local minima or one that does not converge at all.

The idea is, given a number of particles and of iterations, not to choose fixed values for the velocity coefficients but to *adapt* them during the optimization, depending on the behavior of the swarm at the current iteration ("evolutionary state"). According to 5, at each iteration, the evolutionary state must be estimated. In order to do so, for each j-th particle, we compute the average distance d_j with respect to all others and with them we define an "evolutionary factor" f:

$$f = \frac{d_\mathrm{gl} - d_\mathrm{min}}{d_\mathrm{max} - d_\mathrm{min}} \tag{10}$$

with d_gl average distance from the global best particle, d_min and d_max minimum and maximum d_j's. With this f parameter we are able to evaluate the evolutionary state by means of a fuzzy assignment to four possible swarm conditions:

1. *Exploration*: the swarm is exploring the solution space and the particles are not concentrated in one single region (medium to large f);
2. *Exploitation*: a global best was found and it is attracting the rest of the swarm (smaller f);
3. *Convergence*: the swarm converged to the current global best and it is exploring its near vicinity (minimal f);



(a) (b)

Figure 2: (a) Effect of the initial inertia value on the final fitness. (b) Effect of the initial social and cognitive acceleration values on the final fitness.

4. *Jumping out*: the new global best is far away from the rest because the swarm is exiting the current minimum (large f).

A detailed explanation of the workings of the APSO will be omitted here, but it is available in 5. For completeness, Fig. 1 contains the flowcharts for the main additional operations of the algorithm.

4. RESULTS

According to 5, a set of APSO hyperparameters that works empirically well is $c_\mathrm{i} = 0.9$ and $c_\mathrm{s} = c_\mathrm{c} = 2.0$. These are just their initial values, since they are subsequently modified according to the rules mentioned in the previous section. In order to test whether this "rule of thumb" applies to the problem we are tackling and to evaluate if there are better choices, we decided to start by analyzing the effect of the initial hyperparameters values on the final convergence of the algorithm. The optimization target is a set of three experimental L-I curves at three different temperatures (25 °C, 40 °C, 60 °C), reported in 3. The error (*fitness*, in PSO jargon) is evaluated as:

$$\Delta = \frac{||\text{prediction} - \text{target}||}{||\text{target}||} \tag{11}$$

The results of this analysis are reported in Fig. 2(a) and Fig. 2(b). All three curves represent the average error at the corresponding initial value of the parameter. The error bars represent the standard deviation from these average values for the multiple runs at different random seeds: since the result of the optimization depends on the random initialization and the random parameters in the velocity rules, it is important to take into account this effect by repeating the optimization multiple times with different random seeds (in this case 50). In order to generate these graphs, we modified one parameter at the time while keeping the others fixed at the values suggested by 5. The APSO is executed with $N_\mathrm{p} = 200$ particles and $N_\mathrm{s} = 200$ steps. In general, the average value is almost constant, but if we consider the standard deviation from the average, it is possible to observe that $c_\mathrm{i} = 0.9$, $c_\mathrm{s} = 2.0$, and $c_\mathrm{c} = 2.0$ provide small average values with small standard deviations. Despite not being the absolute best, it is for sure a safe choice without a prior analysis of the effect of the hyperparameters on the optimization.

At this point, using $c_\mathrm{i} = 0.9$ and $c_\mathrm{s} = c_\mathrm{c} = 2.0$, we applied the method directly to the three experimental L-I curves from 3 without a fixed random seed. The APSO method predicts the parameters of Tab. 1, which are then used in the model to obtain the curves of Fig. 3. As it is possible to appreciate from Fig. 3, the

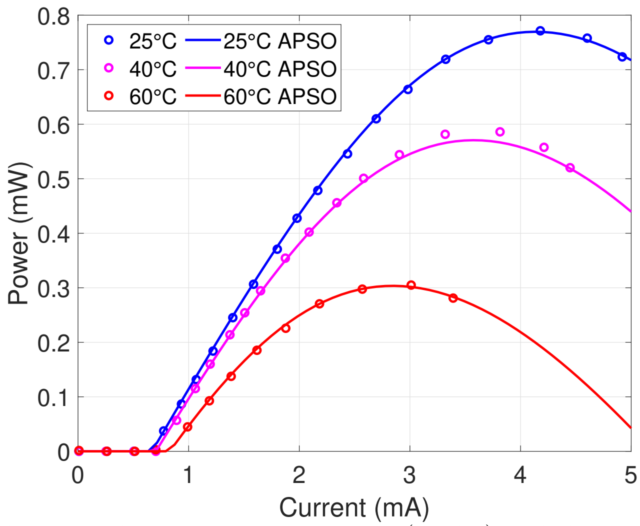


Figure 3: Comparison between the target measurements (circles) and the predicted curves (solid lines).

parameters obtained from the APSO algorithm closely reproduce the experimental ones, capturing the various effects such as the thermal roll-over, the thermal-dependent slope change, the modified threshold current, etc. The results obtained with APSO even surpass the set of parameters reported in 3, obtained by means of a numerical optimization.

5. CONCLUSIONS

In this work, we presented a method based on the Adaptive Particle Swarm Optimization algorithm to extract physical-model parameters directly from the L-I curves of a VCSEL to be characterized. After the analysis of the effect of the initial choice of hyperparameters on convergence, we showed how the method is able to extract a set of parameters that accurately reproduce the behavior of a real device. As we suggested in our previous work,[4] this proves that the method is reliable not only if applied to ideal simulated curves, but also if the target is a set of non-ideal measurements. With the improvements to the code, we also managed to reduce the optimization time from 16 min to 20 s, thus making the algorithm even more powerful.

ACKNOWLEDGMENTS

A. Marchisio's PhD scholarship is co-funded by the Italian Ministry of University and Research (MUR) D.M. n. 352/2022 project funds and by Synopsys Inc., in the framework of the Italian *Piano Nazionale di Ripresa e Resilienza* and the European *NextGenerationEU*.

REFERENCES

[1] Liu, A., Wolf, P., Lott, J. A., and Bimberg, D., "Vertical-cavity surface-emitting lasers for data communication and sensing," *Photon. Res.* **7**(2), 121–136 (2019).

[2] Tibaldi, A., Bertazzi, F., Goano, M., Michalzik, R., and Debernardi, P., "Venus: A vertical-cavity surface-emitting laser electro-opto-thermal numerical simulator," *IEEE Journal of Selected Topics in Quantum Electronics* **25**(6), 1–12 (2019).

[3] Mena, P., Morikuni, J., Kang, S.-M., Harton, A., and Wyatt, K., "A comprehensive circuit-level model of vertical-cavity surface-emitting lasers," *Journal of Lightwave Technology* **17**(12), 2612–2632 (1999).

[4] Marchisio, A., Ghillino, E., Curri, V., Carena, A., and Bardella, P., "A particle swarm optimization assisted approach for the extraction of vcsel model parameters," *Optics Letters* (2023).

[5] Zhan, Z.-H., Zhang, J., Li, Y., and Chung, H. S.-H., "Adaptive particle swarm optimization," *IEEE Trans. Syst., Man., Cybern. B, Cybern* **39**(6), 1362–1381 (2009).

[6] Synopsys OptSim, "OptSim User Guide." https://www.synopsys.com/photonic-solutions/pic-design-suite.html.

[7] Kennedy, J. and Eberhart, R., "Particle swarm optimization," in [*Proceedings of ICNN'95 - International Conference on Neural Networks*], **4**, 1942–1948 vol.4 (1995).