Meta-heuristic Optimization of Control Structure and Design for MMC-HVdc Applications

Carolin Hirsching, Steven de Jongh, Daniela Eser, Michael Suriyah and Thomas Leibfried
Insitute of Electrical Energy Systems and High Voltage Engineering (IEH)
Karlsruhe Institute of Technology (KIT)
Karlsruhe, Germany
{carolin.hirsching, steven.dejongh, daniela.eser, michael.suriyah, thomas.leibfried}@kit.edu

***Abstract*—This contribution applies meta-heuristic optimization algorithms for the optimization of controller implementations for modular multilevel converters in HVdc applications.**

The proposed toolbox allows optimization of parameters and structure of the controllers with arbitrary user-defined objective functions related to macroscopic as well as microscopic stability properties of the converter itself as well as within the overall system. Different meta-heuristic algorithms are compared with respect to calculation efficiency and achieved optimized values.

***Index Terms*—control design, converter control, EMT, meta-heuristic optimization, passivity-based stability assessment**

I. Introduction

With an emerging rate of power electronic based transmission units planned within the power grid as, for instance, modular multilevel converters (MMC), thorough studies regarding their transient behaviour as well as screening studies related to their frequency characteristic have gained tremendous importance [1]. Both aspects are directly impacted by the design as well as structural implementation of converter controllers as shown for different inner control structures and control modes in e.g. [1]–[4], respectively. This opens the research question on how to optimize control design and implementations of a converter with respect to several stability properties affecting the control accuracy and overall system stability after installation.

Meta-heuristics offer algorithmic frameworks to solve complex optimization problems. Their main applications include the optimal design in the areas of parametric, topological and combinatorial problems. Comprehensive literature and overviews are widely available [5]–[7]. In contrast to white-box optimization problems where an exact mathematical model of the underlying system is applied, meta-heuristics aim at solving a potentially high-dimensional black-box system. While white-box models allow the application of methods that analytically or numerically optimize a known mathematical model, in meta-heuristics only a simulation or real-world experiment is given. [8] reviews meta-heuristic algorithms for parameter optimization of power converters for tasks regarding power quality and waveform related issues, control tuning as well as circuitry. The paper reviews the application of meta-heuristics to general tasks in converter optimization but has no distinct focus on HVdc applications.

The purpose of this paper is to expand the idea of control design related optimization by an additional optimization of the implemented control structure as, for instance, feed-forward or setpoint connection paths. Therefore, classic control design methods are not applied.

The approach chosen within this paper makes use of an MMC model implemented in the software PSCAD for electromagnetic transient (EMT) studies. This has the advantage that non-linearities of relevant equipment (e.g. transformers and power electronics) or controllers (e.g. limiters, saturation) may be included [9], [10]. Moreover, grey-boxed models might be taken into account.

Therefore, a toolbox is developed which allows an interface between meta-heuristic optimization algorithms implemented in python and an EMT model of an MMC-HVdc link with an arbitrary control structure implementation in PSCAD. The optimizer creates solutions with parametric and structural information for controllers which is transferred to the EMT simulation environment. Here, the resulting behaviour is recorded and a certain objective function value is obtained with respect to a user-defined desired behaviour which is subsequently returned to the optimization algorithms. Hence, the objective function defines the optimization idea and performance.

As elaborated in previous research [4], different controller implementations may result in similar step responses in time domain, but lead to different passivity and damping properties of the converter within the frequency domain. Therefore, the introduced toolbox enables to optimize control structure and design related aspects in the time as well as the frequency domain. The aim is to obtain a controller implementation and design with optimized transient, damping as well as passivity properties. Thus, the introduced toolbox can be utilized to eliminate or minimize macroscopic instability problems over a defined frequency range with respect to a desired control accuracy of the observed converter model itself.

For the optimization of controller parameters 18 different meta-heuristic optimization algorithms [11]–[27] from the classes of evolutionary, physically, biologically and system- and human-inspired as well as swarm based algorithms are applied. The developed tool allows the parallelized calculation

of multiple parameter sets to decrease optimization time. Meta-heuristic algorithms are evaluated with respect to obtained optimized values, time efficiency as well as the proposed optimization setting.

The remainder of this paper is organized as follows. Section II introduces the basics for the optimization framework with respect to the meta-heuristic optimization procedure, considered optimization algorithms and the considered EMT model. The implementation and mathematical description of the optimization framework is provided in Section III whereas Section IV describes the development of the objective function. Section V evaluates the optimization performance and Section VI concludes the paper.

II. Optimization Toolbox Basics

This contribution introduces an optimization framework for arbitrary control structure implementations of an MMC. An introduction of the applied optimization techniques as well as the underlying EMT converter and control model for the toolbox are provided in the following.

A. Meta-heuristic optimization

Fig. 1 shows the general interaction between the system (called *environment*) and the optimization solver.

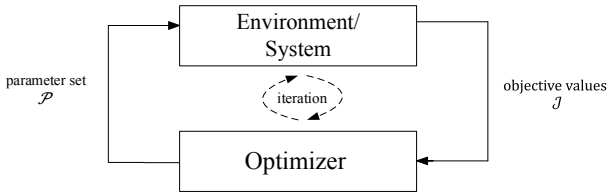

Fig. 1. General Interaction during meta-heuristic optimization procedure

Each iteration consists of a candidate population $\mathcal{P}$ that comprises multiple parameter sets and is passed to the environment. The environment can be an arbitrary parameterized system model that can be directly queried (e.g. tested in an experiment) or simulated on a computer. Given the candidate population $\mathcal{P}$, the environment is evaluated for each provided parameter set in $\mathcal{P}$, yielding a set of respective objective values $\mathcal{J}$. These objective values are then passed to the optimizer which calculates the next parameters to be evaluated. This procedure is applied iteratively with the goal of finding a parameter configuration that minimizes a certain user-defined objective function. Note that in several applications the optimization problem is formulated as a maximization of an objective function. To ensure consistency with typical formulations of objective functions in engineering, we make use of the notation as a minimization of an objective function. However, an equivalent formulation is possible by multiplying the objective by minus one and turning the task into a maximization problem (e.g. maximizing a fitness function). Ideally, the optimizer finds the global optimum in the shortest possible time. In reality, however, there is no algorithm that is the *best* optimizer for all classes of problems [28]. This opens the research question to find the most suitable solver for a given problem class. The task of finding a solution to a given objective function faces a certain trade-off between *exploitation* and *exploration* which varies for different optimization tasks. These properties are described as:

- *exploitation*: Using information from previous iterations to focus on a region with low objective function values
- *exploration*: Exploring different (previously unvisited) regions of the solution space

A fully exploring algorithm (e.g. sampling uniformly randomly from the solution space) might take too long to find a suitable solution. A fully exploiting algorithm, however, can get easily stuck in local minima of the objective function. Finding a good trade-off between these two criteria for the given problem class is an important procedure when applying meta-heuristic algorithms. The various meta-heuristic optimization algorithms can be, based on their source of inspiration, divided into different categories. In this paper, common implementations of algorithms from the literature of the categories *swarm based*, *evolutionary algorithms*, *physically, biologically, system* and *human inspired* are applied and compared with respect to their optimization performance on the given task. The choice of the respective algorithms is done based on the availability of open-source implementations, as well as the performance of these algorithms in benchmark optimization problems. An overview of these algorithms and their respective implementations is shown in Tab. I.

TABLE I
Overview of applied meta-heuristic optimization algorithms

Category	Algorithm	Acronym
swarm based	Particle Swarm Optimization [11]	PSO
	Phasor Particle Swarm Optimization [12]	PPSO
	Self-organising Hierarchical PSO with Time-Varying Acceleration Coefficients [13]	HPSO
	Chaos Particle Swarm Optimization [14]	CPSO
	Whale Optimization Algorithm [15]	WOA
	Grey Wolf Optimizer [16]	GWO
	Random Walk Grey Wolf Optimizer [17]	RW-GWO
evolutionary	Differential Evolution [18]	JADE
physically	Nuclear Reaction Optimization [19]	NRO
	Equilibrium Optimizer [20]	EO
biologically	Virus Colony Search [21]	VCS
system inspired	Artificial Ecosystem-based Optimization [22]	AEO
	Adaptive Artificial Ecosystem-based Optimization [23]	Ad-AEO
	Enhanced Artificial Ecosystem-based Optimization [24]	Enh-AEO
human inspired	Teaching-Learning-based Optimization [25]	TLO
	Social Ski Diver Optimization [26]	SSDO
	SSDO & Levy Flight [23]	LevySSDO
	Forensic-Based Investigation Optimization [27]	FBIO

B. EMT model

The idea of the presented toolbox is to optimize the behaviour of an arbitrary MMC control system. Hence, a 525 kV HVdc scheme in symmetric monopolar configuration according to Fig. 2 is considered as environment within the optimization procedure. An MMC converter and control

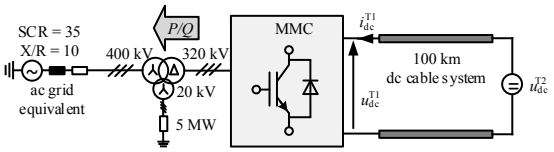

Fig. 2. Overview of the considered EMT model.

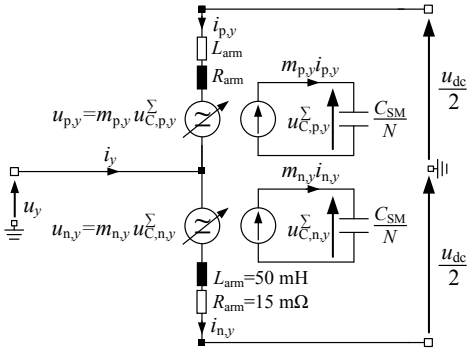

Fig. 3. Single phase view of the utilized MMC average value model.

model of the 1.3 GVA station T1 is implemented in detail as well as a frequency-dependent phase model of a 100 km dc cable system. The cable terminal is connected to a controlled dc source u_{dc}^{T2} providing the output of dc side controls at station T2. The rated power of the dc link is 1 GW.

The MMC is modeled as an Average Value Model based on switching functions according to [9] which is applicable to high level control system design studies as undertaken within this contribution. A single phase representation of the MMC model is depicted in Fig. 3 valid for each phase $y \in \{1, 2, 3\}$. Utilizing mesh analysis and neglecting ohmic losses, the following equations are derived for the positive and the negative arm in each phase y:

$$u_y + L_{\mathrm{arm}} \cdot \frac{\mathrm{d}}{\mathrm{d}t} i_{\mathrm{p},y} + u_{\mathrm{p},y} - \frac{u_{\mathrm{dc}}}{2} = 0 \tag{1}$$

$$u_y - L_{\mathrm{arm}} \cdot \frac{\mathrm{d}}{\mathrm{d}t} i_{\mathrm{n},y} - u_{\mathrm{n},y} + \frac{u_{\mathrm{dc}}}{2} = 0 \tag{2}$$

Arm capacitor voltage $u_{\mathrm{C},s,y}^{\Sigma}$ corresponds to the sum of all capacitor voltages (2.5 kV each) in one arm, whereas $s \epsilon \{\mathrm{p}, \mathrm{n}\}$ denotes quantities for positive and negative arms, respectively. Arm voltages $u_{s,y}$ depend on the modulation index $m_{s,y}$ as well as the arm capacitor voltages in each arm. Charging dynamics of all $N = 270$ submodule capacitors C_{SM} (8.5 mF each) are modelled according to (3). The phase module current $i_{\mathrm{phm},y}$ as well as ac current i_y are introduced in (4).

$$u_{s,y} = m_{s,y} u_{\mathrm{C},s,y}^{\Sigma}, \qquad \frac{C_{\mathrm{SM}}}{N} \frac{\mathrm{d}}{\mathrm{d}t} u_{\mathrm{C},s,y}^{\Sigma} = m_{s,y} i_{s,y} \tag{3}$$

$$i_{\mathrm{phm},y} = \frac{i_{\mathrm{p},y} + i_{\mathrm{n},y}}{2}, \qquad i_y = i_{\mathrm{n},y} - i_{\mathrm{p},y} \tag{4}$$

Now, arm delta voltages as well as arm sum voltages are introduced according to (5) and (6).

$$u_{\Delta,y} = \frac{u_{\mathrm{n},y} - u_{\mathrm{p},y}}{2} = \frac{L_{\mathrm{arm}}}{2} \frac{\mathrm{d}}{\mathrm{d}t} i_y + u_y \tag{5}$$

$$u_{\Sigma,y} = \frac{u_{\mathrm{p},y} + u_{\mathrm{n},y}}{2} = -L_{\mathrm{arm}} \frac{\mathrm{d}}{\mathrm{d}t} i_{\mathrm{phm},y} + \frac{u_{\mathrm{dc}}}{2} \tag{6}$$

Here, (5) represents the ac side system dynamics of an MMC, (6) reflects internal and dc side dynamics.

C. MMC controls

Within this contribution, the introduced optimization methodology is utilized to find optimized controller parameters as well as controller structures in grid-following converters. A grid-following converter follows the voltage of the connected ac grid in amplitude and frequency and adapts its control values to provide a desired active and/or reactive current. Here, several control methodologies exist [9], [10], [29], whereas the hierarchical structure of the main control functionalities is depicted in Fig. 4(a).

Within the outer control layer, the user decides if optimization is undertaken for the converter station in u_{dc}/Q control mode or P/Q control mode i.e. within this layer the user specifies if the detailed station T1 provides the dc voltage for the dc link (u_{dc}/Q control) and acts as a master or if the station controls the dc current and hence, defines the power transfer within the dc link (P/Q control). For more detailed explanations regarding the control modes, the reader is referred to [9]. As the chosen control mode is applied for the detailed station T1, the dc side of the dc station equivalent T2 ($u_{\mathrm{dc}}^{\mathrm{T2}}$) is controlled the other way with controller output $2u_{\Sigma,0}^{\mathrm{ref}}$.

Fig. 4(b) gives an overview of the considered MMC control options. Measured quantities are denoted with superscript m. For sake of simplicity, all controllers G represent proportional integral controllers of type $G = k_{\mathrm{P}} + \frac{1}{T_{\mathrm{i}} s}$. In Fig. 4(b), structural optimization parameters are of type binary and highlighted in blue. Moreover, controller parameters k_{P} and T_{i} of red edged boxes of controllers G are taken into account for optimization which are of type real.

Within this contribution, the focus is set on the P/Q control mode, see G_{P} and G_{Q} for active and reactive power, respectively. The ac currents are controlled by G_{ac} in synchronous reference frame ($dq-$frame). The related transformation angle is provided by a phase locked loop (PLL) which is designed according to [30]. Inner as well as dc currents are controlled in stationary reference frame ($\alpha\beta$-frame) by means of G_{int} and G_{dc}, respectively.

As indicated in Fig. 4(b), several power controller implementations are taken into account in order to provide the reference values for the ac current controller i_{dq}^{ref}. Here, different methods are introduced to obtain i_{dq}^{ref} from the power controller outputs P^{ref} and Q^{ref}, for instance, see (7) and (8). Variable u_{d0} is the steady-state voltage value of u_{d} which corresponds to $\frac{\sqrt{2}}{\sqrt{3}} 400\,\mathrm{kV}$, as the PLL controls u_{q} to zero.

$$i_{\mathrm{d}}^{\mathrm{ref}} = \frac{2P^{\mathrm{ref}}}{3u_{\mathrm{d0}}}, \quad i_{\mathrm{q}}^{\mathrm{ref}} = -\frac{2Q^{\mathrm{ref}}}{3u_{\mathrm{d0}}}. \tag{7}$$

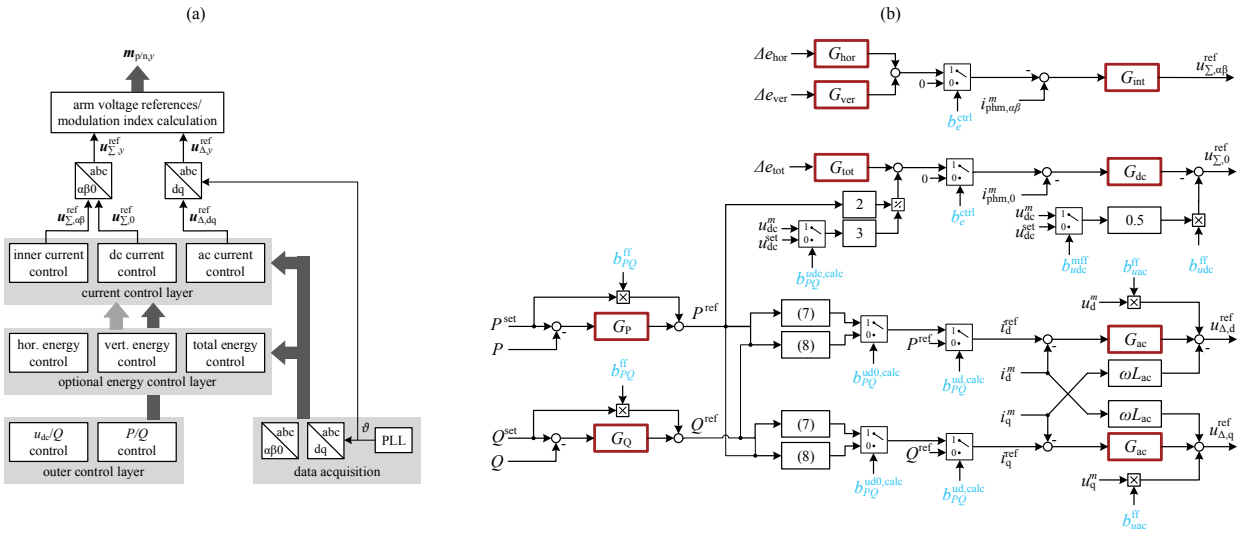

Fig. 4. (a) Hierarchical MMC control overview, (b) detailed overview of the P/Q-controlled station control highlighting all considered controller options.

$$i_{\mathrm{d}}^{\mathrm{ref}} = \frac{2(P^{\mathrm{ref}}u_{\mathrm{d}}^{m} + Q^{\mathrm{ref}}u_{\mathrm{q}}^{m})}{3((u_{\mathrm{d}}^{m})^2 + (u_{\mathrm{q}}^{m})^2)}, \quad i_{\mathrm{q}}^{\mathrm{ref}} = \frac{2(P^{\mathrm{ref}}u_{\mathrm{q}}^{m} - Q^{\mathrm{ref}}u_{\mathrm{d}}^{m})}{3((u_{\mathrm{d}}^{m})^2 + (u_{\mathrm{q}}^{m})^2)}. \tag{8}$$

As shown in [10], energy-based as well as non-energy-based MMC control approaches exist. For the former method, the energy within the MMC is explicitly balanced by means of several energy balancing controllers (depicted as optional energy controllers in Fig. 4), as, for instance, energy controllers G_{hor}, G_{ver} and G_{tot} according to [31] to compensate horizontal (Δe_{hor}) and vertical (Δe_{ver}) imbalances within the MMC as well as average energy deviations (Δe_{tot}) within the submodule capacitors. For the non-energy-based approach, circulating current suppression controllers are applied (see e.g. [9]) i.e. the reference values for phase module current controllers are set to zero. As indicated in Fig. 4(b), both options may be chosen by the optimization algorithms.

Limitation schemes are not applied within the controllers to improve the detection of an unwanted controller behaviour. Hence, a controller design leading to high overshoots must be penalized in the objective function during optimization.

III. Description of the Optimization Framework

The developed framework aims at finding optimal parameters of an arbitrary control structure. As backend, simulations in the software PSCAD are used. Fig. 5 shows the interaction between the optimizer (written in the software python) and the EMT simulations in PSCAD.

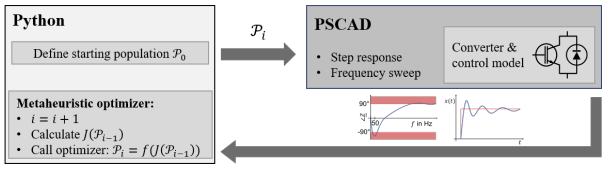

Fig. 5. Overview of proposed meta-heuristic optimization interface.

A. Mathematical Description

As in the general interaction depicted in Fig. 1, the optimizer creates individual solutions $L_j = \{x_{j,1}, \ldots, x_{j,n}\}$, for n parameters. The user-defined objective function of each solution returns a scalar value $J(L_j)$ after the simulation is finished. A population is defined as the combination of tuples of individual solutions and their respective evaluated objective function values $\mathcal{P}_i = \{(L_1, J(L_1)), \ldots, (L_p, J(L_p))\}$, where p is the population size and describes how many candidate solutions are generated in each iteration of the optimization procedure. Equation (9) shows the objective function in its general form.

$$J(L_j) = \alpha \cdot J_{\mathrm{freq}}(L_j) + \beta \cdot J_{\mathrm{time}}(L_j) + \lambda \cdot \varepsilon(L_j) \tag{9}$$

The objective function consists of three terms that can be weighted with the fixed parameters α, β and λ. The first term $J_{\mathrm{freq}}(L_j)$ penalizes deviations in the frequency domain from the (user-defined) desired properties. The second term $J_{\mathrm{time}}(L_j)$ applies the same concept to the time domain. A third term $\varepsilon(L_j)$ can be used to penalize additional undesired behaviour. Hence, the last term may be utilized to ensure that the chosen parameters are within a technical feasible range.

Based on the parameters and their respective objective function values of previous iterations $\{\mathcal{P}_0, \ldots, \mathcal{P}_i\}$, the optimizer calculates the new population of candidate solutions $\mathcal{P}_{i+1}$. Note that the individual optimizers all work differently, which means that the information from previous iterations might not or only partially be used. The generic interaction between optimizer and environment is summarized in Algorithm 1. It is important to note that meta-heuristic algorithms are highly dependent on the choice of starting population $\mathcal{P}_0$,

Algorithm 1 Generic meta-heuristic interaction algorithm

Input
α, β, λ: objective func params
`maxit, stopfit`: Stopping criterions
p: population size
n: number of parameters

Initialize
$\mathcal{P}_0 \leftarrow$ `GetInitialSolution`
`Env` $\leftarrow$ `Environment`(α, β, λ)
`Opt` $\leftarrow$ `Optimizer()`
`done` $\leftarrow$ `False`

procedure PARAMETER OPTIMIZATION($i = 0, 1, 2, \dots$)
  while not `done` **do**
    $\{J(L_1), ..., J(L_p)\} \leftarrow$`Env.sim`$(\{L_1, ..., L_p\}_i)$
    $\mathcal{P}_i \leftarrow \{(L_1, J(L_1)), ..., (L_p, J(L_p))\}_i$
    $\{L_1, \dots, L_p\}_{i+1} \leftarrow$ `Opt.step`$(\mathcal{P}_i)$
    $i \leftarrow i + 1$
    if $i >$`maxit` or min$(J) <$`stopfit` **then**
      `done` $\leftarrow$ `True`
    end if
  end while
end procedure

which makes mandatory multiple runs of each algorithm from different starting points.

B. Implementation of the Optimization Framework

The optimization procedure and interaction framework is written in `Python 3.7`, making use of the python implementation of the optimization algorithms of [32], where the algorithms in Tab. I are adopted to be applicable in the presented *ask-tell* interaction procedure. Communication with PSCAD is done using the PSCAD Automation Library. As indicated in Section II-C, a solution $\mathcal{L}_j$ comprises structural as well as parametric information for controllers. The population size is limited to a multiple of the number of available parallel licenses. In our case, the population size can be chosen to be $p = 16 \cdot z$ where $z \in \mathbb{N}^+$. This represents a compromise between investment costs and computation efficiency. The optimization is performed on a computer with an AMD Ryzen Threadripper 3990X 64 core processor with 128 GB of RAM.

IV. DEVELOPMENT OF THE OBJECTIVE FUNCTION

The choice of an adequate objective function is a sensitive part within the optimization framework as it directly influences the real-world behaviour of the optimized controller. When defining a desired behaviour of an MMC, several physical quantities are important and require monitoring.

A. Time domain objective function

For the objective function related to the time domain J_{time} several individual summands are taken into account. For sake of simplicity, a P/Q controlled terminal is optimized within this contribution. Therefore, all individual summands consider the response of several MMC quantities to a power setpoint step as soon as the HVdc link is started up. First, an active power step from 0 to nominal active power transfer occurs ($P^{\text{set}} = 1$ GW) and subsequently a reactive power step from 0 to nominal reactive power ($Q^{\text{set}} = 300$ Mvar).

After simulation of the current population $\mathcal{P}_i$, the following individual summands are calculated in each iteration in order to obtain J_{time}:

- $J_{P/Q}$: At the P/Q-controlled terminal the desired behaviour related to the ac side of the MMC is an injection of the set points for active and reactive power (P^{set} and Q^{set}). Here, a mean squared error (MSE) between power setpoints and measured power quantities is calculated for active and reactive power, respectively. Moreover, parameter sets leading to unrepresentable power values within the EMT simulation are strongly penalized by returning high objective values.
- J_{dc}: In order to optimize the dc side behaviour of the MMC, the dc voltage is taken as criteria. A MSE between the steady-state dc voltage value and the measured dc voltage is taken into account. Hence, critical overshoots resulting in possible insulation coordination issues are penalized. Again, unrepresentable dc voltage values are strongly penalized by returning high objective values.
- J_{energy}: The functionality of internal quantities as well as the energy balance within the submodule capacitors is evaluated by means of energy quantities. Here, the MSE of energy quantities related to Δe_{hor}, Δe_{ver} and Δe_{tot} are considered, penalizing unrepresentable energy quantities.

The objective function related to time domain aspects is summed up as shown in (10) by means of specified weights w for the introduced individual summands. These weights allow to weight the three different summands of the objective function by applying high weights to *important* characteristics and low weight values to *unimportant* characteristics.

$$J_{\text{time}} = w_{P/Q} J_{P/Q} + w_{\text{dc}} J_{\text{dc}} + w_{\text{energy}} J_{\text{energy}} \qquad (10)$$

As e.g. power and energy values are, due to their respective physical quantities, on other scales, the weights are utilized to create a balanced importance of all individual summands J_ξ for $\xi \in \{P/Q, \text{dc}, \text{energy}\}$. To obtain adequate weights, a known *good* start parameter set L_{good} is taken into account. This start parameter set is obtained by typical control design methods (symmetrical optimum method and modulus optimium method) for a fixed controller implementation as shown in [31]. Subsequently, a population $\mathcal{P}_{\text{good}}$ is created using a Gaussian distribution around the specified starting parameters with a standard deviation of $\sigma = 0.5 \cdot x_{0,i}$ and mean $\mu = x_{0,i}$ for all parameters i. Then, the objective values of individual summands are obtained by running a simulation set utilizing the parameter sets from the created *good* population $\mathcal{P}_{\text{good}}$.

The weights w_ξ are defined by $w_\xi = \kappa_\xi / \tilde{J}_\xi(\mathcal{P}_{\text{good}})$ with $\tilde{J}_\xi(\mathcal{P}_{\text{good}})$ representing the median objective value of the individual summands resulting from the *good* start population

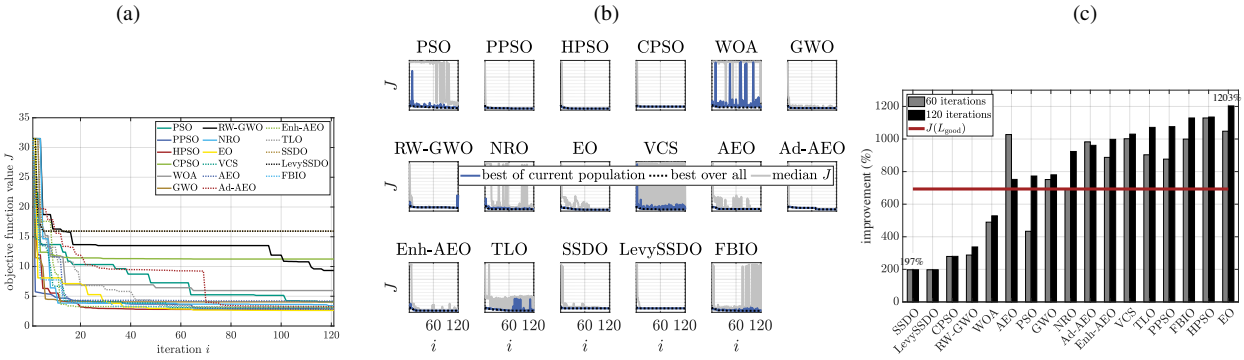

Fig. 6. Evaluation of optimization results: (a) Best solution for considered optimization algorithms over all iterations, (b) Performance of optimization algorithms over all iterations and (c) Relative improvement of all algorithms for an optimization with both 60 and 120 iterations.

and κ_ξ is a user-defined weight for the respective summand. Within this contribution κ_ξ is set to 1.

B. Frequency domain objective function

For the objective function related to the frequency domain J_{freq} we evaluate results obtained by frequency sweeps. Depending on the desired use case, the frequency sweep may be conducted at the ac or dc terminal of the converter station.

Regarding system stability, it is preferred that the overall system's impedance $\underline{Z}$ has enough phase margin to avoid unwanted interactions [1]. The overall system, consisting of the converter and the ac (or dc) grid, is therefore evaluated at its connection point of interest (ac or dc). It highly depends on the knowledge base (e.g. grid data) of the user whether the impedance of the overall system or solely the impedance of the converter shall be chosen by the user to be evaluated.

Again, several factors are regarded which can be weighted as defined by the user or can be obtained similarly as described for J_{time}. The latter option is chosen in this contribution. For the overall system evaluation, a passive system, i.e. $\Re\{\underline{Z}\} > 0$, results in good objective values, whereas non-passive regions are penalized in an individual summand J_{passive}. Moreover, the phase margin at resonant points is optimized. For the converter-standalone evaluation, a passive behaviour results in good objective values, too. However, depending on the operation point and due to existing system delays, non-passive regions still occur. Therefore, the converter's damping properties are optimized for these non-passive regions in J_{d}.

V. Evaluation of Optimization Framework

In order to evaluate the performance of the introduced optimization framework, we started different optimizations for the control design problem shown in Fig. 4(b). For sake of validation, we show time domain optimization results in a first stage, see Fig. 6. Hence, α in (9) is set to zero. Moreover, it is worth mentioning that an additional penalizing term related to technical constraints led to a reduced performance of optimization algorithms as minimums were found less easy.

We applied the same random seed $\mathcal{P}_0$ for all optimizations to obtain comparable results. Therefore, all algorithms start from the same objective function value ($J(\mathcal{P}_0) = 31.49$), see Fig. 6(a). The optimization was conducted over 120 iterations for all implemented algorithms. Obviously, all algorithms result in an improvement of control behaviour, i.e. J is minimized, whereas the final objective function value differs for the algorithms. Therefore, Fig. 6(b) evaluates the optimization behaviour of each algorithm. The black dotted lines correspond to the best J over the iterations as depicted in Fig. 6(a), whereas the blue lines show the best J of $\mathcal{P}_i$ at iteration i. Median J is the median over all objective values for the population $\mathcal{P}_i$. One can observe that algorithms like VCS and FBIO itensify *exploration*, whereas algorithms like PPSO, HPSO, CPSO, GWO, RW-GWO, EO, AD-AEO, Enh-AEO, SSDO and LevySSDO reach low medians at little iterations and thus obviously change from *exploration* to *exploitation* at a certain iteration. When evaluating WOA, the best solution shows very high J values even at high iterations.

To show consistency of the optimization algorithms across different executions, all algorithms were run several times at a fixed random seed. Fig. 6(c) shows the relative improvement of J for two optimizations of each algorithm with 60 and 120 iteration steps. Comparing these results, it is observed that the algorithms manage to give better results at a higher iteration number (except for AEO and Ad-AEO) and that the achieved objective values are in similar ranges for both executions. The best relative improvement of the objective function value was 1203% by means of EO and the smallest improvement rate (197%) was realized by SSDO and LevySSDO. Algorithms EO, HPSO and FBIO are the most successful optimizers for the formulated problem regardless of the number of iterations. Although, the algorithms SSDO, LevySSDO, CPSO and RW-GWO have also managed to improve, their best parameter sets are at rather lower improvement rates compared to the other algorithms. Focussing on time efficiency, algorithms VSC, NRO and EO show the highest relative improvement rate per hour, whereas SSDO, LevySSDO, CPSO RW-GWOA and WOA have the smallest due to their comparatively low improvement. This suggests that they are less suitable for the given problem. Generally, initialization of PSCAD as well as the EMT simulation itself represents the bottleneck of

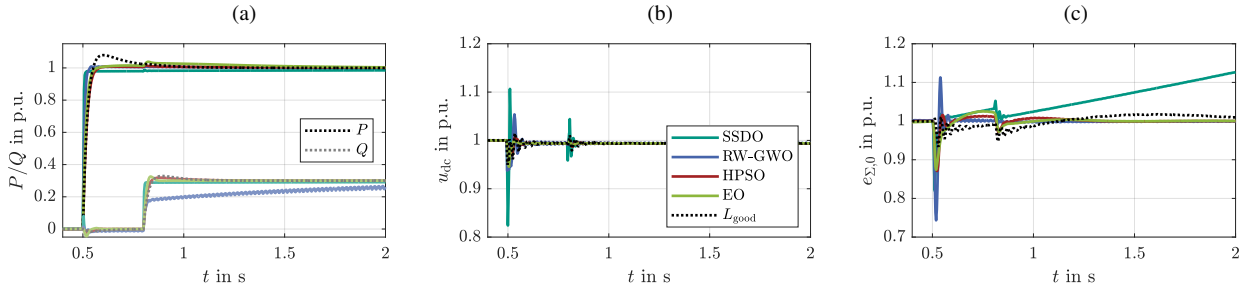

Fig. 7. Time domain results for L_{good} as well as achieved solutions by algorithms with small improvement rates (SSDO and RW-GWO) as well as highest improvement rates (HPSO and EO) for (a) active power, (b) reactive power as well as (c) total energy.

the optimization framework as it claims approximately 97 % of the computation time per iteration. For the time domain optimization, this results in an average computation time of 4 hours per 100 iterations for $p = 16$.

As indicated in Fig. 6(c), most of the algorithms achieve even better objective function values than the known *good* parameter set L_{good}. In order to evaluate the optimization results from the operational perspective, the time domain results related to the best solution L of each algorithm are analysed as well as shown in Fig. 7. For solutions L with $J(L) < J(L_{\text{good}})$ overshoots in dc voltage and ac power quantities are reduced at the cost of temporarily higher energy deviations (within technical bounds). This is realized by slightly higher ac current, higher dc current and reduced inner current dynamics compared to L_{good}. Moreover, power controller dynamics are reduced, i.e. the controller bandwidth is reduced. For solutions L with $J(L) > J(L_{\text{good}})$ power quantities achieve their set points comparatively slow or show high-frequency oscillations (RW-GWO) and high overshoots of dc voltage.

Results related to the controller implementation are stated in Tab. II. It is derived that most of the algorithms choose similar controller implementations for the given optimization problem. However, the *best* algorithms choose a detailed calculation of current references via (8), whereas the *worst* ones choose (7). Another structural difference represents the choice of b_{udc}^{mff}. Interestingly, only SSDO and LevySSDO choose the non-energy based control approach which indeed leads to an asymptotic instability of energy quantities (see Fig. 7(c)) and hence, high objective function values.

In the next step, we started different optimizations for the control design problem with respect to a combined frequency domain and time domain optimization. Within the frequency domain we optimized the ac side frequency response of the standalone model. From the optimization results (not shown here) one can derive that the solution with the best J_{freq} indeed improves passivity of the converter but at the cost of no stationary accuracy and high power oscillations. This situation is defined as Pareto optimality. Now, prioritizing stationary accuracy we chose higher weights for the time domain optimization i.e. $\alpha = 0.4$ and $\beta = 0.6$. The best solution was found by the HPSO algorithm.

TABLE II
CONTROLLER IMPLEMENTATION RESULTS OF FIVE *best* (BOLD) AND FIVE *worst* (IN BRACKETS) ALGORITHMS ACCORDING TO FIG. 6(C)

b_{PQ}^{ff}	$b_{PQ}^{\text{ud0,calc}}$	$b_{udc}^{\text{m,calc}}$	b_{e}^{ctrl}	b_{udc}^{mff}	b_{udc}^{ff}	b_{uac}^{ff}
1(1)	**0**(1)	**1**(1)	**1**(1)	**0**(1)	**1**(1)	**1**(1)

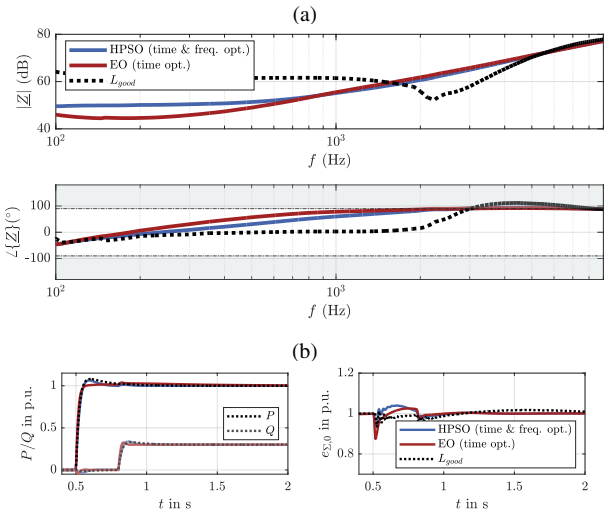

Fig. 8. (a) Frequency scan and (b) time domain responses for L_{good}, the best solution obtained by a combined frequency and time domain optimization (HPSO) as well as by the time domain optimization (EO).

In order to classify the combined optimization, Fig. 8 shows frequency domain as well as time domain responses for L_{good}, the best solution obtained by a combined frequency and time domain optimization by means of HPSO as well as the best solution obtained by the time domain optimization by means of EO. The non-passive region arising in the frequency range above 1 kHz is related to system delays and may be influenced by the controller design as well as implementation, as elaborated extensively in [2]. Generally, the best algorithms for both optimization settings find solutions that result in an improved damping behaviour within non-passive regions compared to L_{good}. When regarding the time domain results, indeed the solution obtained by the pure time domain optimization shows a better response for the given problem than the best solution found by the combined optimization.

VI. Conclusion

This contribution presents a toolbox that allows an interface between meta-heuristic optimization algorithms and an MMC model with an arbitrary control structure implementation. The toolbox enables to optimize the controller implementations with respect to time domain as well as frequency domain related issues.

It is shown that some algorithms are more suitable than others for the given problem and that obtained results are consistent across several optimization executions. Moreover, the time domain as well as frequency domain objective function was validated analysing the time domain and frequency domain results of the best solutions, respectively. On basis of the introduced objective functions, the algorithms manage to optimize the controller implementation and design in the desired manner related to an operational perspective. As most parts of the objective function are related to physical quantities at the MMC's ac or dc terminal, the introduced optimization framework is applicable for grey-boxed EMT models, too. In this case, physical domain knowledge is implemented in the optimization routine and allows to include prior knowledge of the optimized system. This can be done by including the signal flows and control blocks of PSCAD as additional information for the optimization.

It was shown briefly that the tool manages to optimize frequency domain related issues in a desired manner, too. Future research will evaluate the optimization performance with respect to certain weights for time and frequency domain optimization. Moreover, the optimization framework will be tested for several MMC control modes.

In this paper, only a selection of 18 meta-heuristic optimization algorithms from six categories is considered. Evaluating other algorithms might give valuable insights on how algorithms from different categories perform on the task at hand.

References

[1] CIGRE WG B4.67, "AC side harmonics and appropriate harmonic limits for VSC HVDC," *Technical Brochure 754*, 2019.

[2] C. Hirsching, S. Wenig, S. Beckler, M. Goertz, P. Praeger, M. Suriyah and T. Leibfried, "Passivity-Based Sensitivity Analysis of the Inner Current Controller in Grid-Following MMC-HVdc Applications - An Overview," in *IECON 2020 The 46th Annual Conference of the IEEE Industrial Electronics Society*, 2020.

[3] H. Wu, X. Wang, L. Kocewiak, L. Harnefors, "C Impedance Modeling of Modular Multilevel Converters and Two-Level Voltage-Source Converters: Similarities and Differences," in *19th Workshop on Control and Modeling for Power Electronics (COMPEL)*, 2018.

[4] C. Hirsching, A. Bisseling, S. Wenig, M. Goertz, C. Bischoff, M. Suriyah and T. Leibfried, "On the impact of controller implementations on passivity and damping properties in grid-following MMC-HVdc applications," in *IECON 2021 The 47th Annual Conference of the IEEE Industrial Electronics Society*, 2021.

[5] L. Bianchi, M. Dorigo, L. M. Gabardella and W. J. Gutjahr, "A survey on metaheuristics for stochastic combinatorial optimization," *Natural Computing*, vol. 8, no. 2 pp. 239–287, 2009.

[6] C. Blum and A. Roli, "Metaheuristics in Combinatorial Optimization: Overview and Conceptual Comparison," *ACM Comput. Surv.*, vol. 35, pp. 268–308, 2001.

[7] F. Glover and G. Kochenberger, *Handbook of Metaheuristics*, International Series in Operations Research & Management Science. Springer US, 2003. [Online]. Available: https://books.google.de/books?id=xmhoG772iuQC

[8] S. E. De León-Aldaco, H. Calleja, J. Aguayo Alquicira, "Metaheuristic Optimization Methods Applied to Power Converters: A Review," *IEEE Trans. on Pow. Electron.*, vol. 30, no. 12, pp. 6791–6803, 2015.

[9] CIGRE WG B4.57, "Guide for the Development of Models for HVDC Converters in a HVDC Grid," *Technical Brochure 604*, 2014.

[10] CIGRE WG B4.70, "Guide for electromagnetic transient studies involving VSC converters," *Technical Brochure 832*, 2021.

[11] R. Eberhart and J. Kennedy, "A new optimizer using particle swarm theory," in *MHS'95. Proceedings of the Sixth International Symposium on Micro Machine and Human Science*, IEEE, 1995, pp. 39–43.

[12] M. Ghasemin, E. Akbari, A. Rahimnejad, S. Razavi, S. Shavidel and L. Li, " Phasor particle swarm optimization: a simple and efficient variant of PSO," *Soft Computing*, vol. 23, pp. 9701–9718, 2019.

[13] M. Ghasemin,J. Aghaei, J. and M. Hadipour,"New self-organising hierarchical PSO with jumping time-varying acceleration coefficients," *Electronic Letters*, vol. 53, no. 20, pp. 1338–1388, 2017.

[14] B. Liu, L. Wang, Y.-H. Jin, F. Tang, and D.-X. Huang, "Improved particle swarm optimization combined with chaos," *Chaos, Solitons & Fractals*, vol. 25, no. 5, pp.1261–1271, 2005.

[15] S. Mirjalili and A. Lewis, "The whale optimization algorithm," *Advances in engineering software*, vol. 95, pp. 51–67, 2016.

[16] S. Mirjalili, S. Mirjalili and A. Lewis, "Grey Wolf Optimizer," *Advances in engineering software*, Vol. 69, pp. 46–61, 2014.

[17] S. Gupta and K. Deep, "A novel Random Walk Grey Wolf Optimizer," *Swarm and Evolutionary Computation*, vol. 44, pp. 101–12, 2019.

[18] J. Zhang and A. Sanderson, "JADE: Adaptive Differential Evolution With Optional External Archive," *Transactions on Evolutionary Computation*, vol. 13, no. 5, pp. 945–958, 2009.

[19] Z. Wei, C. Huang, X. Wang, T. Han and Y. Li, "Nuclear reaction optimization: A novel and powerful physics-based algorithm for global optimization," *IEEE Access*, vol. 7, pp. 66084–66109, 2019.

[20] A. Faramarzi, M. Heidarinejad, B. Stephens and S. Mirjalili, "Equilibrium optimizer: A novel optimization algorithm," *Knowledge-Based Systems*, vol. 191, 2020.

[21] M. Li, H. Zhao, Y. Weng and T. Han, "A novel nature-inspired algorithm for optimization: Virus colony search," *Advances in Engineering Software*, vol. 92, pp. 65–88, 2016.

[22] W. Zhao, L. Wang and Z. Zhang, "Artificial ecosystem-based optimization: a novel nature-inspired meta-heuristic algorithm," *Neural Computing and Applications*, vol. 13, 2019.

[23] V. T. Nguyen, "A collection of the state-of-the-art MEta-heuristics ALgorithms in PYthon: Mealpy," 2020. [Online]. Available: https://doi.org/10.5281/zenodo.3711948

[24] A. Eid, S. Kamel, A. Korashy, T. Khurshaid, "An Enhanced Artificial Ecosystem-Based Optimization for Optimal Allocation of Multiple Distributed Generations," *IEEE Access*, vol. 8, pp. 178493–178513, 2020.

[25] R. Rao and V. Patel, "An elitist teaching-learning-based optimization algorithm for solving complex constrained optimization problems," *Int. Journal of Ind. Eng. Comput.*, vol. 3, pp. 535–560, 2012.

[26] A. Tharwat and T. Gabel, "Parameters optimization of support vector machines for imbalanced data using social ski driver algorithm," *Neural Computing and Applications*, vol. 11, 2020.

[27] J. Chou and N. Nguyen, "FBI inspired meta-optimization," *Applied Soft Computing*, vol. 93, 2020.

[28] D. Wolpert and W. Macready, "No free lunch theorems for optimization," *IEEE Trans. Evol. Comput.*, vol. 1, no. 1, pp. 67–82, 1997.

[29] K. Sharifabadi, L. Harnefors, H.-P. Nee, S. Norrga and R. Teodorescu, *Design, Control and Application of Modular Multilevel Converters for HVDC Transmission Systems*, Wiley-IEEE Press, 2016.

[30] S. Golestan, M. Monfared, F. D. Freijedo, "Design-Oriented Study of Advanced Synchronous Reference Frame Phase-Locked Loops", *IEEE Trans. on Pow. Electron.*, vol. 28, no. 2, pp. 765–778, 2012.

[31] S. Wenig, "Potential of Bipolar Full-Bridge MMC-HVdc Transmission for Link and Overlay Grid Application," Doctoral thesis, Karlsruhe Institute of Technology (KIT), Karlsruhe, Germany, 2019.

[32] T. Nguyen, N. Tran, B. Nguyen, and G. Nguyen, "A resource usage prediction system using functional-link and genetic algorithm neural network for multivariate cloud metrics," in *2018 IEEE 11th conference on service-oriented computing and applications (SOCA)*, IEEE, 2018.