Practical Parameter Tuning of RF Jammers by Optimization of Expensive Black-box Functions

Fabian Schmied, Jan Lewandowsky
Communication Systems
Fraunhofer FKIE
53343 Wachtberg, Germany
{firstname.lastname}@fkie.fraunhofer.de

Jörg Schöbel
Institute of High Frequency Technology
Technical University of Braunschweig
38106 Braunschweig, Germany
j.schoebel@tu-bs.de

***Abstract*—In the areas of drone countermeasures, protection of critical infrastructure, and detention facility security, jamming of radio frequency communication systems is seeing increasing importance. Selecting optimized radio signals for jamming a specific target is associated with considerable efforts, as high jamming efficiency is required. This process can be performed experimentally by continuous combination of different tuning parameters describing the corresponding jamming signal. The signal is emitted and assessed by its jamming impact on the radio link. With the amount of different parameter combinations increasing, the number of required measurements grows exponentially and so does the time investment. To approach this task, we model the search for an optimized jamming signal as an expensive black-box optimization problem. As the number of measurements is required to be minimized while satisfying real-world constraints arising from the practical context, feasible algorithms have to be found in order to solve this optimization problem. For this, we present a multidimensional iterative grid search algorithm with superior features compared to manual Human-in-the-Loop approaches in the fields of unpredictable measurements and premature termination.**

This paper was originally presented at the NATO Science and Technology Organization Symposium (ICMCIS) organized by the Information Systems Technology (IST) Panel, IST-200 RSY – the ICMCIS, held in Skopje, North Macedonia, 16-17 May 2023.

***Index Terms*—RF Signals, Jamming, Algorithmic efficiency, Optimization methods, Search problems, Iterative algorithms, Hardware-in-the-loop, Benchmark**

I. Introduction

The use of radio frequency communication jammers is becoming increasingly important in the areas of protecting critical infrastructure and detention facilities, as well as drone countermeasures in the vicinity of airports, major events and VIPs. This ensures that unintended or even hostile communication is not correctly transmitted. At the same time, friendly communication must be affected as little as possible requiring precise and effective jamming deployments. Especially in mobile applications, also efficiency becomes a major point of interest, due to limitations in supplied power.

In order to successfully interfere with a specific target, an optimized jamming signal has to be emitted by the jamming device. The jamming signal is set by tuning various device parameters like pulse-form, sweep time, etc. Finding suitable parameters describing the signal to optimally jam a certain target is a hard task, as the amount of different parameter combinations can be prohibitively high. The prediction of proper parameter sets using analytical approaches can hardly be applied, as the internals of target architectures are usually unknown and differ greatly from case to case. Experimental setups have to be employed instead.

The experimental approach is a time consuming process, where the emitted jamming signal is evaluated by its jamming impact on the real-world hardware target. Different combinations of the tuning parameters have to be evaluated successively. Associated with the transmission time of the target and the measurement time of the lab devices, individual measurements can take up to multiple seconds to perform. With increasing numbers of parameters to evaluate, the total time investment grows exponentially and becomes unfeasible. This results in the need for a smart and time saving algorithm to automatically determine upcoming jamming parameters taken into account for evaluation.

In order to approach this task, we model the search for an optimized jamming signal as mathematical optimization problem including the experimental setup as a black-box function. Practical problems associated with the setup like unpredictable measurements and premature termination extend the task. Due to the aforementioned restrictions, common optimization techniques are ineffective or cannot be applied.

In this paper, we present a multidimensional iterative grid search algorithm denoted as *MIGS* to address these problems including the practical context. The *MIGS* algorithm provides a smart search procedure by mimicking human search behaviour. It also reflects the current state of research in the field of optimization-related search for a set of signal parameters to be used by a jammer, which is called jamming signal search. Finally, the *MIGS* algorithm is proposed as benchmark for further research.

We summarize our contribution in this paper as follows:

- We model the parameter tuning of RF jammers as mathematical optimization problem and describe its specific properties.
- We present the *MIGS* algorithm as ad-hoc solution for the stated optimization problem with respect to the practical

context.

- We discuss the features of the *MIGS* algorithm and propose it as benchmark for further research.

This article is structured as follows. In Section II we provide an overview of related work in the fields of optimization-related jamming signal search. In addition, we introduce more general approaches to optimization problems from other areas of research with similarities to the *MIGS* algorithm. In Section III we explain the experimental setup within the problem statement and deduce a mathematical model to form a black-box optimization problem. The *MIGS* algorithm is introduced in Section IV and experimental results are provided. Followed by a discussion in Section V, we evaluate advantages and disadvantages of the proposed algorithm. Finally, Section VI concludes our paper.

II. Related Work

With this paper, we address multiple problem domains originating from different branches of research. For that, this section provides an overview of the current state of research and related work in the fields of optimization-related jamming signal search, black-box optimization, *MIGS*-related algorithms and benchmark topics. We show similarities and compare to different approaches.

A. Optimization-related Jamming Signal Search

The optimization of radio frequency communication jammers has not been a major point of interest in previous publications. Modelling of jamming signals in the context of mathematical optimization is underrepresented. In most cases, tuning by hand and less sophisticated, easy to implement but also inefficient approaches are used as ad-hoc solutions, which can be deduced from [1]–[4].

Moving to a wider scope, the modelling of a jamming application in wireless networks as optimization search problem can be found in [5], where *genetic algorithms* are proposed as solvers. The authors model the jamming signal with multiple parameters applying an on-off duty cycle technique. The dimensionality of the search space is discussed in the context of time complexity. Genetic algorithms are compared with the benchmark algorithm that which is based on *iterative improvement*.

Advanced approaches using artifical intelligence and adaptive techniques [6] are more frequently published in the fields of radar jamming. However, these research results cannot directly be transfered to the stated domain of communication jamming.

B. Solvers for Black-box Optimization

A large number of algorithms from simple *random search* (RS), over various heuristics such as *genetic algorithms* (GA), *sequential model-based methods* (SMBO), *Bayesian optimization* (BO) up to complex hybrid methods [7] have been presented and compared with each other [8]–[11] in order to address the problems of *black-box optimization* (BBO), *hyperparameter optimization* and other applications. Even in terms of *expensive* optimization [12], more sophisticated solutions like *sequential surrogate-based optimization* (SSBO) algorithms have been reported in literature. Their suitability for the stated task has not been investigated yet.

C. MIGS-related Research

In contrast to the deterministic nature of *MIGS*, *Latin Hypercube sampling* (LHS) has been presented as statistical method to sample from a multidimensional hypercube [13]. Similarities to the *cutting-plane* method [14] and *branch-and-bound* methods [15] can be observed with *MIGS* concerning the iteration process and its recursive implementation. The non-exhaustive process utilizing *local* and *global search* methods has been described in [16], where a multidimensional hypercube is iteratively divided into differently sized subrectangles. In contrast to *MIGS*, these algorithms only sample at centerpoints of subrectangles and evaluate information gained from already sampled points. In terms of cheap *response surfaces* in grid based approaches, *sparse grids* have been established as superior over full grids due to less required evaluations [17] like being used in *finite element* methods.

D. Benchmark Comparison

There are common guidelines for the comparison of optimization algorithms as stated in [18]. Several test suites have been presented to automatize benchmarking of different types of optimization algorithms. The *COCO* platform [19] is suitable for continuous optimizers whereas the platform presented in [20] works on discrete optimization heuristics. Their benefit for automatized comparison of algorithms in the domain of slow evaluation speed has to be considered carefully as the conduction of multiple sets of measurements is prohibitively time consuming as reflected in [21]. In the fields of jammer research, [5] suggests at least 30 runs per algorithm per hyperparameter set to achieve statistical significance. This order of magnitude is to be adopted for our further investigations and to be examined with regard to statistical significance.

III. Modelling the Black-box Optimization Problem

In this section, we provide an overview of the problem statement and its characteristics including the experimental setup. Based on this, we deduce a mathematical model describing an optimization problem by embedding the setup as black-box function. The properties of the optimization problem are examined with respect to the practical context.

A. Problem Setup

RF communication jamming devices require tuned settings prior to deployment. One of these settings is the mode of operation. It is primarily distinguished between *active* and *reactive*. A reactive jammer may, in principle, conduct an in-depth signal analysis [1] after detection in order to deduce an optimized jamming signal for the particular situation. In practice, reactive jammers will likely benefit from predefined, optimized jamming signals in the same way as active jammers.

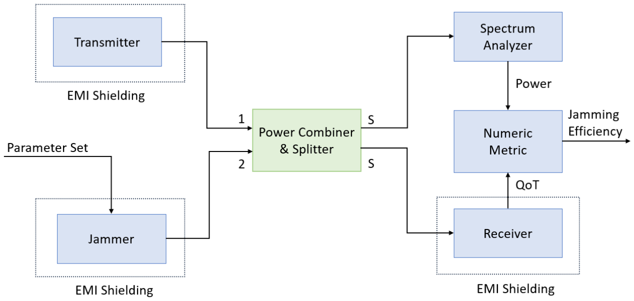


Fig. 1: Experimental evaluation of a jamming signal

A certain adaptivity may even be introduced in a reactive system by choosing a predefined jamming signal based on some features of the detected signal. This approach would particularly apply to mobile communications, where detection would rather occur in the uplink, but jamming would be applied to the downlink. As a result of the above, this paper is focused on the determination of optimized jamming signals predefined for certain communication systems, so that these signals may be used in active and reactive jammers.

Another important setting of a jamming device is the jamming signal. Each jamming signal has a certain impact on a specific target. This jamming signal is defined by type and different adjustable parameters. Among the most frequently used jamming signal types are *continuous wave* (CW), *additive white gaussian noise* (AWGN), and *sawtooth or triangular sweep*. These signals are defined by different numbers of parameters. For example, CWs are characterized by center frequency, AWGN by center frequency and bandwidth and sweeps by center frequency, bandwidth and sweep time. More complex sweeps can be created by adding more parameters [4].

The jamming device and its emitted signal is not only required to successfully jam the considered target, but also to do so while satisfying various constraints arising with the real-world use case. This includes power efficiency in environments with limited power supply, band occupancy to allow neighouring communication and avoidance of jamming friendly transmissions in shared channels. Different signals vary greatly in jamming impact and power efficiency, resulting in the task of finding the optimal parameter set respectively the optimal jamming signal to interfere with a specific target for the particular scenario.

In some cases, when a priori knowledge of the target's internal system is provided, the jamming impact can be simulated to a certain degree of accuracy [2]. Due to the variety in the architectures of the targets to be examined, we cannot assume that a priori knowledge is available for all targets, especially for targets using proprietary radio communication techniques. There are individual cases known, where proprietary protocols have been reverse engineered in order to simulate the behaviour of real hardware, as done for the *chirp spread spectrum* (CSS) based *LoRa* in [22].

With the restriction of missing a priori knowledge, we exclude the analytical approach of predicting or simulating a jamming signal's impact for this paper. This allows us to explore a more generic strategy addressing a wider range of different targets. For that, we approach the task of finding an optimized jamming signal by using an experimental setup. Each jamming signal is evaluated individually regarding its jamming impact on the specific target as shown in Fig. 1. For that to happen, the jamming signal is created as given by the parameter set and is emitted by the jamming device to interfere with the transmitted signal. The corresponding receiver results are commonly measured by data loss rates like *packet error rate* (PER) and *bit error rate* (BER) or by the number of disconnections [23], generally denoted as *quality of transmission* (QoT). Power measurements are typically stated as logarithmic signal-to-jamming ratio $\text{SJR} = 10 \cdot \log_{10}(P_S/P_J)$. Power measurements and quality of transmission are used to determine the jamming efficiency as a numeric metric to describe the benefit of the examined signal with respect to the considered case of use.

The experimental setup provides a *Hardware-in-the-Loop* architecture to work with different types of targets. Parameter sets are generated as combinations of tuning values within a predefined range to describe a corresponding jamming signal. The consecutive process of subsequently evaluating individual parameter sets is also known as *sequential sampling*. Due to the transmission times of real-world targets and measuring times of the lab devices, a single evaluation step can take several seconds. In company with involved *not strictly deterministic* and *noisy* behaviour of applied hardware, repetition of the same measurement is required to achieve statistical significance. Some communication systems may even use adaptive methods of transmission, which may result in the avoidance of regions (e.g. frequency ranges) where jamming efficiency is high. As a consequence, the number of measurements to achieve stable and statistical relevant results in such cases will be even larger.

The number of possible parameter sets to be evaluated grows exponentially with the increase of the number of tuning parameters and the size of predefined value ranges. Due to the aforenamed properties, full measurement cycles can take up to days, weeks, and more, hence be infeasible when using systematic and exhaustive search methods. Therefore, a smart selection of sampling points is required to reduce the number of evaluations significantly, while maintaining search effectiveness by using a specialized search algorithms.

In addition, more practical aspects must also be taken into account. Since no exact statements can be made in advance regarding the required measuring time and the parameter value range to be set, a technique for rough exploration of the search space is required in order to give an initial assessment. In the further course of the measuring, gradually refining the roughly explored search space is intended to increase the quality of the results. For this purpose it is important that measurements can be terminated prematurely to adjust search parameters without the loss of exploitability of already acquired data. This means, when terminating for example a linear search early, a specific part of the search space is explored extensively, but all other parts are not considered at all. Data aquired by a prematurely terminated linear search is not suitable for an initial assessment. In contrast, data evenly distributed over the entire search space is required.

B. Mathematical Model

The stated problem is approached by modelling the experimental setup as a function f. The response characteristics of each individual component like jamming signal generator, real-world target, lab devices and jamming efficiency calculation are combined into that single function. With respect to the previously mentioned restrictions, an analytical expression for f is not given, leaving f as black-box. The optimal parameter set $\boldsymbol{x}^*$ describing a jamming signal for a specific target set by f can be found by optimizing f forming a black-box optimization problem as follows.

Let $\mathcal{J}_\theta$ denote an arbitrary jamming signal of type θ in the time domain as $\mathcal{J}_\theta(\boldsymbol{x}, t)$. For notational convenience, we drop t in the further description, i.e.,

$$\mathcal{J}_\theta(\boldsymbol{x}) := f_\theta(\boldsymbol{x}) = f_\theta(x_1, ..., x_n) \tag{1}$$

where $\boldsymbol{x} = (x_1, ..., x_n)$ denotes the parameter set respectively the search parameter vector with search parameters $x_i \in [b_i^-, b_i^+]$ within bounded intervals, where b_i^- is the lower and b_i^+ the upper bound of the corresponding search dimension indexed by $i \in \{q \in \mathbb{N} \mid 1 \leq q \leq n\}$. The matrix $\boldsymbol{b}$ of all bounds with

$$\boldsymbol{b} = \begin{bmatrix} b_1^- & b_1^+ \\ \vdots & \vdots \\ b_n^- & b_n^+ \end{bmatrix} \tag{2}$$

describes the value ranges for all combinations of adjustable parameters. It defines the search space or choice set X as feasible set $X \subseteq \mathbb{R}^n$ with

$$X = \{\boldsymbol{x} \in \mathbb{R}^n \mid \forall i : b_i^- \leq x_i \leq b_i^+\} \tag{3}$$

where n denotes the dimensionality $n \in \mathbb{N}$ of the search space X. An element $\boldsymbol{x} \in X$ is called feasible or candidate solution.

Let Q_τ denote the jamming efficiency of a signal $\mathcal{J}_\theta$ interferring with target τ as defined by

$$Q_\tau(\boldsymbol{x}) := f_\tau(\mathcal{J}_\theta(\boldsymbol{x})) = f_\tau(f_\theta(\boldsymbol{x})) \tag{4}$$

Referring to the black-box function f, both functions f_θ and f_τ are modelled into f resulting in the representation of the initially presented jamming signal search problem as function $f : X \rightarrow Y$, which assigns the search parameter vector $\boldsymbol{x}$ to the corresponding jamming efficiency Q with dropped indices θ and τ by

$$Q(\boldsymbol{x}) := y = f(\boldsymbol{x}) = f(x_1, ..., x_n) \tag{5}$$

where evaluation value $y \in Y$ denotes a scalar value to define an objective function f for a given sample point $\boldsymbol{x}$. The set of all feasible objective values y is denoted as evaluation space $Y \subseteq \mathbb{R}$.

Let $\boldsymbol{x}^*$ be the *optimal* search parameter vector denoting an optimization problem with

$$\boldsymbol{x}^* := \underset{\boldsymbol{x} \in X}{\operatorname{argmax}}\, f(\boldsymbol{x}), \text{ then } Q^* = f(\boldsymbol{x}^*) \tag{6}$$

defines the *optimal* jamming efficiency Q^*, where f requires maximization to achieve the global optimimum, such that $f(\boldsymbol{x}^*) \geq f(\boldsymbol{x}) \mid \forall x$. Depending on the particular implementation of the jamming efficiency metric Q, modelling a minimization problem is valid too.

Referring to the different types of jamming signals given in Section III-A, an example sawtooth sweep signal is defined by fixed center frequency f_c equaling the targets carrier frequency, bandwidth B and sweeptime T as follows.

$$\begin{aligned} \theta &= \text{Sweep} \\ n &= 2 \\ \boldsymbol{x} &= [B, T] \\ \boldsymbol{b} &= \begin{bmatrix} 100\,\text{kHz} & 300\,\text{kHz} \\ 20\,\text{ms} & 250\,\text{ms} \end{bmatrix} \end{aligned} \tag{7}$$

When characterizing the properties of the modelled black-box optimization problem, there is no information on continuity and differentiability of f. Therefore, commonly used solvers utilizing gradient-based techniques cannot be applied. In addition, f must be assumed to be *multimodal* containing several local optima despite the sought global optimum. With evaluation value y being a scalar, f is denoted as real valued single objective function with *multidimensional* argument $\boldsymbol{x}$ and may be strongly non-linear. The black-box definition also causes $\boldsymbol{x}$ not being reducible, leading to exponentially growing search spaces X with the increase of dimensionality n, denoted as *curse of dimensionality* [17].

The experimental evaluation process causes measurements to be *noisy* and *not* to be *strictly deterministic*. The associated time consumption for individual evaluations is multiple magnitudes higher than the computational time of the search algorithm per evaluation characterizing the problem as *expensive*. This results in many solvers not being applicable.

Especially in large search spaces X, a very small *support* of f with

$$\operatorname{supp}(f) = \{\boldsymbol{x} \in X : f(\boldsymbol{x}) \neq 0\} \tag{8}$$

can be experimentally observed, showing a poor progagation among objective values with adequate jamming impact. This can be explained with a strong target dependence [3] of the jamming signal $\mathcal{J}$, resulting in a hard task to find any configuration $\boldsymbol{x}$ in X with desirable jamming efficiency Q at all.

As an example, a sawtooth sweep with a sweep time too long will not affect a fast frequency-hopping spread spectrum (FHSS) transmission, which is often used in off-the-shelf drone remote controls. A packet will only be lost if its current carrier frequency and the momentary sweep frequency coincide within the instantaneous bandwidth. With a suitable error correction, a certain packet loss may be fully recovered. Generally, error correction results in a nonlinear behavior of

TABLE I: Notations for *MIGS* algorithm

Notation	Description
$\boldsymbol{x}$	search parameter vector
x_i	i-th search parameter
X	search space
X'	transformed unit search space
n	number of search dimensions $[1..\infty]$
i	index of search dimension $[1..\infty]$
b_i^-, b_i^+	lower, upper bound of search dimension i
y	evaluation value
Y	evaluation space
k	iteration step $[0..\infty]$
j	node index $[0..\infty]$
$d_{k,j}$	node in iteration k of node index j
h_k	node spacing in iteration k
D_k	set of nodes in iteration k
$D_{k,i}$	set of nodes along coordinate axis i
$\widetilde{D}_k$	set of all node combinations in iteration k
$\widetilde{D}'_k$	set of all node combinations to be evaluated

the quality of transmission (QoT) versus signal-to-noise ratio (SNR), or signal-to-jamming ratio (SJR), respectively.

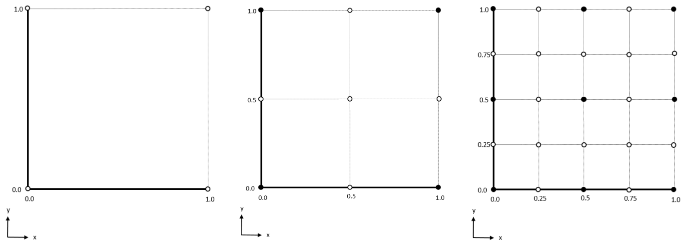
The modelled optimization does not have any constraints other than *bound constraints* $[b_i^-, b_i^+]$. They are given by the real-world constraints of used hardware. Tightening bounds by utilizing minor *a priori knowledge* would increase search efficiency and reduce time complexity markedly, if given. Bound knowledge on the evaluation space Y is typically not given and not strictly required for the optimization. The spaces X and Y are indicated as being *continuous* for generalization purpose. Some real-world hardware might only accept discrete values to set up jammers and measurement equipment, which is not reflected in the stated model.

IV. The *MIGS* Algorithm

Due to the previously described problems and constraints including the practical background, the selection of suitable search algorithms is severely limited. As with the lack of appropriate solvers for the stated setup, we present the *multi-dimensional iterative grid search* algorithm denoted as *MIGS* in order to provide the requested features. As a modified version of a deterministic *brute-force* algorithm, this method exhaustively evaluates the entire search space X by using a grid structure to identify sample points $\boldsymbol{x}$. The grid is gradually refined over all search dimensions i. *MIGS* aims to mimic intuitive human search behaviour and reflects the current state of research in the jamming context, as no advanced research on more sophisticated algorithms has been presented yet. The stated implementation of *MIGS* expects the optimization problem f with search space bounds $[b_i^-, b_i^+]$ and a termination criterion like *maximum number of iterations*.

A. Working Principle

The *MIGS* algorithm defines f on the unit hypercube $[0,1]^n$, where $y = f(\boldsymbol{x})$ can be evaluated for all $\boldsymbol{x} \in [0,1]^n$ requiring transformation $X' := X \to [0,1]^n$ of the original search domain. The corresponding normalized lower and upper bounds $b_i^- = 0.0$ and $b_i^+ = 1.0$ for each dimension i define



(a) Iteration $k = 0$ (b) Iteration $k = 1$ (c) Iteration $k = 2$

Fig. 2: *MIGS* with 2-dimensional search problem

the search space X' as hypercube in a Cartesian coordinate system, where each coordinate axis i is denoted as search dimension i.

For an n-dimensional grid, each unit bound coordinate axis i is subdivided into equidistant points denoted as *nodes* $d_{k,j}$ with node index j in iteration step k. The set $D_k = \{d_{k,j} \mid j = 0, ..., 2^k\}$ with nodes $d_{k,j} = jh_k$ and node spacing $h_k = 2^{-k}$ contains all nodes $d_{k,j}$ along an coordinate axis i in iteration step k. For n-dimensional grids, the Cartesian product of all sets $D_{k,i}$ along all coordinate axes i contains all possible combinations of nodes as of

$$\widetilde{D}_k := \prod_{i=1}^{n} D_{k,i} := D_{k,1} \times ... \times D_{k,n} \tag{9}$$

The search vector $\mathbf{x} \in \widetilde{D}_k$ is denoted as *sample point*. The subset $\widetilde{D}'_k \subseteq \widetilde{D}_k$ with $\widetilde{D}'_k = \widetilde{D}_k \setminus \widetilde{D}_{k-1}$ exclusively contains all sample points, which have not been evaluated in previous iterations $k-1$. Per step of iteration k, the set $\widetilde{D}'_k = \{\mathbf{x}_1, ..., \mathbf{x}_m\}$ with

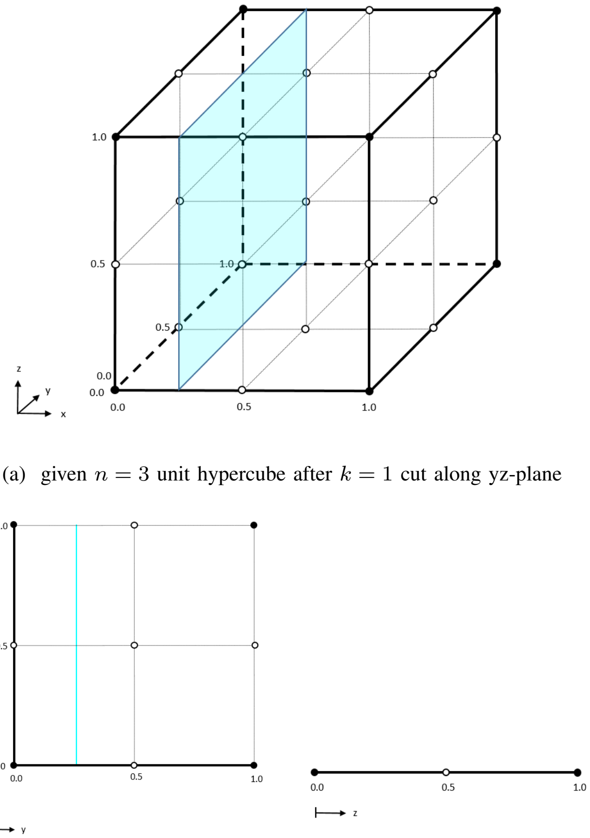
$$m = \begin{cases} 2^n, & \text{for } k = 0, \quad (10a) \\ (2^k+1)^n - (2^{k-1}+1)^n, & \text{for } k \geq 1, \quad (10b) \end{cases}$$

is evaluated as of $y_l = f(\mathbf{x}_l)$ to form a solution tuple $s = (y_l, \mathbf{x}_l)$ with $l = 1...m$.

B. Example for Determination of $\widetilde{D}'_k$ in $\mathbb{R}^2$

Given a 2-dimensional objective function $f : X' \subseteq \mathbb{R}^2 \to \mathbb{R}$ in iteration $k = 0$, the set $\widetilde{D}_0 = \widetilde{D}'_0$ contains 2-tuples of all combinations of the lower and upper bounds b_i^- and b_i^+ of the search space X' resulting in $\widetilde{D}'_0 = \{(0.0, 0.0), (0.0, 1.0), (1.0, 0.0), (1.0, 1.0)\}$ as shown in Fig. 2a. Unfilled circles on the 2-dimensional grid represent sample points $\mathbf{x}_l \in \widetilde{D}'_k$, which need to be evaluated in the current iteration k. Black filled circles represent already evaluated sample points $\mathbf{x}_l \in \widetilde{D}_k \setminus \widetilde{D}'_k$ from the previous iteration $k-1$, which need to be omitted from current and future evaluations.

With the next step of iteration $k = 1$ in Fig. 2b, the grid is refined by introducing a new search node $d_{k=1,j=1} = 0.5$ into $D_{k=1}$ in the middle of two nodes from the previous iteration $k-1$ with spacing h_k. Index j is updated with every iteration in ascending order when adding new nodes.



(a) given $n = 3$ unit hypercube after $k = 1$ cut along yz-plane

(b) resulting hyperplane in yz-plane cut again along z-axis

(c) resulting hyperplane is 1-dimensional along z-axis

Fig. 3: Recursive reduction of dimensionality

The set of all sample points yet to be evaluated results in set $\widetilde{D}'_1 = \{(0.0, 0.5), (0.5, 0.0), ..., (1.0, 0.5)\}$. This process continues in iteration $k = 2$ with the insertion of $d_{2,1} = 0.25$ and $d_{2,3} = 0.75$ resulting in set $\widetilde{D}'_2 = \{(0.0, 0.25), (0.0, 0.75), ..., (1.0, 0.75)\}$ as shown in Fig. 2c. This refinement continues until a stopping condition has been reached.

C. Implementation of MIGS

The implementation of *MIGS* concentrates on the set $\widetilde{D}'_k$ of samples points to be evaluated in an n-dimensional search space. Applying a recursion technique, the n-dimensional search space has to be broken down into multiple 1-dimensional search spaces. This is done by slicing the n-dimensional unit hypercube search space between every node $d_{k,j}$ and $d_{k,j+1}$ $\forall j \in \{q \in \mathbb{N}_0 \mid 0 \leq q \leq 2^k - 1\}$ into the corresponding $(2^k + 1)$ slices as $(n - 1)$-dimensional hyperplanes. For the exam ple of a 3-dimensional hypercube after iteration $k = 1$, Fig. 3a shows a cut along the yz-plane between the nodes $d_{k=1,j=0}$ and $d_{k=1,j=1}$ on the x-axis. The corresponding 2-dimensional hyperplane is shown in Fig. 3b represented as hyperplane in the yz-plane.

As the dimensionality of the resulting hyperplane in Fig. 3b is $n = 2$, the hyperplane is cut again in the next recursion level along the z-axis between the nodes $d_{k=1,j=0}$ and $d_{k=1,j=1}$ on the y-axis resulting in a 1-dimensional line in Fig. 3c. As the final recursion depth has been reached due to $n = 1$, all required nodes respectively sample points are evaluated as specified in $\widetilde{D}'_k$.

Generalizing the recursion technique, each of the resulting (2^k+1) hyperplanes after every cut can be formally considered a hypercube with $(n - 1)$ dimensions too. This way, the newly created hypercubes can be successively sliced until the resulting hyperplanes reach dimensionality $n = 1$. As a result of this iteration process, for all iterations $k \geq 1$, only sample points with odd node index j have to be evaluated on the final hyperplane with $n = 1$. This way it is ensured, that no sample point is evaluated twice.

D. Experimental Evaluation

Due to the complexity of the already simplified depicted experimental setup described in Section III-A and the difficulties in interpretation of measurement results connected with that, an approximate assessment of the feasibility of *MIGS* is conducted by using artificial optimization problems as models for the setup. This way, similarities in the structures of the problems are exploited to transfer the evaluation results for those artifical problems to the initially stated problem. This results in the advantage that artifical problems in terms of benchmark functions can be calculated considerably faster due to their nature as pure software functions. Due to prior research, the optimal solutions of benchmark functions are already known, which is important in order to assess the quality of the solutions found by the *MIGS* algorithm.

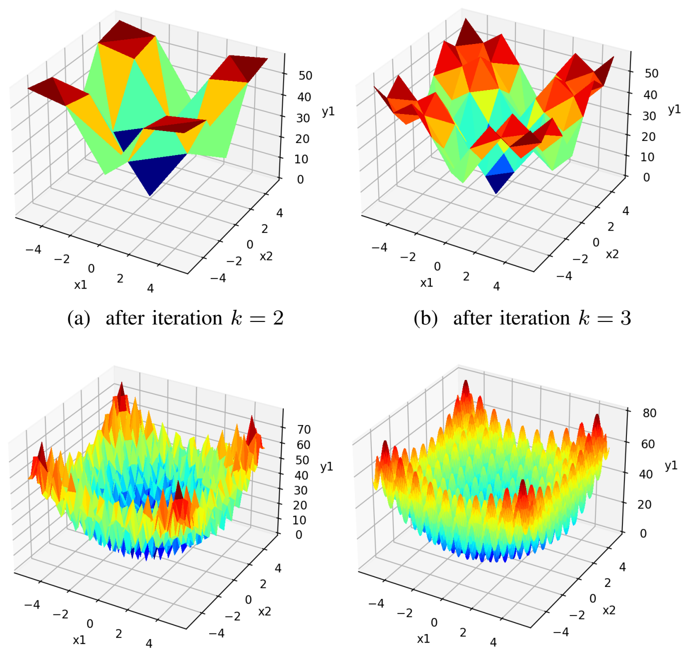
One of the most commonly used benchmark problems was proposed by Rastrigin and popularized as n-dimensional version by [24] and [25]. The function is defined as follows:

$$f_{Ras}(\boldsymbol{x}) = 10n + \sum_{i=1}^{n} x_i^2 - 10cos(2\pi x_i) \tag{11}$$

where $x_i \in [-5.12, 5.12]^n$ with global optimum at

$$\boldsymbol{x}^* = \arg\min_{\boldsymbol{x} \in X} f_{Ras}(\boldsymbol{x}) = \boldsymbol{0} \text{ and } f_{Ras}(\boldsymbol{x}^*) = 0 \tag{12}$$

When using *MIGS* on the 2-dimensional version of f_{Ras}, an evaluation space landscape can be drawn as shown in Fig. 4 for different numbers of iterations. The differences in search resolution can be observed with the number of iterations. The visual resolution is significantly worse with smaller numbers and thus less sample points. The global minimum of f_{Ras} is located at $\boldsymbol{x}^* = \boldsymbol{0}$, which is already found in the initial iteration $k = 0$. This coincidence happens by pure chance with the stated function and is caused by the deterministic nature *MIGS*.



(a) after iteration $k = 2$ (b) after iteration $k = 3$

(c) after iteration $k = 5$ (d) after iteration $k = 7$

Fig. 4: Visualization of *MIGS* algorithm with 2-dimensional Rastrigin optimization problem

TABLE II: Evaluation results of *MIGS* algorithm with 2-dimensional artifical optimization problems

		Rosenbrock (2D)			Styblinski-Tang (2D)		
k	evls.	x1	x2	y	x1	x2	y
1	9	0.00	0.00	1.00	0.00	0.00	0.00
2	25	0.00	0.00	1.00	-2.50	-2.50	-73.44
3	81	0.00	0.00	1.00	-2.50	-2.50	-73.44
4	289	0.00	0.00	1.00	-3.13	-3.13	-76.51
5	1089	1.25	1.56	0.63	-2.81	-2.81	-78.05
6	4225	1.25	1.56	0.63	-2.97	-2.97	-78.18
7	16641	1.02	1.02	0.25	-2.89	-2.89	-78.33
$x_i \in$		$[-5.00, 10.00]^2$			$[-5.00, 5.00]^2$		
$f(\boldsymbol{x}^*)$		0.00			-78.33		
$\boldsymbol{x}^*$		$(1.00, 1.00)$			$(-2.90, -2.90)$		

In contrast to that, other test functions with optima far off the grid provide different results. For example, the 2-dimensional versions of the Rosenbrock function defined by

$$f_{Ros}(\boldsymbol{x}) = \sum_{i=1}^{n-1} [100(x_{i+1} - x_i^2)^2 + (x_i - 1)^2] \tag{13}$$

and the Styblinski-Tang function defined by

$$f_{Sty}(\boldsymbol{x}) = \frac{1}{2} \sum_{i=1}^{n} (x_i^4 - 16x_i^2 + 5x_i) \tag{14}$$

result in TABLE II. As can be seen, the number of sample points is exponentially growing. The amount of presented iterations reflects practical time limits as generating more samples becomes impracticable. With regard to this limitation, solutions for f_{Ros} are of moderate quality. Compared to this, f_{Sty} can be sufficiently solved after iteration $k = 7$.

TABLE III: Evaluation results of *MIGS* algorithm with 5-dimensional artifical optimization problem

		Styblinski-Tang (5D)					
k	evls.	x1	x2	x3	x4	x5	y
1	243	0.00	0.00	0.00	0.00	0.00	0.00
2	3125	-2.50	-2.50	-2.50	-2.50	-2.50	-183.60
3	59049	-2.50	-2.50	-2.50	-2.50	-2.50	-183.60
4	1419857	-3.13	-3.13	-3.13	-3.13	-3.13	-191.30
$x_i \in$		$[-5.00, 5.00]^5$					
$f(\boldsymbol{x}^*)$		-195.83					
$\boldsymbol{x}^*$		$(-2.90, -2.90, -2.90, -2.90, -2.90)$					

Referring to the initial problem of multidimensional parameter vectors describing a jamming signal, the aforementioned 2-dimensional benchmark functions can for example be imagined to describe the function of a sweep signal characterized by the 2 parameters *bandwidth* and *sweeptime*. As already stated, more complex signals with higher dimensionality are possible [4]. To illustrate that, a 5-dimensional version of Styblinski-Tang is evaluated with the *MIGS* algorithm. As shown in TABLE III, the amount of evaluations is significantly higher. After $k = 3$ iterations, the best solution found reaches 93.8% of the optimal solution's value. This result must be interpreted with respect to the relative definition of sufficiency, which dependends on the actual problem. Without the specific context of problem statement and use case, no statement can be made as to whether 93.8% is sufficient. It must also be noted that the number of evaluations connected with $k = 4$ is practically not realizible due to the connected time investment.

V. Discussion

The *MIGS* algorithm is presented as ad-hoc solution for the stated problem. Due to its nature as an exhaustive search, it must be classified as *slow* algorithm of *moderate search efficiency*. However, it solves practical problems in real-world applications when it comes to *early termination*, *unknown runtime budgets*, *unexpected crashes* and *missing measurement readings*. Noisy and not strictly deterministic measurements do not affect the algorithm, as future sample points do not rely on previous evaluation values. Due to the iterative procedure, premature termination is always possible within an iteration step without loss of diversification among evaluation values, hence, making the algorithm particularly appealing for time consuming measurement based blackbox optimization in the jamming context.

As already shown in Fig. 4, the iterative feature can be used for graphical illustration of search spaces. Equidistant sample points are optimal for 3-dimensional representation. These plots can be used to assess the problem's complexity to decide on changing parameter value ranges or switching to a more sophisticated search algorithm, which is yet to be researched. For future purpose, when it comes to switching to other search algorithms, *MIGS* can also be used as an initial check to validate, that jamming signals provided by all combinations of the *extrema* of the parameters's value ranges are practically realizable. This indicator can be used to deduce, but must not be fully relied on, the validity of other combinations within the value ranges to comply with the *unconstrained* attribute of the black-box model.

MIGS is suitable for *continuous* optimization, limiting its use with discrete domains. Discretizing grid node values can result in re-evaluation of already used sample points, which breaks a main feature of MIGS of not evaluating a point twice. The algorithm cannot operate in *infinite search spaces* and requires *domain bounds*, which is given in real-world applications as provided by the settings of lab devices. The non problem-specific nature of the algorithm pursues a generalized approach, which makes it *widely applicable*. However, mechanisms to select and explore interesting *search regions* are matters of further research. The convenience of ease to use comes at the expense of efficiency.

Solving the modelled optimization problem requires further research. For that, different algorithms have to be tested and compared to each other with respect to the *expensive* attribute of the task. In terms of algorithm performance comparison, *MIGS only* needs to be evaluated as function of the *number of iterations to conduct* respectively the total number of evaluations. In contrast to more complex algorithms, there are *no other hyperparameters* to be tuned. Hence, comparison to *MIGS* is *significantly easier*. It also serves the purpose of reflecting the current state of research in the specified field of jamming. Considering the previously mentioned arguments, proposing *MIGS* as benchmark in the stated application field appears to be reasonable.

VI. CONCLUSION

In this paper, we have shown how to describe abstract jamming signals by different adjustable parameters and how to model the search for feasible parameter sets as expensive black-box optimization problem. We have characterized this task as complex and hard, requiring special search algorithms to approach the difficulties originating from the practical background. The incomplete research in the field of jamming signal optimization has been reflected with the introduction of the *MIGS* algorithm as ad-hoc solution. The algorithm has been discussed and found to be eligible as benchmark to be compared to other algorithms.

The results of the investigation of *MIGS* on artificial optimization problems show that the particularly emphasized properties of MIGS, especially the early stopping feature, also appear to be useful. In a next step, the application with the measurement setup from Fig. 1 will also be examined in order to obtain benchmarking values for the evaluation of further search methods. Based on this, more sophisticated and intelligent algorithms will be investigated in the area of solving the optimization problem modelled in this paper.

DECLARATION OF RESTRICTIONS

The authors of this paper declare that there are no restrictions regarding the presentation, during the event nor on the publication of the paper in the Meeting Proceedings and IEEE Xplore. This paper is indicated as young scientist contribution.

REFERENCES

[1] D. Rauschen, D. Gläsel, H. P. Stuch, P. Zimmermann, and M. Antweiler, "Commercial of the shelf counter uav," in *2019 International Conference on Military Communications and Information Systems (ICMCIS)*, 2019, pp. 1–5.

[2] K. Pärlin, M. M. Alam, and Y. Le Moullec, "Jamming of uav remote control systems using software defined radio," in *2018 International Conference on Military Communications and Information Systems (ICMCIS)*, 2018, pp. 1–6.

[3] M. Mileusnic, B. Pavic, V. Marinkovic-Nedelicki, P. Petrovic, D. Mitić, and A. Lebl, "Analysis of jamming successfulness against rcied activation with the emphasis on sweep jamming," *FACTA UNIVERSITATIS Series Electronics and Energetics*, vol. 32, pp. 211–229, 06 2019.

[4] M. Mileusnić, P. Petrović, B. Pavić, V. Marinković-Nedelicki, J. Glišović, A. Lebl, and I. Marjanović, "The radio jammer against remote controlled improvised explosive devices," in *2017 25th Telecommunication Forum (TELFOR)*, 2017, pp. 1–4.

[5] Y. S. Kim, B. DeBruhl, and P. Tague, "Stochastic optimization of flow-jamming attacks in multichannel wireless networks," in *2013 IEEE International Conference on Communications (ICC)*, 2013, pp. 2165–2170.

[6] C. G. Shi, F. Wang, S. Salous, and J. J. Zhou, "Adaptive jamming waveform design for distributed multiple-radar architectures based on low probability of intercept," *Radio Science*, vol. 54, no. 1, pp. 72–90, 2019. [Online]. Available: https://agupubs.onlinelibrary.wiley.com/doi/abs/10.1029/2018RS006668

[7] C. Gong, X. Li, H. Su, J. Guo, and L. Gu, "Global optimization of expensive black-box models based on asynchronous hybrid-criterion with interval reduction," 2018. [Online]. Available: https://arxiv.org/abs/1811.12142

[8] P. Liashchynskyi and P. Liashchynskyi, "Grid search, random search, genetic algorithm: A big comparison for nas," 12 2019.

[9] S. Alarie, C. Audet, A. E. Gheribi, M. Kokkolaras, and Sebastien, "Two decades of blackbox optimization applications," *EURO Journal on Computational Optimization*, vol. 9, p. 100011, 2021. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S2192440621001386

[10] L. Rios and N. Sahinidis, "Derivative-free optimization: A review of algorithms and comparison of software implementations," *Journal of Global Optimization*, vol. 56, 11 2009.

[11] R. M. Lewis, V. Torczon, and M. W. Trosset, "Direct search methods: then and now," *Journal of Computational and Applied Mathematics*, vol. 124, no. 1, pp. 191–207, 2000, numerical Analysis 2000. Vol. IV: Optimization and Nonlinear Equations. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0377042700004234

[12] D. R. Jones, M. Schonlau, and W. J. Welch, "Efficient global optimization of expensive black-box functions," *Journal of Global Optimization*, vol. 13, no. 4, pp. 455–492, Dec 1998. [Online]. Available: https://doi.org/10.1023/A:1008306431147

[13] M. D. McKay, R. J. Beckman, and W. J. Conover, "A comparison of three methods for selecting values of input variables in the analysis of output from a computer code," *Technometrics*, vol. 21, no. 2, pp. 239–245, 1979. [Online]. Available: http://www.jstor.org/stable/1268522

[14] R. Gilmore and R. Gomory, "A linear programming approach to the cutting stock problem i," *Oper Res*, vol. 9, 01 1961.

[15] A. H. Land and A. G. Doig, "An automatic method of solving discrete programming problems," *Econometrica*, vol. 28, no. 3, pp. 497–520, 1960.

[16] D. Jones, C. Perttunen, and B. Stuckman, "Lipschitzian optimisation without the lipschitz constant," *Journal of Optimization Theory and Applications*, vol. 79, pp. 157–181, 01 1993.

[17] J. Valentin, "B-splines for sparse grids : algorithms and application to higher-dimensional optimization," 2019. [Online]. Available: http://elib.uni-stuttgart.de/handle/11682/10521

[18] T. Bartz-Beielstein, C. Doerr, J. Bossek, S. Chandrasekaran, T. Eftimov, A. Fischbach, P. Kerschke, M. López-Ibáñez, K. Malan, J. Moore, B. Naujoks, P. Orzechowski, V. Volz, M. Wagner, and T. Weise, "Benchmarking in optimization: Best practice and open issues," 07 2020.

[19] N. Hansen, A. Auger, O. Mersmann, T. Tusar, and D. Brockhoff, "Coco: A platform for comparing continuous optimizers in a black-box setting," *Optimization Methods and Software*, vol. 36, 03 2016.

[20] C. Doerr, F. Ye, N. Horesh, H. Wang, O. Shir, and T. Bäck, "Benchmarking discrete optimization heuristics with iohprofiler," 07 2019, pp. 1798–1806.

[21] Z. Hoffman and S. Huntsman, "Benchmarking an algorithm for expensive high-dimensional objectives on the bbob and bbob-largescale testbeds," in *GECCO Workshop on Black-Box Optimization Benchmarking (BBOB 2022)*, Boston, United States, Jul. 2022. [Online]. Available: https://hal.archives-ouvertes.fr/hal-03665291

[22] B. Al Homssi, K. Dakic, S. Maselli, H. Wolf, S. Kandeepan, and A. Al-Hourani, "Iot network design using open-source lora coverage emulator," *IEEE Access*, vol. 9, pp. 53 636–53 646, 2021.

[23] A. Lebl, M. Mileusnic, P. Petrovic, B. Pavic, and V. Marinkovic-Nedelicki, "Successful sweep jamming rate determination of mpsk modulated rcied activation message signals," vol. 3, pp. 37 – 44, 12 2019.

[24] F. Hoffmeister and T. Baeck, "Genetic algorithms and evolution strategies: Similarities and differences," in *Parallel Problem Solving from Nature*, H.-P. Schwefel and R. Maenner, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 1991, pp. 455–469.

[25] H. Miehlenbein, "Genetic algorithms and parallel processing," *Wuhan University Journal of Natural Sciences*, vol. 1, no. 3, pp. 630–639, Dec 1996. [Online]. Available: https://doi.org/10.1007/BF02900898