www.czasopisma.pan.pl PAN POLSKA AKADEMIA NAUK www.journals.pan.pl

jet INTL JOURNAL OF ELECTRONICS AND TELECOMMUNICATIONS, 2022, VOL. 68, NO. 4, PP. 695-708
Manuscript received October 8, 2022; revised November 2022. DOI: 10.24425/ijet.2022.141291

AH Method: a Novel Routine for Vicinity Examination of the Optimum Found with a Genetic Algorithm

Daniel Andrzej Piętak, Piotr Bilski, and Paweł Jan Napiorkowski

***Abstract*—The paper presents a novel heuristic procedure (further called the AH Method) to investigate function shape in the direct vicinity of the found optimum solution. The survey is conducted using only the space sampling collected during the optimization process with an evolutionary algorithm. For this purpose the finite model of point-set is considered. The statistical analysis of the sampling quality based upon the coverage of the points in question over the entire attraction region is exploited. The tolerance boundaries of the parameters are determined for the user-specified increase of the objective function value above the found minimum. The presented test-case data prove that the proposed approach is comparable to other optimum neighborhood examination algorithms. Also, the AH Method requires noticeably shorter computational time than its counterparts. This is achieved by a repeated, second use of points from optimization without additional objective function calls, as well as significant repository size reduction during preprocessing.**

***Keywords*—heuristics; evolutionary computations; genetic algorithms; uncertainty estimation; parameter study**

I. Introduction

EVOLUTIONARY computations are well-developed global optimization techniques. Many different approaches to this topic have been designed since the generic strategy of concurrent seeking the optimal solution was originally proposed by Holland [1]. Various techniques involve combinations of space exploration and exploitation of previously revealed attraction regions. The search process is performed by processing a set of candidate solutions, called a population.

During this operation a vast number of objective function values are calculated for different points in the search space. Assuming that the optimization procedure succeeded in finding an optimum, the sampling collected in the process contains not only the global optimum, but also some information about its vicinity. Such a repository may be the subject of further processing, which would result in a cost-free outcome, i.e. revealing the approximated shape of the optimum neighborhood, without the need for additional objective function value calculations.

While assembling the points into a repository of space sampling is a straightforward task for most optimization problems, extracting the required information can be challenging. One should be certain that: the points cover the whole area of interest and the sampling distribution is scattered uniformly enough to deliver exact information.

Knowing that the statistical distribution of points depends predominantly on the adopted optimization strategy and the shape of the search space, it is crucial to choose the set of points with the desired characteristic for the second use in the function shape examination. This helps to save a lot of computational time as no additional objective function value calculations are required during the following function shape investigation.

This paper is composed as follows. Following the introduction in Section I, a brief review of the related work is provided in Section II. Section III gives a description of the AH Method. Section IV defines the genetic algorithm under consideration and depicts a two-dimensional test-case based on the F7 function by Schwefel [2]. In Section V the results of the test-case are presented. Section VI concludes this paper. It outlines possible directions for further research and specifies possible areas where the AH Method may be applied.

II. Related Work

Although multiple approaches to the search space sampling have been applied (through various heuristic algorithms, such as simulated annealing, evolutionary processes, Particle Swarm Optimization, etc.), no research has been done on the second use of space sampling. If the generation of new solutions is difficult or computationally demanding, function shape may be investigated in the direct vicinity of the found solution. A large number of references exist associated with the operations used in particular steps of the presented procedure.

There are multiple sources regarding optimization approaches. Genetic Algorithms (GA) are a benchmark method, used as the standard tool [1, 3-5].

In case of the GA's population diversity there is an extensive review in [4] providing a set of equations describing population variance in a quasi-equilibrium state for an infinite population model. This methodology can be applied to the number of points higher in order of magnitude than the case considered in this paper. The test-case results in [6] show the approximation error below ±5% with the error median about -1%. This was achieved for an order of 10^7 points and only a one-dimensional Gaussian fitness function. This paper investigates multidimensional multimodal function shape and the number of points in our two-dimensional test-case is of the order of 10^3.

The arithmetic crossover operator, considered in [6], has a significant drawback of unauthorized self-convergence inclination to the center of a domain range [7]. This makes it

Daniel Andrzej Piętak and Piotr Bilski are with Institute of Radioelectronics and Multimedia Technology, Warsaw University of Technology, Poland (e-mail: piotr.bilski@pw.edu.pl).

Paweł Jan Napiórkowski is with Heavy Ion Laboratory, University of Warsaw, Poland (e-mail: pjn@slcj.uw.edu.pl).

useless if a location of the global optimum is previously unknown and may lie close to some of the search space boundaries, because otherwise it would disturb finding the minimum. Thus we decided to use a different approach and to base the tool for the sampling survey of a relatively small finite set of points on a sophisticated statistical and histogramic analysis (sampling analysis module) rather than on the Glivienko-Cantelli theorem which assumes an infinite model.

As the clustering procedure [8-13] we decided to implement the NBC algorithm [14], aiming to isolate from the repository the points located in the attraction region of the optimum. In the future we consider applying Self-Organizing Maps [15] to overcome the (sometimes challenging) need to specify the values of three preprocessing module arguments.

Uncertainty estimation is often conducted in empirical sciences [16-24]. One can distinguish the branch based on an analytical approach and the one using the Monte Carlo Method [25]. The former is often infeasible due to the complexity of the model, while the latter requires a significant number of additional objective function calls, increasing the computational time. In this paper we investigate a function shape using, for the second time, the space sampling collected during the optimization process with an evolutionary algorithm.

Multiple parameter study methods have been developed [26]. Most are based on Monte Carlo sampling; sapling on a predefined grid (mesh); or the vector parameter study (where sampling goes along the set vector). To provide a proper comparison in our test-case, the repositories containing the first two above-mentioned sampling routines are collated with the repository containing the genetic algorithm space sampling assembled in one of its runs.

All graphical visualizations have been prepared with either Matlab [27] or Gnuplot [28] plotting regimes.

TABLE I
OUTLINE OF THE AH METHOD

Stage		Name	Description
Preprocessing	A	Raw Dataset	input repository of the space sampling
	B	Threshold Cut	all points with a function value above the given threshold, y_{CUT}, are removed
	C	Minimum Distance Rule	all points having a closer distance to a point with a better function value than the given minimum distance, d_{min}, are sieved out
	D	Clusterization	detection of the points lying in the attraction region of the global optimum – application of NBC algorithm
Sampling Analysis	E	Statistics	information about statistical distribution of the points in the attraction region of the global optimum (ranges, moments, covariation and correlation matrices)
	F	Histograms	information about distribution of the points in the attraction region of the global optimum, based on distances between the points
Uncertainty Determination	G	Front-Line Algorithm	detection of the points located closest to the plane of uncertainty determination
	H	Parameter Uncertainties	parameter uncertainties determination for the given plane, y_{PU}

III. THE AH METHOD DESCRIPTION

The AH Method consists of eight stages labeled A to H consecutively (Table I). The processing of the initial repository is divided into three phases (modules). The preprocessing module comprises stages A to D and aims to isolate from the repository, the points located in the attraction region of the optimum. Stages E and F form the sampling analysis module that delivers a quantitative description of the distribution of the points in space. Finally, stages G and H constitute the uncertainty determination module. The uncertainty is interpreted as the tolerance boundaries of the parameter set, according to the user-specified increase in the function value above the located minimum.

From the geometric point of view, this is identical to the extreme (maximal and minimal) values of the parameters belonging to the contour line, being the result of cutting the function surface by the threshold plane at the user-specified height. The AH Method is especially useful when dealing with functions describing some statistical relationships, e.g. χ^2-test for goodness-of-fit. In this case the outcome is referred to as uncertainty intervals in the field of statistics.

The AH Method needs four arguments to be specified by the user:

- y_{CUT} – (stage B) function threshold value, above which the points are removed out from the repository;
- d_{min} – (stage C) minimum distance between the points in the repository;
- k – (stage D) cardinality of k-nearest neighbors set in the NBC algorithm;
- y_{PU} – (stage G) position of the threshold plane for parameter uncertainty determination. It defines a slice of the surface that is going to be investigated by the AH Method.

The description of all stages in the AH Method is contained in the subsections below.

A. Preprocessing Module

In stage A, datasets are as yet unprocessed by the AH Method. The raw sampling of any sampling routine used for function shape investigation is supposed to cover the whole domain. The genetic algorithm has an apparent inclination to concentrate the sampling in the attraction region of the optimum. This hallmark makes it a useful Artificial Intelligence (AI) space sampler for obtaining a desired repository in a less time-consuming manner than classical sampling routines.

In stage B, the samplings of different attraction regions are separated by removing points with a too high function value from the repository and thus introducing gaps between the points concentrated around particular optimums. These gaps significantly simplify clusterization in stage D. The choice of the proper y_{CUT} value is crucial here. It should be:

low enough to introduce gaps between the sampling of different attraction regions;

low enough to shorten the computational time of subsequent stages by eliminating points useless for function shape investigation around the optimum;

high enough not to interfere with the accuracy of the parameter uncertainties estimation.

In stage C, the repository is sieved. Points are placed in an ascending queue according to their function value and validated one by one. If a currently processed point lies closer than the minimum distance, d_{min}, to any of the already accepted points (the ones with a lower function value), it is sieved out from the repository. It brings no extra information of the function surface to its shape investigation and thus the repository is further downsized.

Stage D focuses on labeling the points, i.e. recognizing them as belonging to the attraction region of a particular optimum, or rejecting them as noise. This clusterization process is an application of the NBC algorithm [14]. The user must define the cardinality of k-nearest neighbors set.

For most real-case data, the stages of the preprocessing module should be repeated a few times to find the optimal values of its arguments: y_{CUT}, d_{min}, k. Moreover, any of the stages may be substituted with a procedure of similar workings.

After completing the preprocessing module, no further removal of points occurs. The resultant repository is limited to the set of points located in the attraction region of the optimum and subsequent stages are conducted using only this sampling.

B. Sampling Analysis Module

The quantitative description of the sampling distribution is done in two stages, E and F, previously presented in [29]. Only the points' parameter vectors are analyzed, their function values are neglected. The pivotal question addressed by the sampling analysis module is whether the sampling of the optimum attraction region is sufficient for parameter uncertainties estimation.

Stage E is a statistical analysis of a repository, containing N points. It uses the following equations for empirical parameters. Each of them in an unbiased estimator:

- the mean value of the i-th dimension:

$$\bar{x}_i = \frac{1}{N}\sum_{k=1}^{N} x_{ik} \tag{1}$$

- the standard deviation of the i-th dimension:

$$s_i = \sqrt{\frac{1}{N-1}\sum_{k=1}^{N}(x_{ik} - \bar{x}_i)^2} \tag{2}$$

- the standard deviation of the i-th dimension normalized to the domain range of that parameter:

$$s_i^{norm} = \frac{s_i}{samplingRange_i} \tag{3}$$

- the skewness of the i-th dimension:

$$SKE_i = \frac{N}{(N-1)(N-2)} \cdot \frac{\sum_{k=1}^{N}(x_{ik} - \bar{x}_i)^3}{s_i^3} \tag{4}$$

- the kurtosis of the i-th dimension:

$$Kurt_i = \frac{N(N+1)}{(N-1)(N-2)(N-3)} \cdot \frac{\sum_{k=1}^{N}(x_{ik} - \bar{x}_i)^4}{s_i^4} - 3 \cdot \frac{(N-1)^2}{(N-2)(N-3)} \tag{5}$$

- the covariance between i-th and j-th dimensions:

$$s_{i,j} = \frac{1}{N-1}\sum_{k=1}^{N}(x_{ik} - \bar{x}_i)(x_{jk} - \bar{x}_j) \tag{6}$$

- the correlation between i-th and j-th dimensions:

$$r_{i,j} = \frac{s_{i,j}}{s_i \cdot s_j} \tag{7}$$

These empirical parameters deliver information about sampling distribution separately for each dimension (the moments), and about linear dependences between pairs of dimensions (the covariance and correlation matrices). Based on the resulting information one can deduce a repository's quality.

Stage F follows up the sampling study with histogramic analysis of the repository. The deductions are brought out on the strength of the distances between points. The information about repository presented in stage F consists of:

- Space Sampling Histogram (SSH), which is carried out for each dimension separately. The points are allocated into a user-specified number of bins of equal width. It clarifies whether the sampling is free from gaps within the optimum attraction region;
- Distance from Best Histogram (DBH), i.e. the numbers of points located in the concentric adjoining hyper-rings with user-specified width. This allows one to check if the number of points in the consecutive distance intervals from the best point increases proportionally to the volume of the hyper-rings;
- Vertex Degree Histogram (VDH), where the repository is transformed into an Euclidean Graph with user-specified maximum edge length. If most vertices have the same degree, the points are uniformly distributed in the attraction region.

C. *Uncertainty Determination Module*

At this phase of processing, the repository has already been positively verified by the sampling analysis module. It stores only the points lying in the optimum attraction region and contains precise enough information about the function shape in the vicinity of the minimum. This final module of the AH Method focuses on the estimation of the uncertainty intervals of the parameter vector values for the user-specified threshold plane, y_{PU}. It is divided into two stages: G and H.

In stage G, the points located closest to the plane of uncertainty determination are labeled. This subset contains information about the shape of the uncertainty contour and allows for conclusions regarding the accuracy of uncertainty estimation on the strength of the given space sampling repository. This procedure, the Front-Line Algorithm (FLA), was originally presented in [30].

The inspiration for FLA comes from the martial strategy where two finite sets of soldiers (dots on a map), belonging to two involved parties, are separated by an abstract and artificial frontline. As with the uncertainty contour resulting from a repository of finite size, the exact location of the frontline is unknown and may only be estimated to some degree through the knowledge of the points' (soldiers') positions.

The greater the spread of points closest to the frontline, the poorer the quality of its estimated position, and thus the accuracy of uncertainty estimation.

In the FLA, each point in the repository receives at least one of the following labels recognizing it as belonging to the appropriate set. Each label has a shorthand form, specified after the coma (Fig. 1):

- p_{BEST}, $BEST$ – point with the minimal $f(\boldsymbol{x})$
- Allies, A – set of points with $f(\boldsymbol{x}) \leqslant y_{PU}$;
- Enemies, E – set of points with $f(\boldsymbol{x}) > y_{PU}$;
- Whole, Ω – set of all points in the optimum attraction region;

$$A \cup E = \Omega; A \cap E = \emptyset$$

- AllySoldiers, AS – set of points belonging to the Allies. Each one is the closest such point to at least one Enemy. If necessary, the set is replenished with the extreme points in the Allies;

$$a_S \in AS \Leftrightarrow a_S \; is \; extreme \; in \; A \quad \vee \quad \exists e \in E \quad \forall a \in A \quad dist(a_S, e) \leqslant dist(a, e)$$

- EnemySoldiers, ES – set of points belonging to the Enemies. Each one is the closest such point to at least one Ally;

$$e_S \in ES \Leftrightarrow \exists a \in A \quad \forall e \in E \quad dist(e_S, a) \leqslant dist(e, a)$$

- Soldiers, S - sum of the AllySoldiers and the EnemySoldiers sets;

$$AS \subset A; \quad AS \cup ES = S$$
$$ES \subset E; \quad AS \cap ES = \emptyset$$

- AllySoldierFriends, ASF – set of points belonging to the Allies yet not belonging to the AllySoldiers. Each one is the closest such point to at least one AllySoldier;

$$a_{SF} \in ASF \Leftrightarrow \exists a_S \in AS \quad \forall a \in A \setminus AS \quad dist(a_{SF}, a_S) \leqslant dist(a, a_S)$$

- EnemySoldierFriends, ESF – set of points belonging to the Enemies yet not belonging to the EnemySoldiers. Each one is the closest such point to at least one EnemySoldier;

$$e_{SF} \in ESF \Leftrightarrow \exists e_S \in ES \quad \forall e \in E \setminus ES \quad dist(e_{SF}, e_S) \leqslant dist(e, e_S)$$

- SoldierFriends, SF – sum of the AllySoldierFriends and the EnemySoldierFriends sets;

$$ASF \subset A \setminus AS; \quad ASF \cup ESF = SF$$
$$ESF \subset E \setminus ES; \quad ASF \cap ESF = \emptyset$$

- AllyFrontLines, AFL – sum of the AllySoldiers and the AllySoldierFriends sets;

$$AS \cup ASF = AFL; \quad AS \cap ASF = \emptyset$$

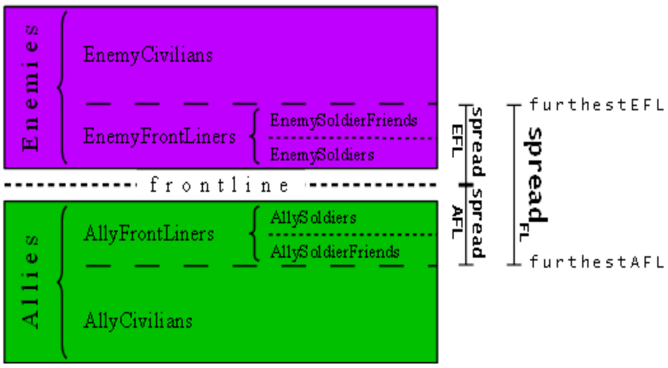


Fig. 1. Scheme of sampling division by Front-Line Algorithm

- EnemyFrontLiners, EFL – sum of the EnemySoldiers and EnemySoldierFriends sets;

$$ES \cup ESF = EFL; \quad ES \cap ESF = \emptyset$$

- FrontLiners, FL – sum of the AllyFrontLiners and the EnemyFrontLiners sets;

$$AFL \cup EFL = FL; \quad AFL \cap EFL = \emptyset$$

- AllyCivilians, AC – set of points belonging to the Allies yet not belonging to the AllyFrontLiners;

$$AC = A \setminus AFL$$

- EnemyCivilians, EC – set of points belonging to the Enemies yet not belonging to the EnemyFrontLiners;

$$EC = E \setminus EFL$$

- Civilians, C – sum of the AllyCivilians and the EnemyCivilians sects;

$$AC \cup EC = C; AC \cap EC = \emptyset$$

The FLA also introduces the following factors:

- furthestAllyFrontLiner, *furthest*$_{AFL}$ – lowest $f(\boldsymbol{x})$ value of a point in the AFL set;

$$furthest_{AFL} = \min_{x \in AFL}\left(f(\boldsymbol{x})\right)$$

- spreadOfAllyFrontLiners, *spread*$_{AFL}$ – difference between $furthest_{AFL}$ and y_{PU};

$$spread_{AFL} = furthest_{AFL} - y_{PU}$$

```
1. Create an etiquette vector based on the length of the
   point number.
2. On the grounds of the $y_{PU}$ value and the $f(\boldsymbol{x})$ values of
   particular points, divide the points into the A and the
   E sets. Mark the BEST point.
3. On the strength of distances between the points in the A
   and the E sets, assign the points to the AS and the ES
   sets. If necessary, replenish the AS set with the extreme
   points from the A set.
4. Allocate the points, which belong to the A set yet do not
   belong to the AS set, into the AC set. Allocate the
   points, which belong to the E set yet do not belong to
   the ES set, into the EC set.
5. Based on the distances between the points in the AC and
   the AS sets, find and label the points belonging to the
   ASF set. Remove from the AC set, the points ranked as
   belonging to the ASF set.
6. Based on the distances between the points in the EC and
   the ES sets, find and label the points belonging to the
   ESF set. Remove from the EC set, the points ranked as
   belonging to the ESF set.
7. Recognize the sets: Ω, S, SF, FL, C as the sums of the
   appropriate sets.
8. Determine the values of the factors: $furthest_{AFL}$,
   $furthest_{EFL}$, $spread_{AFL}$, $spread_{EFL}$, $spread_{FL}$.
```

Fig. 2. Front-Line Algorithm pseudocode

- furthestEnemyFrontLiner, *furthest*$_{EFL}$ – highest $f(\boldsymbol{x})$ value of a point in the EFL set;

$$furthest_{EFL} = \max_{x \in EFL}\left(f(\boldsymbol{x})\right)$$

- spreadOfEnemyFrontLiners, *spread*$_{EFL}$ – difference between $furthest_{EFL}$ and y_{PU};

$$spread_{EFL} = furthest_{EFL} - y_{PU}$$

- spreadOfFrontLiners, *spread*$_{FL}$ – difference between $spread_{EFL}$ and $spread_{AFL}$;

$$spread_{FL} = spread_{EFL} - spread_{AFL}$$

```
$x_{BEST}$ = $p_{BEST}$.x; $x_{AS}$ = $p_{AS}$.x; $x_{CURRENT}$ = $x_{AS}$; $y_{CURRENT}$ = $p_{AS}$.y;
step = unitVector($x_{BEST}$, $x_{AS}$);
if (|$step_i$| < $step_{min}$) step *= $step_{min}$ / |$step_i$|;
while (1)
  $x_{NEXT}$ = $x_{CURRENT}$ + step; $y_{NEXT}$ = $f_{APPROX}$($x_{NEXT}$);
  if ($y_{NEXT}$ > $y_{PU}$ || $y_{NEXT}$ < $y_{CURRENT}$)
    if ($step_i$ = $step_{min}$) break; //ends while loop
    else
      step /= 3;
      if (|$step_i$| < $step_{min}$) step *= $step_{min}$ / |$step_i$|;
  else
    $y_{CURRENT}$ = $y_{NEXT}$;
    $x_{CURRENT}$ = $x_{NEXT}$;
if ($x_{CURRENT}$ != $x_{AS}$) $p_{APPROX}$ = Point($x_{CURRENT}$, $y_{CURRENT}$);
else $p_{APPROX}$ = NULL;
```

Fig. 3. Further Uncertainty Estimation algorithm pseudocode

Fig.2 presents the Front-Line Algorithm. Its performance strongly depends on a fast determination of the distances between the points. Hence it is recommended to speed-up calculations with a pre-prepared triangular matrix of distances.

The points, belonging to the S set, lie in direct neighborhood of the uncertainty contour. This set of points is extended with the SF set and forms the FL set.

The enhanced set of points (FL) contains information about the shape of the uncertainty contour. The vertical spread of points in the FL set (visible in 3D plots) represents the degree of their scattering around the contour and thus delivers guidelines about the accuracy of the uncertainty estimation.

In stage H of the AH Method the uncertainties are determined.

- The min-max values among the extreme points in the set A are adapted as the uncertainty intervals of the particular parameters.
- The *spread*$_{FL}$ factor value (together with the 4 remaining auxiliary factors) is adapted as the information about the accuracy of the uncertainty determination.

The determined uncertainty intervals are apparently always slightly underestimated due to the finite size of the repository. To tackle this issue, we propose the Further Uncertainty Estimation (FUE) algorithm (Fig.3).

The FUE algorithm makes an attempt to extend the potentially underestimated uncertainty intervals by finding a point (from outside of the repository) for which $f_{approx}(\boldsymbol{x}) \le y_{PU}$. The search is conducted along the half-line $p_{BEST}\,p_{AS}$. The calculation of the approximated function value for a parameter vector, **x**, is carried out using the linear approximation. The weighted mean of the neighbor points' function values is adapted as $f_{approx}(\boldsymbol{x})$ value. The distances from **x** are used as weights.

Operations of the AH Method are exemplified with its test-case results in Section V.

```
1.  Generate the initial population of 40 points randomly,
    with uniform distribution from search space.
2.  Determine the function value for all points in the
    population.
3.  For 120 generations perform:
    a)  truncation selection: remove from the population
        the 24 points (60%) with the worst function value;
    b)  heuristic crossover2: using the remaining 16 points
        as parents, fill the population, i.e. create 24
        offspring:
            •  Parents are selected using the roulette-
               wheel method.
            •  A point cannot be selected as a parent more
               than 5 times in one generation. It also
               cannot be crossed with itself (incest
               prevention);
    c)  Gaussian mutation: each of the 40 points has a 10%
        chance of being mutated. The parameter vector of
        the point is modified by a value drawn from the
        Gaussian distribution truncated to the domain of
        the search space. The standard deviation of the
        drawing is equal to 0.20 multiplied by the
        particular parameter range and thus may differ
        between dimensions.
4.  Assemble all points, for which a function value was
    calculated, into the sampling repository of the search
    space.
```

Fig. 4. Outline of the Genetic Algorithm

```
foreach dimension
   if ($f_{fitness}$(parent$_1$) >= $f_{fitness}$(parent$_2$))
      $x_i$(offspring) = $\lambda_i$($x_i$(parent$_2$)-$x_i$(parent$_1$)) + $x_i$(parent$_2$);
   else $x_i$(offspring) = $\lambda_i$($x_i$(parent$_1$)-$x_i$(parent$_2$)) + $x_i$(parent$_1$);
   while ($x_i$(offspring) < min$_i$) $x_i$(offspring) += range$_i$;
   while ($x_i$(offspring) > max$_i$) $x_i$(offspring) -= range$_i$;

where:
$\lambda_i$      - random number uniformly drawn from [0,1] interval,
f (·)     - function value of the individual,
$x_i$ (·)    - ith parameter value of the individual.
```

Fig. 5 Heuristic crossover2 pseudocode

IV. The Genetic Algorithm and the Test-case Definitions

The GA in this paper is considered as an intelligent space sampler, able not only to find the optimum but also to deliver the sampling for function shape investigation. It is of traditional structure (Fig.4) and all its parameter values have been adjusted to the two-dimensional test-case. The population size remains constant in subsequent generations, and has been set at 40 individuals. Each new generation results from selection, crossover and mutation.

The chosen selection and mutation are standard operators. The truncation selection concentrates the searching process on the interesting areas of the space. It focuses on the previously revealed attraction regions. The gaussian mutation introduces new genetic material to the population. It increases its diversity and prevents the algorithm from getting stuck in a local optimum attraction region. The detailed description of these two genetic operators is presented in Fig.4.

Most crucial is the heuristic crossover2 [5] (Fig.5). It aims at moving the searching process towards the prospecting areas of the space. Based on the objective function values of the parents, an offspring is created in a place where it has potentially better fitness than either of its parents.

The genetic algorithm terminates after 120 generations. This means that repository assembled from one of its runs contains, on average, 3112 points.

The AH Method is dedicated for multidimensional function shape investigation problems. In this paper, they are represented by the F7 function by Schwefel [2]. The two dimensional test-case was selected for presentation purposes. Each point in the search space is then a two-valued vector with coordinates $\boldsymbol{x}\, \epsilon\, [-40.0, 15.0]^2$. Fig.6 presents the F7 function shifted up by 50.0. The appropriate formula is:

$$f(\boldsymbol{x}) = F7(\boldsymbol{x}) + 50.0 =$$

$$-x_1 \cdot sin\sqrt{|x_1|} - x_2 \cdot sin\sqrt{|x_2|} + 50.0 \qquad (8)$$

During the minimization process the best solution $\boldsymbol{x}_{min} = (-25.877, -25.877)$ can be found with the value of the function $f(\boldsymbol{x}_{min}) \approx 1.834$.

The position of the threshold plane for parameter uncertainty determination can be defined in two analogous ways: absolute and relative. In the first case the threshold level is set as a single value, e.g. $y_{PU} = 10.0$. Similarly, the threshold may be selected with reference to its distance to the function value in the found optimum, e.g. y_{PU}+= 8.166 which together with $f_{min} = 1.834$ totals 10.0. The tolerance boundaries for the given threshold plane $y_{PU} = 10.0$ are:

$$\forall_{i=1,2} \qquad \boldsymbol{x_i} \in [-\mathbf{33.502}, -\mathbf{17.928}]$$

$$\boldsymbol{that\ is} \qquad \boldsymbol{x_i} = -\mathbf{25.877}^{+7.949}_{-7.625} \qquad (9)$$

This notation covers the issue of asymmetry in uncertainty determination of parameter values.

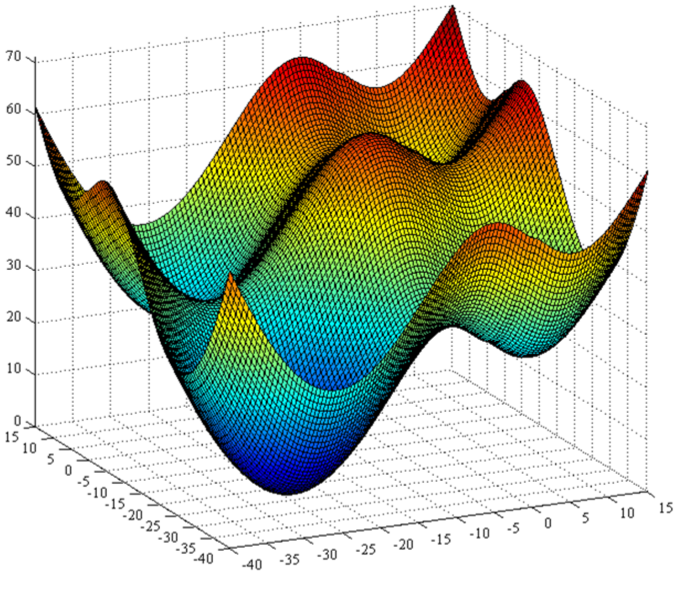


Fig. 6. Test-case Objective Function

V. The Test-Case Results

In the presented test-case, repositories containing sampling through Monte Carlo process (Monte Carlo sampling) and points taken from the predefined mesh (sampling on a grid) are collated with the set containing the genetic algorithm space sampling assembled during a single run. The performance of the AH Method at each stage is evaluated based on all datasets. This provides an opportunity to scrutinize not only the way the AH Method processes a repository but also the quality of function shape investigation for different space sampling routines. Table II shows the compared types of space sampling.

To introduce randomness in `rand3k` and `ga3k` repositories each was generated ten times. The precision of minimum indication and the number of points in the global optimum attraction region were applied as comparison criteria and the best repository from each sampling routine was selected for further processing.

Tables III and IV present numbers of points for all 10 alternative sets of the `rand3k` and of the `ga3k` repositories. The most attractive ones are indicated by the bold font. The benchmark information about the `grid3k` repository is in the rightmost column. Variations within each group are insignificant. Results obtained for the selected sets may be repeated with any of the others.

TABLE II
REPOSITORIES WITH TEST-CASE DATA

repository	sampling routine	number of points
`grid3k`	points distributed uniformly, every 1.0 on a mesh covering whole domain	56^2 = 3136 points
`rand3k`	points distributed randomly with uniform probability within the domain (Monte Carlo Method)	3120 points
`ga3k`	points constitute an assembly from one run of the genetic algorithm defined in Fig.4	approx. 3112 points

TABLE III
COMPARISON OF 10 RAND3K REPOSITORIES

Stage	rep. 1	rep. 2	rep. 3	rep. 4	rep. 5	rep. 6	rep. 7	rep. 8	rep. 9	rep. 10	**Average**	Best	Worst	**grid3k**
A	3120	3120	3120	3120	3120	3120	3120	3120	3120	3120	**3120**.0	3120	3120	3136
B	986	974	958	951	1008	1008	953	981	1021	999	**983**.9	1021	951	964
C	720	694	700	694	724	704	699	700	735	711	**708**.1	735	694	964
D	526	513	524	504	533	517	509	507	521	512	**516**.6	533	504	708
Minimum	1.853	1.840	1.923	1.919	1.892	1.907	1.885	1.878	1.864	1.877	**1**.884	1.840	1.923	1.838

TABLE IV
COMPARISON OF 10 GA3K REPOSITORIES

Stage	rep. 1	rep. 2	rep. 3	rep. 4	rep. 5	rep. 6	rep. 7	rep. 8	rep. 9	rep. 10	**Average**	Best	Worst	**grid3k**
A	3096	3104	3141	3102	3094	3097	3126	3101	3125	3122	**3110**.8	3141	3094	3136
B	2934	2947	2934	2946	2948	2930	2951	2951	2970	2960	**2947**.1	2970	2930	964
C	380	363	365	340	376	385	360	369	351	380	**366**.9	385	340	964
D	362	350	351	326	357	374	349	360	337	368	**353**.4	374	326	708
Minimum	1.834	1.834	1.834	1.834	1.834	1.834	1.834	1.834	1.834	1.834	**1**.834	1.834	1.834	1.838

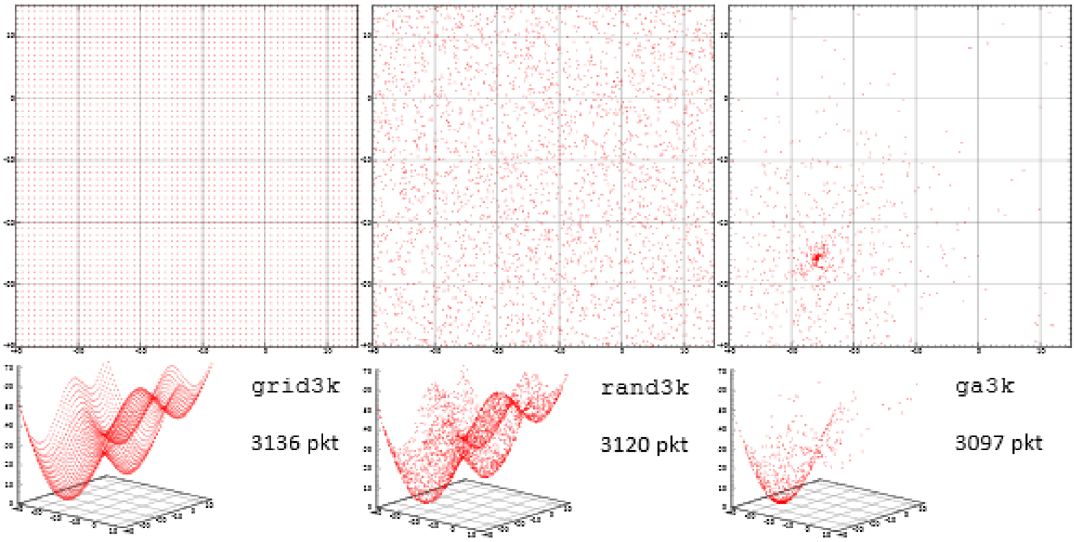


Fig. 7. Outcome of the AH Method: stage A – grid3k (left), rand3k (center), ga3k (right)

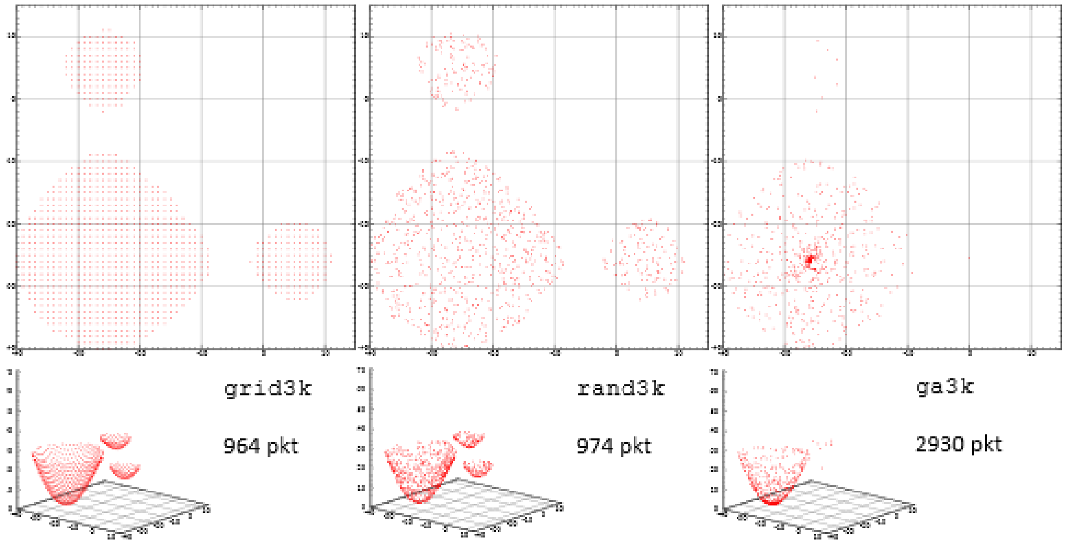


Fig. 8. Outcome of the AH Method: stage B – grid3k (left), rand3k (center), ga3k (right)

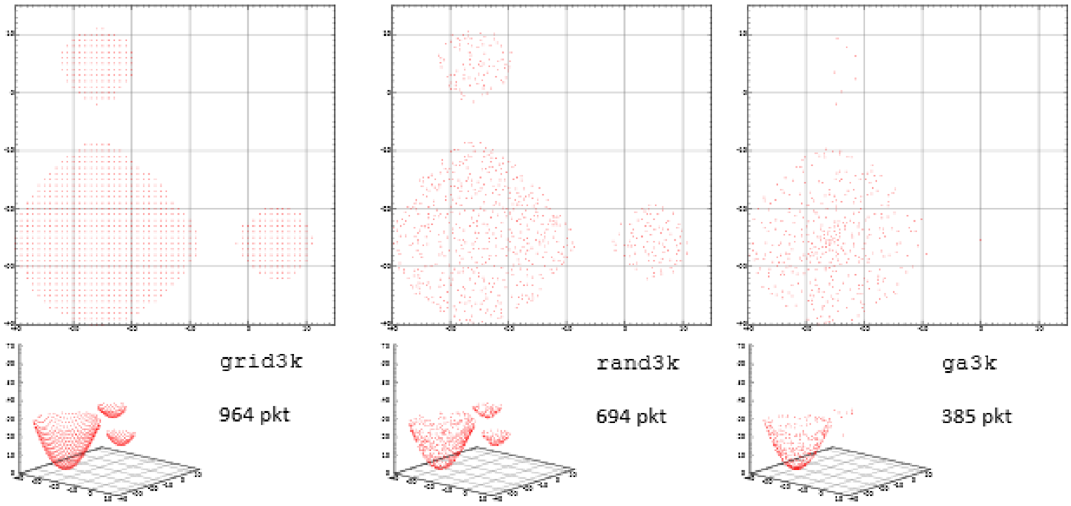


Fig. 9. Outcome of the AH Method: stage C – grid3k (left), rand3k (center), ga3k (right)

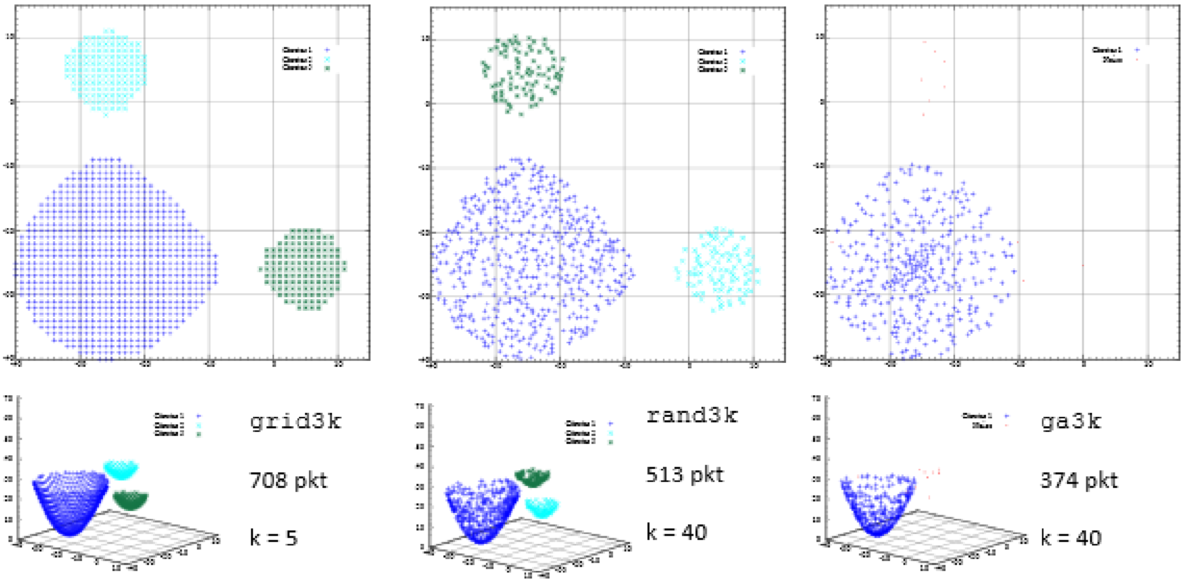
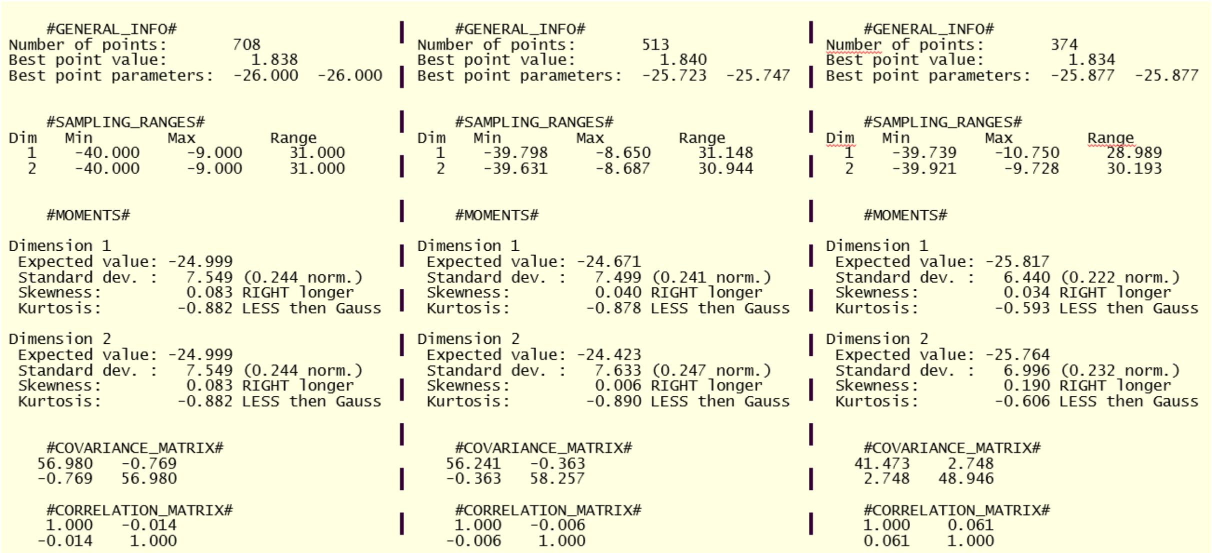


Fig. 10. Outcome of the AH Method: stage D – grid3k (left), rand3k (center), ga3k (right)



```
    #GENERAL_INFO#
Number of points:        708
Best point value:          1.838
Best point parameters:  -26.000   -26.000

    #SAMPLING_RANGES#
Dim    Min         Max          Range
  1    -40.000     -9.000       31.000
  2    -40.000     -9.000       31.000

    #MOMENTS#

Dimension 1
 Expected value: -24.999
 Standard dev. :   7.549 (0.244 norm.)
 Skewness:         0.083 RIGHT longer
 Kurtosis:        -0.882 LESS then Gauss

Dimension 2
 Expected value: -24.999
 Standard dev. :   7.549 (0.244 norm.)
 Skewness:         0.083 RIGHT longer
 Kurtosis:        -0.882 LESS then Gauss

    #COVARIANCE_MATRIX#
   56.980    -0.769
   -0.769    56.980

    #CORRELATION_MATRIX#
    1.000    -0.014
   -0.014     1.000
```

```
    #GENERAL_INFO#
Number of points:        513
Best point value:          1.840
Best point parameters:  -25.723   -25.747

    #SAMPLING_RANGES#
Dim    Min         Max          Range
  1    -39.798     -8.650       31.148
  2    -39.631     -8.687       30.944

    #MOMENTS#

Dimension 1
 Expected value: -24.671
 Standard dev. :   7.499 (0.241 norm.)
 Skewness:         0.040 RIGHT longer
 Kurtosis:        -0.878 LESS then Gauss

Dimension 2
 Expected value: -24.423
 Standard dev. :   7.633 (0.247 norm.)
 Skewness:         0.006 RIGHT longer
 Kurtosis:        -0.890 LESS then Gauss

    #COVARIANCE_MATRIX#
   56.241    -0.363
   -0.363    58.257

    #CORRELATION_MATRIX#
    1.000    -0.006
   -0.006     1.000
```

```
    #GENERAL_INFO#
Number of points:        374
Best point value:          1.834
Best point parameters:  -25.877   -25.877

    #SAMPLING_RANGES#
Dim    Min         Max          Range
  1    -39.739    -10.750       28.989
  2    -39.921     -9.728       30.193

    #MOMENTS#

Dimension 1
 Expected value: -25.817
 Standard dev. :   6.440 (0.222 norm.)
 Skewness:         0.034 RIGHT longer
 Kurtosis:        -0.593 LESS then Gauss

Dimension 2
 Expected value: -25.764
 Standard dev. :   6.996 (0.232 norm.)
 Skewness:         0.190 RIGHT longer
 Kurtosis:        -0.606 LESS then Gauss

    #COVARIANCE_MATRIX#
   41.473     2.748
    2.748    48.946

    #CORRELATION_MATRIX#
    1.000     0.061
    0.061     1.000
```

Fig. 11. Outcome of the AH Method: stage E – grid3k (left), rand3k (center), ga3k (right)

A. *Preprocessing Module*

The isolation of the points located in the attraction region of the optimum is divided into four stages with different objectives. The outcomes of each stage are presented in triplets to facilitate comparison in Figs 8÷11. 3D plots are shown below each 2D counterpart. The values of three preprocessing module arguments have been specified as following:

- the threshold cut value has been set as:

$$y_{CUT} = 28.0 \tag{10}$$

- the minimum distance between points has been set as:

$$d_{min} = 0.5 \tag{11}$$

- the cardinality of the k-nearest neighbors set has been set as:

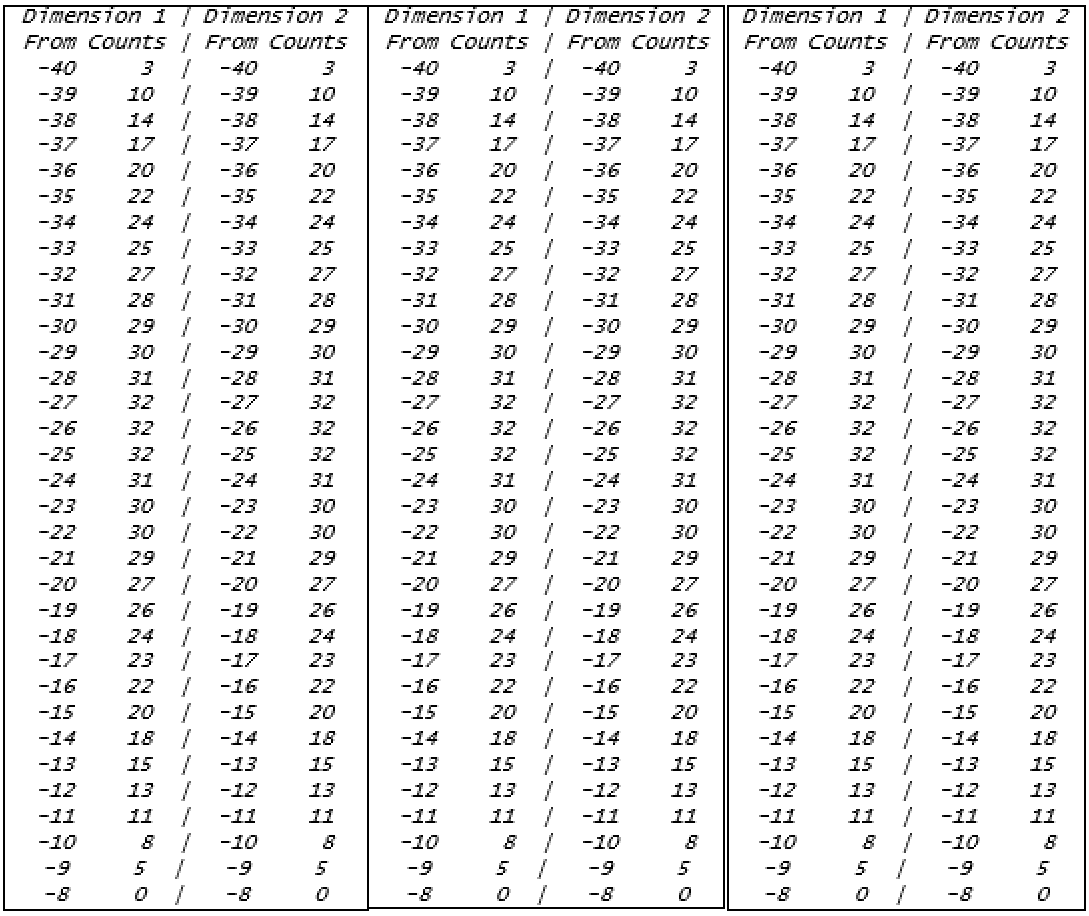
$$\begin{aligned} k &= 5 && \textit{for the grid3k repository} \\ k &= 40 && \textit{for the rand3k and ga3k repository} \end{aligned} \tag{12}$$

B. *Sampling Analysis Module*

The quantitative description of the sampling distribution is done in two stages: E and F. The pivotal question addressed by the sampling analysis module is whether the sampling of the optimum attraction region is sufficient for parameter uncertainties estimation.

Stage E is a statistical analysis of the repository, presented in the form of printouts from the program ScanRep, which implements the AH Method. The outcome is shown in Fig. 12.

Results of stage E prove that the quality of the `ga3k` repository is comparable with the `grid3k` and the `rand3k`. The use of genetic algorithm space sampling for parameter uncertainty estimation in the test-case should bring similar



Dimension 1		Dimension 2		Dimension 1		Dimension 2		Dimension 1		Dimension 2	
From	Counts	From	Counts	From	Counts	From	Counts	From	Counts	From	Counts
-40	3	-40	3	-40	3	-40	3	-40	3	-40	3
-39	10	-39	10	-39	10	-39	10	-39	10	-39	10
-38	14	-38	14	-38	14	-38	14	-38	14	-38	14
-37	17	-37	17	-37	17	-37	17	-37	17	-37	17
-36	20	-36	20	-36	20	-36	20	-36	20	-36	20
-35	22	-35	22	-35	22	-35	22	-35	22	-35	22
-34	24	-34	24	-34	24	-34	24	-34	24	-34	24
-33	25	-33	25	-33	25	-33	25	-33	25	-33	25
-32	27	-32	27	-32	27	-32	27	-32	27	-32	27
-31	28	-31	28	-31	28	-31	28	-31	28	-31	28
-30	29	-30	29	-30	29	-30	29	-30	29	-30	29
-29	30	-29	30	-29	30	-29	30	-29	30	-29	30
-28	31	-28	31	-28	31	-28	31	-28	31	-28	31
-27	32	-27	32	-27	32	-27	32	-27	32	-27	32
-26	32	-26	32	-26	32	-26	32	-26	32	-26	32
-25	32	-25	32	-25	32	-25	32	-25	32	-25	32
-24	31	-24	31	-24	31	-24	31	-24	31	-24	31
-23	30	-23	30	-23	30	-23	30	-23	30	-23	30
-22	30	-22	30	-22	30	-22	30	-22	30	-22	30
-21	29	-21	29	-21	29	-21	29	-21	29	-21	29
-20	27	-20	27	-20	27	-20	27	-20	27	-20	27
-19	26	-19	26	-19	26	-19	26	-19	26	-19	26
-18	24	-18	24	-18	24	-18	24	-18	24	-18	24
-17	23	-17	23	-17	23	-17	23	-17	23	-17	23
-16	22	-16	22	-16	22	-16	22	-16	22	-16	22
-15	20	-15	20	-15	20	-15	20	-15	20	-15	20
-14	18	-14	18	-14	18	-14	18	-14	18	-14	18
-13	15	-13	15	-13	15	-13	15	-13	15	-13	15
-12	13	-12	13	-12	13	-12	13	-12	13	-12	13
-11	11	-11	11	-11	11	-11	11	-11	11	-11	11
-10	8	-10	8	-10	8	-10	8	-10	8	-10	8
-9	5	-9	5	-9	5	-9	5	-9	5	-9	5
-8	0	-8	0	-8	0	-8	0	-8	0	-8	0

Fig. 12a Outcome of the AH Method: stage F – SSH histogram: `grid3k` (left), `rand3k` (center), `ga3k` (right)

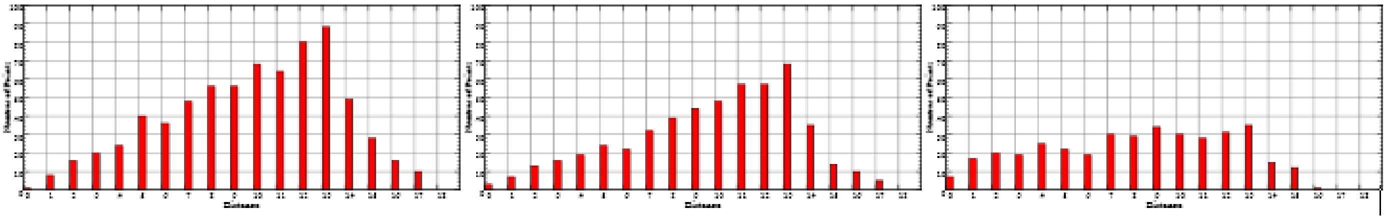


Fig. 12b Outcome of the AH Method: stage F – DBH histogram: `grid3k` (left), `rand3k` (center), `ga3k` (right)

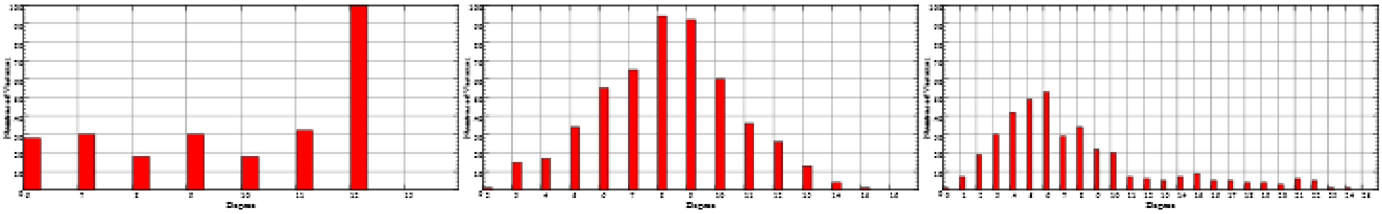


Fig. 12c Outcome of the AH Method: stage F – VDH histogram: `grid3k` (left), `rand3k` (center), `ga3k` (right)

effects to classical sampling routines. This assumption has been found true with the outcomes of the uncertainty determination module.

Stage F follows up the sampling study with histogramic analysis of the repository. Its outcome is presented in Figs 13a-c. The results confirm those of stage E.

C. *Uncertainty Determination Module*

In stage G, the points located closest to the plane of uncertainty determination are labeled. The outcome is presented in Fig.14, where the set S is marked, and in Fig.15, where the set FL is marked (sum of the S and the SF sets). The difference is easily determined by comparing these two triplets of plots.

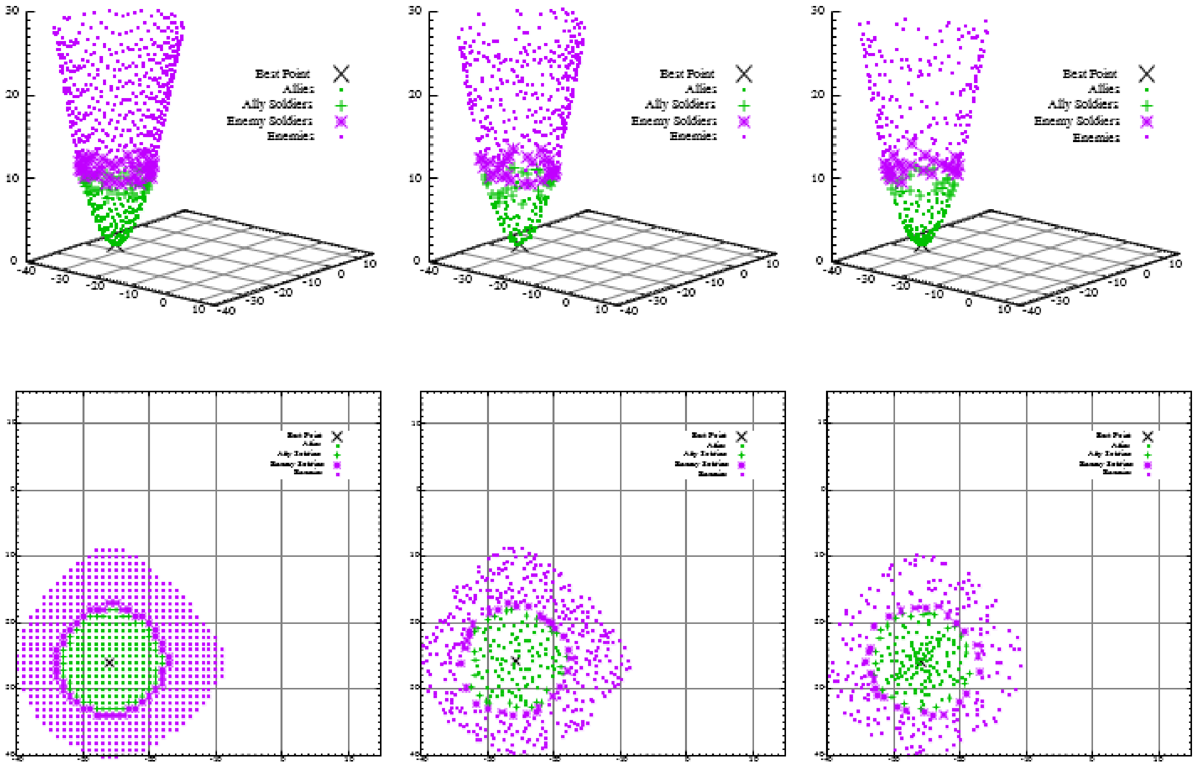


Fig.13. Outcome of the AH Method: stage G – `grid3k` (left), `rand3k` (center), `ga3k` (right)

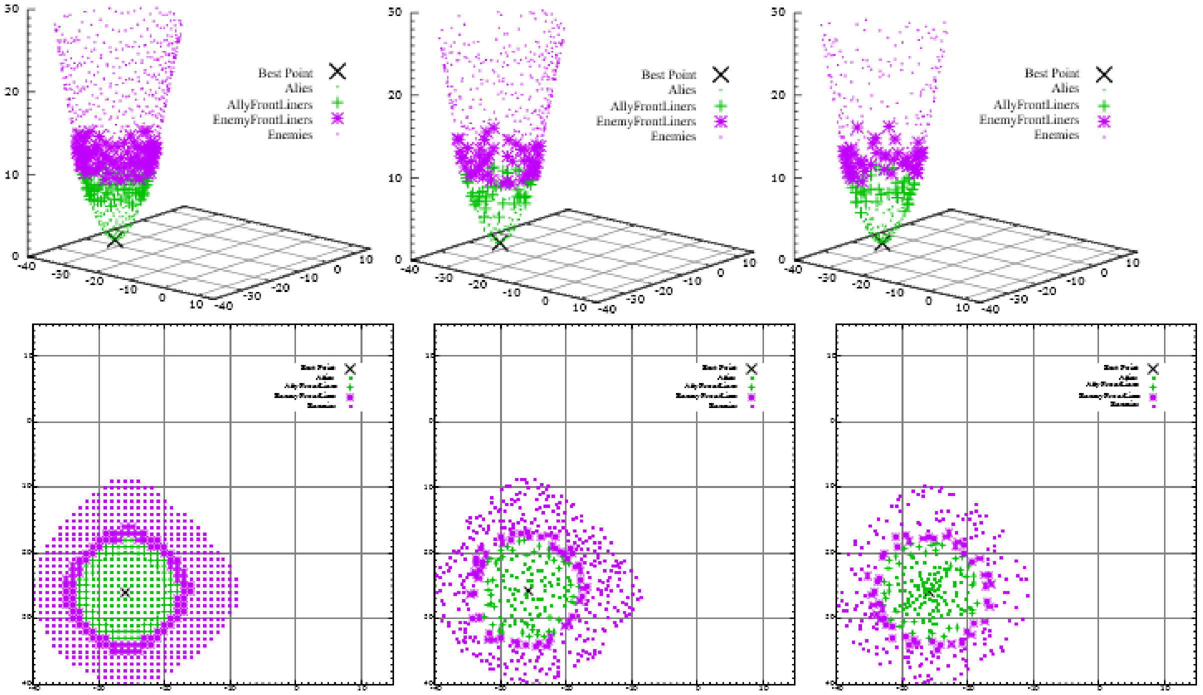
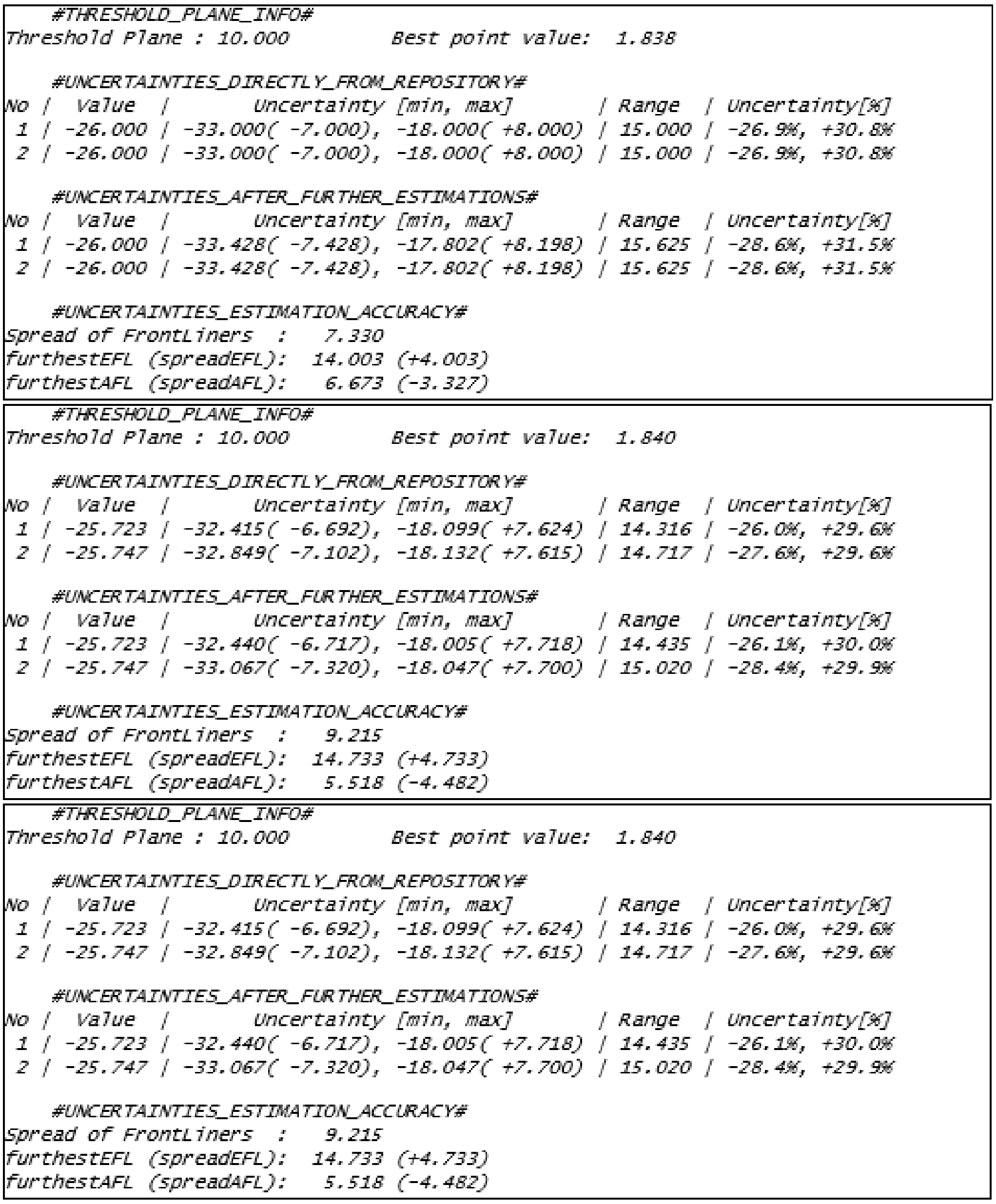


Fig.14. Outcome of the AH Method: stage G (enhanced) – `grid3k` (left), `rand3k` (center), `ga3k` (right)



```
    #THRESHOLD_PLANE_INFO#
Threshold Plane : 10.000         Best point value:  1.838

    #UNCERTAINTIES_DIRECTLY_FROM_REPOSITORY#
No |  Value   |        Uncertainty [min, max]        | Range  | Uncertainty[%]
 1 | -26.000 | -33.000( -7.000), -18.000( +8.000) | 15.000 | -26.9%, +30.8%
 2 | -26.000 | -33.000( -7.000), -18.000( +8.000) | 15.000 | -26.9%, +30.8%

    #UNCERTAINTIES_AFTER_FURTHER_ESTIMATIONS#
No |  Value   |        Uncertainty [min, max]        | Range  | Uncertainty[%]
 1 | -26.000 | -33.428( -7.428), -17.802( +8.198) | 15.625 | -28.6%, +31.5%
 2 | -26.000 | -33.428( -7.428), -17.802( +8.198) | 15.625 | -28.6%, +31.5%

    #UNCERTAINTIES_ESTIMATION_ACCURACY#
Spread of FrontLiners  :    7.330
furthestEFL (spreadEFL):  14.003 (+4.003)
furthestAFL (spreadAFL):   6.673 (-3.327)
```

```
    #THRESHOLD_PLANE_INFO#
Threshold Plane : 10.000         Best point value:  1.840

    #UNCERTAINTIES_DIRECTLY_FROM_REPOSITORY#
No |  Value   |        Uncertainty [min, max]        | Range  | Uncertainty[%]
 1 | -25.723 | -32.415( -6.692), -18.099( +7.624) | 14.316 | -26.0%, +29.6%
 2 | -25.747 | -32.849( -7.102), -18.132( +7.615) | 14.717 | -27.6%, +29.6%

    #UNCERTAINTIES_AFTER_FURTHER_ESTIMATIONS#
No |  Value   |        Uncertainty [min, max]        | Range  | Uncertainty[%]
 1 | -25.723 | -32.440( -6.717), -18.005( +7.718) | 14.435 | -26.1%, +30.0%
 2 | -25.747 | -33.067( -7.320), -18.047( +7.700) | 15.020 | -28.4%, +29.9%

    #UNCERTAINTIES_ESTIMATION_ACCURACY#
Spread of FrontLiners  :    9.215
furthestEFL (spreadEFL):  14.733 (+4.733)
furthestAFL (spreadAFL):   5.518 (-4.482)
```

```
    #THRESHOLD_PLANE_INFO#
Threshold Plane : 10.000         Best point value:  1.840

    #UNCERTAINTIES_DIRECTLY_FROM_REPOSITORY#
No |  Value   |        Uncertainty [min, max]        | Range  | Uncertainty[%]
 1 | -25.723 | -32.415( -6.692), -18.099( +7.624) | 14.316 | -26.0%, +29.6%
 2 | -25.747 | -32.849( -7.102), -18.132( +7.615) | 14.717 | -27.6%, +29.6%

    #UNCERTAINTIES_AFTER_FURTHER_ESTIMATIONS#
No |  Value   |        Uncertainty [min, max]        | Range  | Uncertainty[%]
 1 | -25.723 | -32.440( -6.717), -18.005( +7.718) | 14.435 | -26.1%, +30.0%
 2 | -25.747 | -33.067( -7.320), -18.047( +7.700) | 15.020 | -28.4%, +29.9%

    #UNCERTAINTIES_ESTIMATION_ACCURACY#
Spread of FrontLiners  :    9.215
furthestEFL (spreadEFL):  14.733 (+4.733)
furthestAFL (spreadAFL):   5.518 (-4.482)
```

Fig.15. Outcome of the AH Method: stage H – `grid3k` (top), `rand3k` (center), `ga3k` (bottom)

- Exact solution

$$[-33.502, -17.928]$$

$$x_i = -25.877^{+7.949}_{-7.625}$$

- Repository `grid3k`

$$[-33.000, -18.000] \xrightarrow{FUE} [-33.428, -17.802]$$

$$x_i = -26.000^{+8.000}_{-7.000} \qquad \square^{+8.198}_{-7.428}$$

- Repository `rand3k`

$$[-32.415, -18.099] \xrightarrow{FUE} [-32.440, -18.005]$$

$$x_1 = -25.723^{+7.624}_{-6.692} \qquad \square^{+7.718}_{-6.717}$$

$$[-32.849, -18.132] \xrightarrow{FUE} [-33.067, -18.047]$$

$$x_2 = -25.747^{+7.615}_{-7.102} \qquad \square^{+7.700}_{-7.320}$$

- Repository `ga3k`

$$[-33.003, -18.464] \xrightarrow{FUE} [-33.212, -18.137]$$

$$x_1 = -25.877^{+7.413}_{-7.126} \qquad \square^{+7.740}_{-7.335}$$

$$[-32.993, -18.640] \xrightarrow{FUE} [-33.479, -17.663]$$

$$x_2 = -25.877^{+7.237}_{-7.116} \qquad \square^{+8.214}_{-7.602}$$

Fig. 16. Outcome of the AH Method

In stage H, the uncertainties are determined. Fig.16 presents the outcome of stage H in the from of printouts from the program ScanRep. Fig.17 contains similar information but in a different, perhaps more readible form.

The test-case results prove that all three types of sampling estimate the uncertainties correctly. The inaccuracies are of the third significant digit, whereas the uncertainties are commonly rounded to the second one. This leads to the possibility of radical downsizing in the necessary point number and thus a remarkable reduction in the required computational time for the uncertainty estimation task. It makes the AH Method the perfect alternative in multidimensional space investigations, where a parameter study with classical sampling routines is infeasible due to too high a number of necessary objective function calls.

VI. Conclusions

This paper presents a new heuristic method of uncertainty estimation regarding the function shape investigation in the vicinity of the found optimal solution. The research compares three different ways of obtaining a space sampling around the region in question and reveals that a repository consisting of a sampling with a genetic algorithm run can compete with classical sampling routines used in parameter study, i.e., uniform and random ones.

The AH Method consists of eight stages arranged in three modules: preprocessing, sampling analysis, and uncertainty determination. It creates it a framework where a stage can be easily substituted for another analogous algorithm. The possible challenges for the user lie in the need to specify the values of three preprocessing module arguments. They should be case-matched and should allow one to isolate from the repository, the points located in the attraction region of the optimum. Thus, to facilitate this issue, we consider applying Self-Organizing Maps to the clusterization task (stage D) in the near future.

The AH Method has been primarily developed to assist the data evaluation in nuclear physics experiments, namely Coulomb Excitations [31]. The first real-case results proved the AH Method's fitness for the purpose. Future investigations in this field should cover introducing a procedure for determination of the proper position of the threshold plane for parameter uncertainty estimation based on the particular experimental data.

Acknowledgements

We wish to thank Anna Maria Piętak for her contribution to technical part of this work.

References

[1] A. P. Engelbrecht*: Computational Intelligence,* John Wiley & Sons Ltd., 2002.

[2] H.-P. Schwefel: *Numerical optimization of computer models*, Chichester: Wiley & Sons, 1981.

[3] D. E. Goldberg: *Genetic Algorithms in Search, Optimization and Machine Learning*, Kluwer Academic Publishers, Boston, MA, 1989.

[4] Z. Michalewicz: *Genetic Algorithms + Data Structures = Evolution Programs*, Springer-Verlag, 1999.

[5] Z. Michalewicz, D. B. Fogel: *How to Solve It: Modern Heuristics*, Springer Berlin, Heidelberg, 2004.

[6] J. Arabas: *Approximating the Genetic Diversity of Populations in the Quasi-Equilibrum State*, IEEE Transactions on Evolutionary Computation, 16, 5, 2012, https://doi.org/10.1109/TEVC.2011.2166157

[7] D. A. Piętak, P. J. Napiorkowski, Z. Walczak, J. Wojciechowski: *Application of Genetic Algorithm with Real Representation to COULEX Data Analysis*, Proc. Conference on Evolutionary Computation and Global Optimization, 2009.

[8] M. Ankerst, M. Breunig, H.-P. Kriegel, J. Sander: *OPTICS: Ordering points to identify the clustering structure*, Proc. 1999 ACM-SIGMOD International Conference on Management of Data (SIGMOD'99), pp. 49-60, Philadelphia, 1999, https://doi.org/10.1145/304181.304187

[9] M. Ester, H. Kriegel, J. Sander, X. Xu: *A density-based algorithm for discovering clusters in large spatial databases with noise*, Proc. International Conference on Knowledge Discovery and Data Mining (KDD'96), pp. 226-231, Portland Oregon, 1998.

[10] J. Han, M. Kamber: *Data mining: concepts and techniques*, Morgan Kaufmann, 2000, https://doi.org/10.1016/C2009-0-61819-5

[11] A. K. Jain, M. N. Murty, P. J. Flyn: *Data clustering: A review*, ACM Computing Surveys, 31(3): 264-323, 1999, https://doi.org/10.1145/331499.331504

[12] R. Weber, H.-J. Schek, S. Blott: *A quantitative analysis and performance study for similarity-search methods in high-dimensional spaces*, Proc. Conference on Very Large DataBases (VLDB'98), pp. 194-205, New York City, USA, 1998.

[13] R. Weber, S. Blott: *An approximation based data structure for similarity search*, Technical Report 24, ESPRIT project HERMES (no. 9141), 1997

[14] S. Zhou, Y. Zhao, J. Guan, J. Huang: NBC: *A Neighborhood-Based Clustering Algorithm,* Lecture Notes in Computer Science, Springer Berlin / Heidelberg, 2005, https://doi.org/10.1007/11430919_43

[15] T. Kohonen: *Self-Organized Formation of Topologically Correct Feature Maps*, Biological Cybernetics, 43(1), pp. 59-69, 1982, https://doi.org/10.1007/BF00337288

[16] J. E. Amaro, R. Navarro Pérez, E. Ruiz Arriola: *Error analysis of nuclear matrix elements*, Few-Body Systems, 2013, https://doi.org/10.1007/s00601-013-0756-4

[17] P. R. Bevington, D. K. Robinson: *Data Reduction and Error Analysis for the Physical Sciences*, McGraw-Hill, 2003.

[18] G. E. P. Box, N. R. Draper: *Empirical Model Building and Response Surfaces*, John Wiley & Sons, New York, 1987.

[19] S. Brandt: *Data Analysis. Statistical and Computational Methods for Scientists and Engineers*, Fourth Edition, Springer, New York, 2014.

[20] International Organization for Standardization (ISO): *Guide to the expression of Uncertainty in Measurement (GUM) – Supplement 1: Numerical methods for the propagation of distributions*, Report, 2004.

[21] Joint Committee for Guides in Metrology (JCGM): *JCGM 100:2008 Evaluation of measurement data – Guide to the expression of Uncertainty in Measurement*, Report, 2008.

[22] R. Z. Morawski, A. Miękina: *Monte-Carlo evaluation of measurement uncertainty using a new generator of pseudo-random numbers*, PAK vol. 59, nr 5, 2013.

[23] A. Saltelli, S. Funtowicz: *When all models are wrong,* Issues in Science and Technology,:79, 2013.

[24] J. R. Taylor: *An Introduction to Error Analysis*, Oxford University Press, 1982.

[25] B. A. Wichmann, I. D. Hill: *Generating good pseudo-random numbers*, Computational Statistics and Data Analysis, 51, pp. 1614-1622, 2006, https://doi.org/10.1016/j.csda.2006.05.019

[26] B. M. Adams, W. J. Bohnhoff, K. R. Dalbey, J. P. Eddy, M. S. Eldred, D. M. Gay, K. Haskell, P. D. Hough, L. P. Swiler: *DAKOTA, A Multilevel Parallel Object-Oriented Framework for Design Optimization, Parameter Estimation, Uncertainty Quantification and Sensitivity Analysis: Version 5.0 User's Manual*, Sandia Technical Report, SAND2010-2183, 2009.

[27] C. B. Moler: *Numerical computing with MATLAB*, SIAM, Philadelphia, 2004, https://doi.org/10.1137/1.9780898717952

[28] T. Williams, C. Kelley: *Gnuplot 5.3. An Interactive Plotting Program*, 2018, in: http://www.gnuplot.info/docs_5.5/gnuplot.pdf, access: August 2022.

[29] D. A. Piętak: *Statistical distribution of the genetic algorithm sampling with Schwefel's F7 objective function*, Proc. IEEE International Conference on Signals and Electronic Systems (ICSES), Gliwice, Poland, 2010.

[30] D. A. Piętak, J. Wojciechowski, P. J. Napiorkowski: *A Front-Line algorithm for error estimation in datasets with nonuniform sampling distribution*, Proc. 20th European Conference on Circuit Theory and Design (ECCTD), 6043319, pp. 210-213, Linköping, Sweden, 2011.

[31] D. Cline, T. Czosnyka, A. B. Hayes, P. J. Napiorkowski, N. Warr, C. Y. Wu: *GOSIA User Manual for Simulation and Analysis of Coulomb Excitation Experiments*, http://www.pas.rochester.edu/~cline/Gosia/Gosia_Manual_20120510.pdf, 2012.