METROLOGY AND MEASUREMENT SYSTEMS
Index 330930, ISSN 0860-8229
www.metrology.pg.gda.pl

MULTI-OBJECTIVE OPTIMIZATION PROBLEM IN THE OptD-MULTI METHOD

Wioleta Błaszczak-Bąk[1], Anna Sobieraj-Żłobińska[2], Michał Kowalik[3]

1) *University of Warmia and Mazury, Faculty of Geodesy, Geospatial and Civil Engineering, M. Oczapowskiego 1/25, 10-719 Olsztyn, Poland* (✉ *wioleta.blaszczak@uwm.edu.pl*, +48 89 523 3305)
2) *Gdańsk University of Technology, Faculty of Civil and Environmental Engineering, G. Narutowicza 11/12, 80-233 Gdańsk, Poland* (*annsobie@pg.gda.pl*)
3) *Programmer, Herzogstandstr 142, 85435 Erding, Germany* (*mkowalik@kleiberweb.de*)

Abstract

New measurement technologies, *e.g. Light Detection And Ranging* (LiDAR), generate very large datasets. In many cases, it is reasonable to reduce the number of measuring points, but in such a way that the datasets after reduction satisfy specific optimization criteria. For this purpose the *Optimum Dataset* (OptD) method proposed in [1] and [2] can be applied. The OptD method with the use of several optimization criteria is called OptD-multi and it gives several acceptable solutions. The paper presents methods of selecting one best solution based on the assumptions of two selected numerical optimization methods: the weighted sum method and the ε-constraint method. The research was carried out on two measurement datasets from *Airborne Laser Scanning* (ALS) and *Mobile Laser Scanning* (MLS). The analysis have shown that it is possible to use numerical optimization methods (often used in construction) to obtain the LiDAR data. Both methods gave different results, they are determined by initially adopted assumptions and – in relation to early made findings, these results can be used instead of the original dataset for various studies.

Keywords: reduction, OptD method, optimization, LiDAR.

1. Introduction

Many new data acquisition technologies, *e.g.* LiDAR (*Light Detection and Ranging*), collect large amounts of measurement datasets in a relatively short time. These datasets can be used to develop *e.g. Digital Terrain Model* (DTM), isoline maps, 3D visualizations, architectural modelling. For the DTM generation and 3D visualizations, such an even coverage is not optimal and is characterized by a large data redundancy. The uneven distribution of measurement points is the most advantageous for proper casting of terrain forms: more points in the area of occurrence of small, distinct morphological forms and less – in the area of large, "smooth" field forms. Therefore, LiDAR data are characterized by an excess density in the area with an uncomplicated terrain. Therefore, the reduction of LiDAR data should take into account the local complexity of terrain and should be based on spatial data analysis.

For decreasing the number of points in a dataset different approaches can be used. The first group contains methods based on a regular grid called generation grid, presented, for example,

Article history: received on Oct. 09, 2018; accepted on Dec. 23, 2018; available online on Jun. 28, 2019; DOI: 10.24425/mms.2019.128353.

in [3] The authors defined a new metric named *grid oversampling factor* (GOF) that estimates the local data oversampling appearing during the projection of generic satellite images onto a regular raster grid. Based on the common map projections, we defined sets of spatial grids optimized to minimise the data oversampling. In another work [4] also referring to this subject, the author describes generation of a DTM by applying a combination of interpolation methods. The criterion of method selection takes into account the dispersion of measurement points around a *grid* node. This solution enables successive complementing the resultant dataset of the computed dataset with values determined with a specific error and as a result improves the accuracy of the generated model.

The second group contains the methods of data reduction. The *grid* structure generates new coordinates, while the reduction enables to preserve raw data in the reduced dataset. This group of methods includes the *Optimum Dataset* (OptD) method presented in [1, 2].

The third group of methods refers only to data reduction in the case of DTM, in particular to DTM generalization. This problem was presented, among others, in [5–8].

In this work, the authors deal with the reduction of LiDAR data using the OptD method. It is a method of optimization of measurement datasets that contain spatial coordinates. It can be used in the OptD-single variant, when there is one optimization criterion, or in the OptD-multi one when there are more criteria. The OptD-single method was tested, among others, in [1, 9], while the OptD-multi method in [2]. In the case of processing by means of the OptD-single method, one solution is obtained, while the OptD-multi gives as a result more than one solution. Therefore, the next step is to choose the best solution among the found Pareto-optimal solutions. The decision-making stage is based on the pre-defined preferences and can also be performed before and during optimization. Choosing one solution is very important, therefore this paper focuses on this problem. The best solution from among the results satisfying the assumed optimization criteria can be selected, among others, using the following methods:

– weighted sum method [10, 11] – it is the best known and simplest multi-criteria decision-making method for evaluating the number of alternatives in terms of the number of decision criteria;
– ε-constraint method [12] – it consists in selecting one objective function, on the basis of which the optimization is carried out, and then, in an interval defined by the user, the optimization of the remaining criteria is continued;
– weighted metric methods [13–15] – instead of using a weighted sum of the objectives, other ways of combining multiple objectives can be considered;
– Benson's method [16] – finds the efficient extreme points in the outcome set. The primary concept in Benson's algorithm is to evaluate the upper image of the vector optimization problem by cutting planes.

The paper presents the results of study on the LiDAR dataset reduction by means of the OptD method. The result of the method is a dataset of permissible solutions from which one solution should be chosen. Not all methods of numerical optimization in the scope of the general multi-objective optimization are used in the geodetic data processing. In this study two methods were used: the weighted sum method and the ε-constraint method.

The essence of research is not the effect of reduction by the OptD method, therefore it is not compared with other reducing methods. The work focused on choosing the optimal solution from a set of acceptable solutions. It is important that the OptD-multi method automatically gives only one solution.

2. Numerical optimization methods

The general *multi-objective optimization* (MOO) problem is posed as follows:

$$\begin{gathered} \textit{Minimize}\ \mathbf{F}(\mathbf{x}) = [F_1(\mathbf{x}),\ F_2(\mathbf{x}),\ \ldots,\ F_k(\mathbf{x})]^T, \\ \textit{subject to}:\ g_j(\mathbf{x}) \le 0; \quad j = 1,\ 2,\ \ldots,\ m, \end{gathered} \tag{1}$$

where k is the number of objective functions and m is the number of inequality constrains.

Typically, there are infinitely many Pareto-optimal solutions for a multi-objective problem. Thus, it is often necessary to incorporate user preferences for various objectives in order to determine a single suitable solution. With *a posteriori articulation of preferences*, users can select manually a single solution from a representation of the Pareto-optimal dataset.

Alternatively, with methods that incorporate *a priori articulation of preferences*, the user indicates preferences before running the optimization algorithm and subsequently enables the algorithm to determine a single solution that presumably reflects such preferences.

How to decide which one to take and what the method can be used?

The possible approaches are: (1) the *weighted sum method* (WSM); (2) the *ε-constraint method* (CM).

Using the *weighted sum method* (WSM) to solve the problem in (1) entails selecting scalar weights w_i and minimizing the following composite objective function:

$$U = \sum_{i=1}^{k} w_i F_i(\mathbf{x}). \tag{2}$$

If all of the weights are positive, as assumed in this study, then minimizing (2) provides a sufficient condition for Pareto-optimality, which means that the minimum of (2) is always Pareto-optimal [17, 18]. Although in some literature there is indicated that $\sum_{i=1}^{k} w_i = 1$ and $w \ge 0$ if any one of weights is zero, there is a potential for solution to be only weakly Pareto-optimal [19].

The relations between the adopted weights and the objective function need to be always determined. A *preference function* is an abstract function (of points in the criterion space) in the mind of the decision-maker, which perfectly incorporates the user's preferences. Most MOO methods that involve minimizing a single aggregated objective function, attempt to approximate the preference function with some mathematical representation, called a *utility function*. The gradients of the preference function $P[\mathbf{F}(\mathbf{x})]$ and the utility function in (2) are given respectively as follows [19]:

$$\nabla_{\mathbf{x}} P[\mathbf{F}(\mathbf{x})] = \sum_{i=1}^{k} \frac{\partial P}{\partial F_i} \nabla_{\mathbf{x}} F_i(\mathbf{x}), \tag{3}$$

$$\nabla_{\mathbf{x}} U = \sum_{i=1}^{k} w_i \nabla_{\mathbf{x}} F_i(\mathbf{x}). \tag{4}$$

Each component of the gradient $\nabla_{\mathbf{x}} P$ qualitatively represents how the decision-maker's satisfaction changes with a change in the design point and a consequent change in function values. Comparing (3) and (4) suggests that if the weights are selected properly, then the utility function can have a gradient that is parallel to the gradient of the preference function.

The above relationship indicates that w_i represents $\dfrac{\partial P}{\partial F_i}$. Conceptually, $\dfrac{\partial P}{\partial F_i}$ is the approximate change in the preference function value (change in the decision-maker's satisfaction) that results from a change in the objective function value for F_i. However, it only makes sense to consider the importance of an objective or change in the preference function value, in relative terms. Thus, the weight value is significant relative to the values of others weights, the independent absolute magnitude of a weight is irrelevant in terms of preference (Marler and Arora, 2010).

Another approach is presented for the *$\mathcal{E}$-constraint method* (CM). The CM method was proposed by Haimes *et al*. (1971). It consists in selecting one objective function, on the basis of which the optimization is carried out, and then the ($\mathcal{E}$) optimization of the remaining criteria is continued in the range defined by the user. The method can be formulated as follows: Minimum/maximum $F_n(\mathbf{x})$:

$$\begin{aligned} F_n(\mathbf{x}) \le \varepsilon_m \quad &\text{where} \quad m = 1, 2, \dots, M \quad \text{and} \quad m \neq n, \\ g_j(\mathbf{x}) \ge 0 \quad &\text{where} \quad j = 1, 2, \dots, J, \\ h_k(\mathbf{x}) = 0 \quad &\text{where} \quad k = 1, 2, \dots, K, \\ x_i^{(L)} \le x_i \le x_i^{(U)} \quad &\text{where} \quad i = 1, 2, \dots n, \end{aligned} \tag{5}$$

where:

– parameter ε_m represents the upper constraint for an objective function F_n, for which the optimization was carried out;
– n – a selected objective function;
– m – the next constraint;
– M – the number of all constraints;
– $g_j(\mathbf{x})$, $h_k(\mathbf{x})$ – constraint functions, where $j$and $k$denote selected functions, J and K denote the number of all functions;
– x_i – a selected solution;
– $x_i^{(L)}$, $x_i^{(U)}$ – the lower and upper bounds (constraints).

An advantage of this method is finding different Pareto-optimal solutions, using different values of the ε parameter. In comparison with WSM, there is a possibility of finding the optimal solution belonging to the Pareto dataset of optimal solutions when the space of the problem is either convex or concave. However, a disadvantage of that solution is a significant dependence of the result on the selected parameter ε and the original optimization function. In some cases, a wrong choice of parameters in CM may not return any solution, or give the entire searched field as a solution. However, the most important problem of the CM method is the fact that a simple one-criterion problem is solved on the basis of only one parameter (after eliminating solutions that do not satisfy the criterion ε).

3. Tests

Algorithms of the OptD-multi method were implemented in Java v.9 programming language. The application was tested with both Oracle and OpenJDK runtime environment. A fragment of the program code with main steps of the method is presented below:

```
def main(argv):
    parser = argparse.ArgumentParser(description='Reduce LiDAR data set with OptD-Multi method')
    parser.add_argument('–input', dest='file',
                help='Input file')
    parser.add_argument('–action', dest='action',
                help='One of: reduce, stats')
```

```
    parser.add_argument('–criteria', dest='criteria',
                help='Yaml file containig optimalisation criteria settings')
    parser.add_argument('–output', dest='out_file',
                help='Output file. No file will be written unless specified')
    parser.add_argument('–save-plot', dest='plot_filename',
                help='Save result in plot.')
    Wrapper(validate_args(parser)).execute()
```

The optimization criteria are determined in the file config.yaml:

```
# parse yaml file here
    criteria = { }
    with open(args.criteria) as f:
        criteria = yaml.safe_load(f)
    opts = { **opts, **criteria}
  return opts
```

As the optimization criteria in our tests two parameters were assumed: 1) *standard deviation* (SD) in the dataset after reduction and 2) the percentage of points after reduction in the original point cloud (p).

In the OptD-multi method the Douglas-Peucker [20] algorithm was used:

```
## Douglas-Peucker algorithm implementation:
    def dp(points, tolearnce):
        if len(points) <= 2:
            return points
        # obtain max distance point:
        max_distance = 0
        max_point_index = None
        for i in range(1, len(points) - 1):
            if self.distance(points[0], points[-1], points[i]) >max_distance:
                max_distance = self.distance(points[0], points[-1], points[i])
                max_point_index = i
        if max_distance >tolearnce:
            left = dp(points[0:max_point_index + 1], tolearnce)
            right = dp(points[max_point_index:], tolearnce)
            return left[0:-1] + right
        else:
            return [points[0], points[-1]]
            return sorted(
        dp(points, tolerance),
        # list(set(dp(points, tolerance))),
        key=lambda x: x.id)
```

In the Douglas-Peucker algorithm, the tolerance parameter is very important. The degree of reduction depends mainly on tolerance.

3.1. Example 1

A point cloud from airborne laser scanning was provided by Vimap from Olsztyn. The measurements were taken on July 6, 2017 in Sweden, with a RIEGEL VUX1-UAV laser scanner at an altitude of around 100 m. The fragment of point cloud used in the work contains 651142 points (dataset Ω) and is presented in Fig. 1.

a)

b)

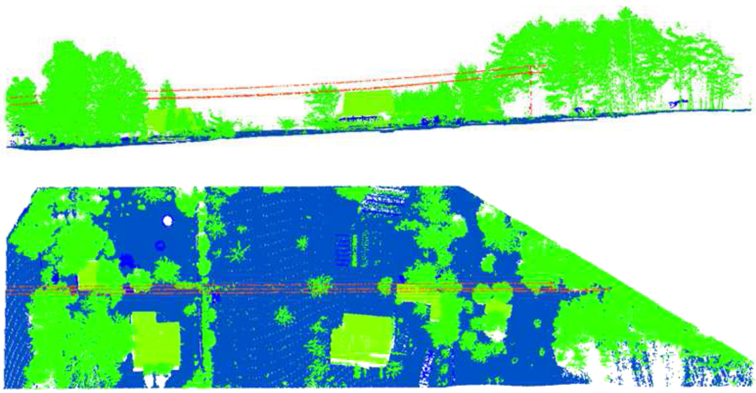


Fig. 1. ALS point cloud provided by Vimap: a) a side view; b) a top view.

The characteristics of Ω are presented in Table 1.

Table 1. Characteristics of Ω.

Parameter	**value**
Number of points	651 142
Z_{max} [m]	107.620
Z_{min} [m]	84.980
Z_{mean} [m]	89.370
SD [m]	5.569
$D_{average}$ [m]	0.058
D_{max} [m]	0.062

where: Z_{max} – the maximum height in ALS dataset; Z_{min} – the minimum height in ALS dataset, Z_{mean} – the mean height in ALS dataset; SD – standard deviation; $D_{average}$ – the average absolute distance between closest points; D_{max} – the maximum absolute distance between closest points.

In the OptD-multi method the following criteria were adopted:

– the differences between SD for the original data and SD for the data obtained after applying the OptD method, SD ≤ 0.400 m;
– $59\% < p < 62\%$.

To satisfy the optimization criteria, in the OptD-multi method four different values of strip widths (s) and tolerance (t) in the Douglas-Peucker algorithm were adopted:

– t = 0.077 m, s = 0.175 m for 59% of original points in 11 iterations;
– t = 0.061 m, s = 0.172 m for 61% of original points in 10 iterations;
– t = 0.066 m, s = 0.171 m for 62% of original points in 11 iterations.

The initial values of s and t are determined on the basis of the minimum distance between points in the dataset; subsequent values are changed in iterations. These two parameters determine the degree of reduction. For the initial values of s, the values of parameter t are checked in succession. It may turn out that at this stage we will find a set that satisfies the optimization criteria. If not, the width of the strip is changed and different tolerance ranges are tested again.

The parameter s controls the number of points in the X0Y plane, while the t parameter controls the number of points removed in the X0Z plane. This approach enables a spatial analysis of the set [1, 2].

As a result of processing by means of the OptD-multi method four datasets that satisfy the optimization criteria were obtained: Ω_1, Ω_2, Ω_3, Ω_4.

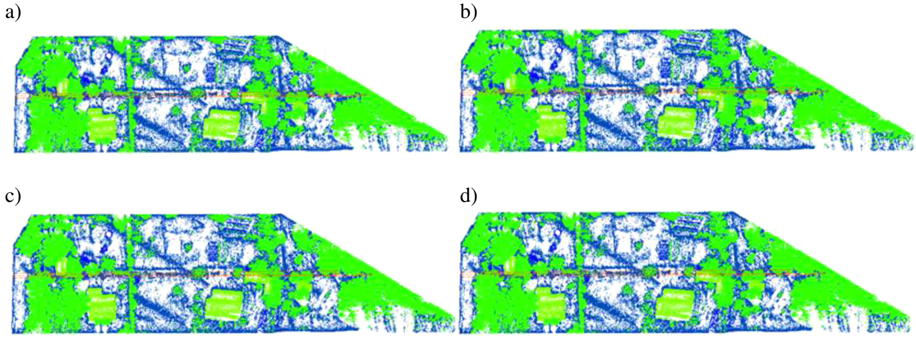


Fig. 2. ALS point cloud after applying the OptD-multi method: a) Ω_1 – 59% of original points; b) Ω_2 – 60% of original points; c) Ω_3 – 61% of original points; d) Ω_4 – 62% of original points.

The datasets obtained after reduction with the OptD-multi method are characterized in Table 2.

Table 2. Characteristics of datasets obtained after reduction with the OptD-multi method.

Parameter	Ω_1	Ω_2	Ω_3	Ω_4
Z_{min} [m]	84.980	84.980	84.980	84.980
Z_{max} [m]	107.620	107.620	107.620	107.620
Z_{mean} [m]	91.067	91.070	90.989	90.943
Number of points	387 708	388 768	396 997	402 326
$D_{average}$ [m]	0.097	0.098	0.097	0.099
D_{max} [m]	0.104	0.106	0.104	0.106
SD [m]	5.956	5.950	5.946	5.947
ABS (SD Ω – SD $\Omega_{i=1,2,3,4}$) [m]	0.387	0.381	0.377	0.378

As it is seen in Table 2, for each of the reduced datasets Ω_1, Ω_2, Ω_3, Ω_4 the Z_{max} and Z_{min} values are the same and equal to the values presented in Table 1 for the original dataset. It results from the way of working of the algorithm OptD – it preserves extreme values of the examined dataset. Due to the changes in the number of points after reduction Z_{mean} adopts various values.

The datasets obtained in the criteria space are presented in Fig. 3.

Assuming that f_1 = SD, f_2 = the number of points, and thus f_1 = minimum and f_2 = maximum of the objective functions, it can be stated that in the given case solutions Ω_3 and Ω_4 dominate over solutions Ω_1 and Ω_2. In solution Ω_3 f_2 is worse than in solution Ω_4, while f_1 is better. However, in the analysed problem, solutions Ω_3 and Ω_4 do not dominate over each other, since each of the solutions is better due to one criterion, and worse regarding the other. So both of these solutions constitute a dataset of Pareto-optimal solutions. The choice of one solution will depend on the purpose of the study and the user's decision. Among the obtained datasets, the optimal result should be chosen, which best suits the purpose of the study.

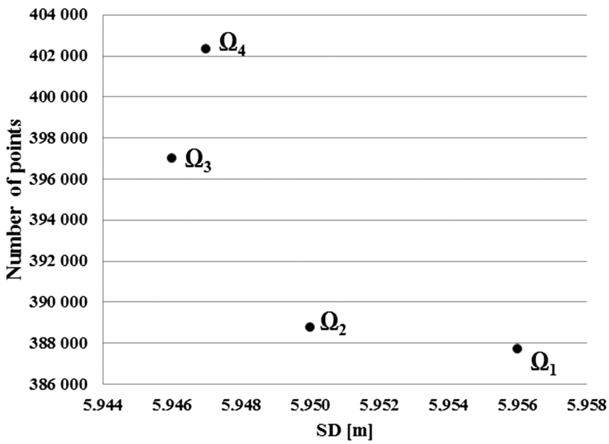


Fig. 3. Obtained datasets – solutions in the criteria space.

3.1.1. WSM for Example 1

In the analysed example, the following weights were adopted: for f_1 the weight was 0.4, while for f_2 the weight was 0.6. Thus, a better solution is the result with a higher number of points in the dataset after reduction.

Table 3. WSM calculation for Example 1.

	f_1	f_2	score
Ω_3	5.946	396 997	238200.578
Ω_4	5.947	402 326	241397.978
w	0.4	0.6	

The analysis shows that the best solution is Ω_4.

3.1.2. CM for Example 1

The CM method assumes that one objective function f_1 must be fulfilled. It was assumed that $f_1 = 5.950$. The set of solutions also includes solutions resulting from rounding the SD value. Consequently, the set criterion encompasses the following datasets: Ω_2, Ω_3, Ω_4. Next, the interval for f_2 was set in which the next criterion was to be satisfied (in our example the size of the dataset after reduction). Following this assumption $3000000 < f_2 < 4000000$ was accepted. This criterion was achieved by dataset Ω_3.

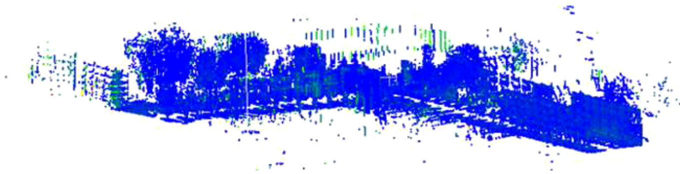
3.2. Example 2

The MLS data were acquired during the Fourth International Working Week on Multi-Sensor Integration for Assured Navigation, October 1 – October 8, 2017. The meeting took place at The Ohio State University (OSU), Department of Civil, Environmental and Geodetic Engineering (CEG) in the Satellite Positioning and Inertial Navigation (SPIN) Laboratory. Two types of Velodyne LiDAR were used for scanning: one Velodyne HDL-32 on the front top and eight Velodyne VLP-16 on the side and rear of the vehicle.

Metrol. Meas. Syst., Vol. 26 (2019), No. 2, pp. 253–266.

A measurement from one Velodyne LiDAR VLP-16 containing 1730748 points was selected for processing. The MLS measurement set (dataset Φ) is presented in Fig. 4.

a)



b)

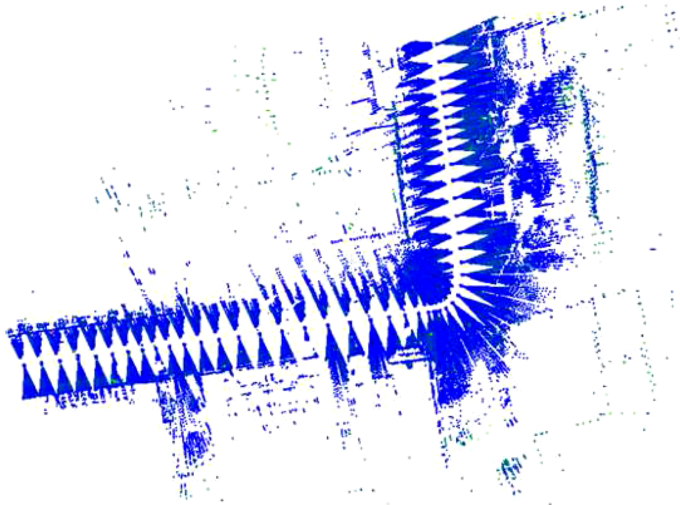


Fig. 4. MLS point cloud provided by SPIN Laboratory: a) a side view; b) a top view.

The characteristics of Φ are presented in Table 4.

Table 4. Characteristics of Φ.

Parameter	**value**
Number of points	1 730 748
Z_{max} [m]	87.712
Z_{min} [m]	−63.075
Z_{mean} [m]	2.959
$D_{average}$ [m]	0.425
D_{max} [m]	0.554
SD [m]	3.780

For the reduction the OptD-multi method was used. As the optimization criteria, similarly to Example 1, the percentage of points after reduction in the original *point cloud* (p) and *standard deviation* (SD) in the set after reduction were assumed as follows:

– the differences between SD for the original data and SD for the data obtained after applying the OptD method, SD ≤ 1.000 m;
– 59% < p < 62% (with interval 0.5%).

To satisfy the optimization criteria, in the OptD-multi method four different values of strip widths (s) and tolerance (t) in the Douglas-Peucker algorithm were adopted:

– t = 0.590 m, s = 0.342 m for 59% of original points in 13 iterations;
– t = 0.595 m, s = 0.350 m for 59.5% of original points in 12 iterations;
– t = 0.600 m, s = 0.342 m for 60% of original points in 11 iterations;
– t = 0.608 m, s = 0.342 m for 61% of original points in 11 iterations;
– t = 0.610 m, s = 0.345 m for 61.50% of original points in 12 iterations;
– t = 0.614 m, s = 0.350 m for 62% of original points in 11 iterations.

As a result of the OptD-multi processing four datasets that satisfied the optimization criteria were obtained: Φ_1, Φ_2, Φ_3, Φ_4, Φ_5, Φ_6.

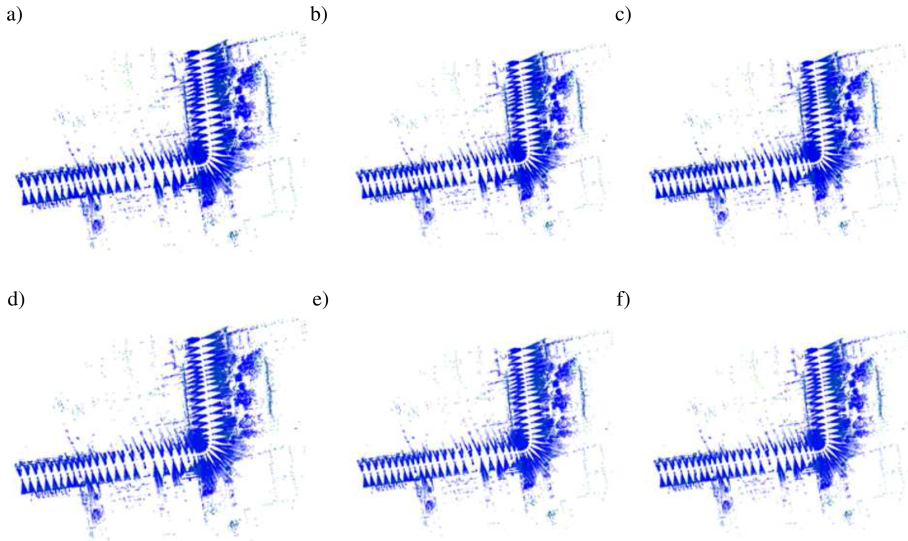


Fig. 5. MLS point clouds after applying the OptD-multi method: a) Φ_1 – 59% of original points; b) Φ_2 – 59.5% of original points; c) Φ_3 – 60% of original points; d) Φ_4 – 61% of original points; d) Φ_5 – 61.5% of original points; d) Φ_6 – 62% of original points.

The datasets obtained after reduction with the OptD-multi method are characterized in Table 5.

Table 5. Characteristics of datasets obtained after reduction with the OptD-multi method.

Parameter	Φ_1	Φ_2	Φ_3	Φ_4	Φ_5	Φ_6
Z_{min} [m]	87.712	87.712	87.712	87.712	87.712	87.712
Z_{max} [m]	−63.075	−63.075	−63.075	−63.075	−63.075	−63.075
Z_{mean} [m]	3.692	3.670	3.670	3.602	3.599	3.602
Number of points	1 021 143	1 029 798	1 038 453	1 055 760	1 064 410	1 073 060
$D_{average}$ [m]	0.720	0.720	0.721	0.720	0.722	0.724
D_{max} [m]	0.889	0.889	0.889	0.091	0.090	0.091
SD [m]	4.038	4.020	4.005	4.005	4.017	4.007
ABS (SD Φ – SD $\Phi_{i=1,2,3,\ldots,6}$) [m]	0.258	0.240	0.225	0.225	0.237	0.224

In this case it also can be observed, that the Z_{max} and Z_{min} values are the same for each reduced Φ_i dataset and equal to the values characteristic for the original MLS set (Table 4). As it was in the previous study, the Z_{mean} value is changing, depending on the adopted reduction criteria.

The obtained solutions are presented in the criteria space (Fig. 6).

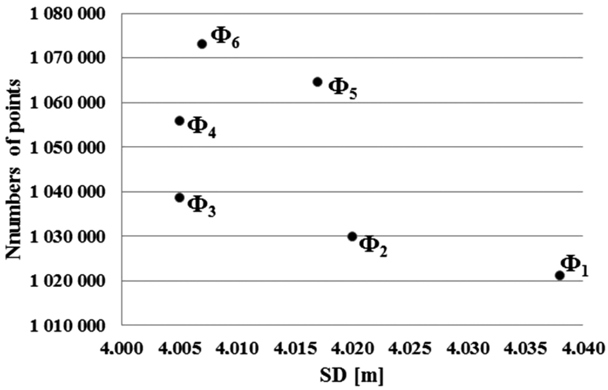


Fig. 6. Obtained datasets – solutions in the criteria space.

For these solutions, it was assumed that f_1 = SD, f_2 = the number of points, and then, as in Example 1, f_1 = minimum, f_2 = maximum of the objective functions. It can be concluded, that in the given case solutions Φ_3, Φ_4 and Φ_6 dominate over solutions Φ_1, Φ_2 and Φ_5. In solutions Φ_3 and Φ_4 f_1 is the same, while f_2 is better for Φ_4. A very similar solution gives Φ_6, where f_2 is the best, and f_1 is only 0.002 m higher than for Φ_3 and Φ_4. The datasets of Pareto-optimal solutions in Example 2 are Φ_4 and Φ_6.

3.2.1. WSM for Example 2

In Example 2, the same weights were assumed as in Example 1: for f_1 the weight was 0.4, and for f_2 the weight was 0.6.

Table 6. WSM calculation for Example 2.

	f_1	**f_2**	**score**
Φ_4	4.005	1 055 760	633457.602
Φ_6	4.007	1 073 060	643837.602
w	0.4	0.6	

The analysis shows that the best solution is Φ_6.

3.2.2. CM for Example 2

The CM method assumes that one objective function f_1 must be fulfilled (similarly to Example 1). It was assumed that f_1 = 4.005. The set of solutions also includes solutions resulting from rounding the SD value. The established criterion was satisfied by Φ_3 and Φ_4.

Next, an interval for f_2 was set in which the next criterion was to be satisfied (in our example, the size of the set after reduction). The assumption of $1030000 < f_2 < 1070000$ was accepted. This criterion is satisfied by Φ_3.

4. Discussion

Applied in the OptD-multi the weighted sum method and the ε-constraint method gave different results. From the point of view of the assumed optimization criteria, all obtained solutions can be accepted as optimal. However, to obtain one result as the final one, in the authors' opinion, one more parameter should be introduced to evaluate the sets after reduction, namely the processing time. The shorter it takes to obtain an optimal solution, the better. Therefore, Table 7 summarizes the processing times of the OptD-multi method spent for selected solutions in the presented examples. The results were obtained on Dell Precision Intel Core i5-2520M CPU@2.50GHz.

Table 7. Processing times.

Datasets after OptD-multi	
Example 1	
with WSM	with CM
Ω_4	Ω_3
Processing time [sec]	
32	**23**
Example 2	
with WSM	with CM
Φ_6	Φ_3
Processing time [sec]	
70	**62**

The times presented in Table 7 make it possible to clearly distinguish the sets obtained after reduction with the OptD method for the two presented examples. In both cases the OptD-multi reduction with using the CM method was shorter than with using the WSM method.

5. Conclusions

In this study the authors analysed the reduced datasets obtained after reduction performed by the OptD method. Two sets of data from ALS and MLS were used for analysis. The result of applying the OptD-multi method was four reduced ALS datasets, while in the case of the MLS data – six datasets. Two methods were used in the work: the weighted sum method and the ε-constraint method to select one solution from among the permissible ones.

General conclusions that can be formulated on the basis of the obtained results are as follows:

1. The OptD-multi method gives a set of Pareto-optimal solutions.
2. The obtained datasets satisfy the set optimization criteria.
3. The weighted sum method and the ε-constraint method can be used to analyse the data from ALS and MLS.
4. The applied numerical optimization methods give different results.

Specific conclusions can be presented as follows:

1. In both Examples, the same assumptions were adopted for both selected methods of numerical optimization.
2. In Example 1, using the weighted sum method, Ω_4 was considered as the best solution, whereas Ω_3 was the best in the ε-constraint method.

3. In Example 2, for the weighted sum method, Φ_6 was considered as the best solution, whereas in the $\mathcal{E}$-constraint method Φ_3 was the best solution.
4. In Example 1, the values of SD parameter for both selected solutions differ by 0.001 m, while the size of the set differs by 5 329 points.
5. In Example 2, the value of SD parameter for both selected solutions differ by 0.002 m, while the size of the set differs by 34 607 points.
6. The OptD-multi reduction using the CM method lasted shorter than that using the WSM method, therefore the authors consider the CM-based solution the better one.

References

[1] Błaszczak-Bąk, W. (2016). New Optimum Dataset method in LiDAR processing. *Acta Geodynamica et Geomaterialia*, 13/4(184), 379–386.

[2] Błaszczak-Bąk, W., Sobieraj-Żłobińska, A., Kowalik, M. (2017). The OptD-multi method in LiDAR processing. *Measurement Science and Technology*, 8(7), 075009.

[3] Bauer-Marschallinger, B., Sabel, D., Wagner, W. (2014). Optimisation of global grids for high-resolution remote sensing data. *Computers & Geosciences*, 72, 84–93.

[4] Gościewski, D. (2014). Reduction of deformations of the digital terrain model by merging interpolation algorithms. *Computers & Geosciences*, 64, 61–71.

[5] Bakuła, K. (2011). Comparison of six approaches in DTM reduction for flood risk determination. *Challenges of Modern Technology, WUT PhD Students Board*, 2(4), 31–36.

[6] Bakuła, K. (2014). *The role of the reduction of elevation data obtained from airborne laser scanning in the process of flood hazard map creation*. PhD Thesis. Warsaw University of Technology.

[7] Weiber, R. (1992). Model and experimente for adaptive computer-assisted terrain generalization. *Cartograph and Geographic Information System*, 19(2).

[8] Chen, Y. (2012). *High performance computing for massive LiDAR data processing with optimized GPU parallel programming*. University of Texas at Dallas. Book. Graduate Program in Geospatial Information Science.

[9] Błaszczak-Bąk, W., Sobieraj-Żłobińska, A., Poniewiera, M., Kowalik, M. (2018). *Reduction of measurement data before DTM generation vs. DTM generalization*. Survey Review, 1–9.

[10] Fishburn, P.C. (1967). *Additive Utilities with Incomplete Product Set: Applications to Priorities and Assignments*. Operations Research Society of America (ORSA), Baltimore, MD, U.S.A.

[11] Triantaphyllou, E. (2000). *Multi-Criteria Decision Making: A Comparative Study*. Dordrecht, The Netherlands: Springer, 320.

[12] Haimes, Y.Y., Lasdon, L.S., Wismer, D.A. (1971). On a bicriterion formulation of the problems of integrated system identification and system optimization. *IEEE transactions on Systems, Man, and Cybernetics*, 1(3), 296–297.

[13] Hwang, C.L., Masud, A.S., *et al*. (1979). *Multiple Objective Decision Making Methods and Applications*. Springer, Berlin.

[14] Asgharpour, M.J. (1998). *Multiple Criteria Decision Making*. Tehran University Press, Tehran.

[15] Deb, K. (2001). *Multi-objective Optimization Using Evolutionary Algorithms*. Wiley, New York, NY.

[16] Benson, H.P. (1998). An Outer Approximation Algorithm for Generating All Efficient Extreme Points in the Outcome Set of a Multiple Objective Linear Programming Problem. *Journal of Global Optimization*, 13(1), 1–24.

[17] Goicoechea, A., Hansen, D.R., Duckstein, L. (1982). *Multiobjective Decision Analysis with Engineering and Business Applications*. J. Wiley& Sons, New York, 519.

[18] Zadeh, A. (1963). Optimality and non-scalar-valued performance criteria. *IEEE Transactions on Automatic Control*, 8, 59–60.

[19] Marler, R.T., Arora J.S. (2010). The weighted sum method for multi-objective optimization: new insights. *Structural and Multidisciplinary Optimization*, 41(6), 853–862.

[20] Douglas, D.H., Peucker, T.K. (1973). Algorithms for the reduction of the number of points required to represent a digitized line or its caricature. *Canadian Cartographer*, 10(1), 112–122.