Visual analytics in the case of multicriteria optimization

T.P. Galkin[1,A], A.P. Nelubin[2,B], A.A. Galaev[3,A], D.D. Popov[4,A],
V.V. Pilyugin[5,A], S.Yu. Misyurin[6,A,B]

[A] NRNU «MEPhI», Moscow, Russia
[B] IMASH RAN, Moscow, Russia

[1] ORCID: 0000-0003-2859-6275, z@wqc.me
[2] ORCID: 0000-0002-7064-3103, nelubin_andrey@inbox.ru
[3] ORCID: 0000-0003-3539-3206, aalexgalaev@gmail.com
[4] ORCID: 0000-0002-3333-749X, DDPopov@mephi.ru
[5] ORCID: 0000-0001-8648-1690, VVPilyugin@mephi.ru
[6] ORCID: 0000-0003-1020-0527, symisyurin@mephi.ru

Abstract

The paper considers the issue of multidimensional data analysis in the field of decision making. The authors made the formal problem statement in the general case and gave the example of the analysis for an optimization of a two-mass dynamic model. The visualization method was applied as the data analysis method. The method was shortly described. In addition, a detailed description of a visualization pipeline which was used for problem solving was given in the article. The authors formulated three types of judgments that can be made by an analyst and created the algorithm for the problem solving, which is based on the visualization method. The software tool called "Visual Analytics Tool" was developed. This article contains the description of this program. It is based on the visualization pipeline and allows to analyze multidimensional data using the algorithm. The results of data analysis, which was conducted with the tool, is given. They contain the description of: the sensitivity of the optimization task's solution, the clustering process of the alternatives, which are close to an ideal, and the correlation between functional limits and objective functions values.

The study was partially supported by RFBR, research project No.16-29-04401.

Keywords: multidimensional data, data analysis, visualization method, multicriteria optimization.

Introduction

Modern engineering design and development mostly leads to the process of choosing the best suitable configuration among all possible variants of the required device. A number of variable parameters of the device or mechanism are defined with corresponding limitations for each parameter. Then, criteria of the best device selection are defined. Thus, a multicriteria optimization task is stated.

Multicriteria optimization is a process of simultaneous optimization of two or more conflicting objective functions C on given domain of definition T. Finding the optimal solution means finding one or multiple set of parameters P in their limitations so the corresponding set of criteria is acceptable for the problem originator.

Solution algorithm of multicriteria optimization task with one objective function is presented in [1]. Several existent approaches of solving the multicriteria optimization tasks are considered in [2] and [3], which represent the evolution of American and French schools of visual analysis. Appearance of multiple sets of analyzable data caused by solving the task is similar in those approaches. The difference is, on the other hand, in the process of the obtained data analysis: the preconditions and the results are not the same.

All mentioned papers use visualization heavily. Studied in those works programs generate images, which help analyst in the process of problems solving. Those images can be used either in illustrative purposes or, being in an intermediate state of research, to clarify the following steps in the data analysis process. By that, visualization naturally integrated into the multicriteria task solving concept.

Visual analytics paradigm, stated by James Thomas, calls visual analysis a way of solving the data analysis tasks using the supporting interactive interface. It is to join human imagination and enormous processing power of modern computers. While a highly computational modeling job is being processed by a computer, the analyst always has feedback through an image on a computer display. The implementation of visualization method is given in [4]. It consists of visualization pipeline, which is an interactive and iterative image rendering algorithm, and an analysis process of the resulting image.

This article describes the development of visual multidimensional data analysis tool. The concluding parts of the work contain detailed capabilities of designed tool and examples of its applications to multicriteria optimization tasks solving process. This tool also allows applying visual analysis to given data. Then it will become a part of a software package, designed to solve multicriteria optimization tasks. Development of this package is a big project of Institute of Cyber Intelligence Systems, NRNU MEPhI. Visual analysis tool is being developed by "Scientific Visualization" laboratory, NRNU MEPhI.

1. Problem definition

Multicriteria optimization task solving includes multidimensional data analysis. General view of this task is as follows:

Given:
$P = \{p_1, p_2, \ldots, p_q\}$ – *q real variables on given domain of definition.*
$C = \{c_1, c_2, \ldots, c_m\}$ – *m procedurally stated objective functions:*
$c_z = c_z(p_1, p_2, \ldots, p_n)$, *where z* $\in$ *[1 .. m].*

$G = \{g_1, g_2, \ldots, g_k\}$ – *k procedurally stated functions, setting additional limitations on domain of definition T.*

$$T = \begin{cases} P_{i-} < p_i < P_{i+} \\ g_j(p_1, p_2, \ldots, p_i) \gtreqless G_j \end{cases},$$

where P_{i-}, P_{i+}, G_j – *real numbers, i* $\in$ *[1 .. n], j* $\in$ *[1 .. k].*

Formulate:
Judgment about functions C joint behavior on domain of definition T.

This article is about one particular multicriteria optimization task, expanded in [5]. This task comes to data analysis task with $q = 5$, $k = 3$, $m = 4$ and is stated below:

Twomass dynamic model, see fig. 1, contains: masses M_1 and M_2, rigidity K_1 and K_2, shock absorption C. Thus, $P = \{k_1, k_2, m_1, m_2, c\}$. The harmonic force $P \cdot \cos(\omega t)$ is applied to M_1, P=2000 N; w=30 (sec^{-1}).
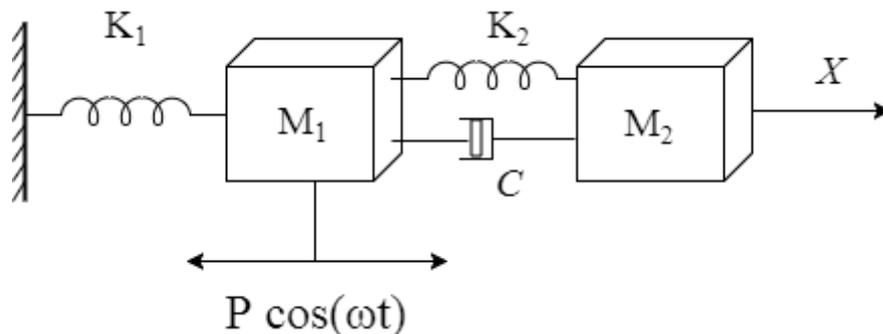

Figure 1. Dynamic model.

Movement equations are as follows:

$$M_1 \ddot{X}_1 + C(\dot{X}_1 - \dot{X}_2) + K_1 X_1 + K_2(X_1 - X_2) = P \cdot \cos(\omega t)$$
$$M_2 \ddot{X}_2 + C(\dot{X}_2 - \dot{X}_1) + K_2 X_2 + K_2(X_2 - X_1) = 0$$

Five variable parameters are:

$p_1 = K_1,\ p_2 = K_2,\ p_3 = M_1,\ p_4 = M_4,\ p_5 = C$

with variation boundaries, forming parallelepiped Π,

$$1.1\times10^6 \le p_1 \le 2.0\times10^6\ (\text{Н/м});$$
$$4.0\times10^4 \le p_2 \le 5.0\times10^4\ (\text{Н/м});$$
$$950 \le p_3 \le 1050\ (\text{кг});$$
$$30 \le p_4 \le 70\ (\text{кг});$$
$$80 \le p_5 \le 120\ (\text{Н×с/м}).$$

Three functional constraints (on mass sum and partial frequencies):

$$g_1(\alpha) = p_3 + p_4 \le 1100\ (\text{kg});$$
$$33.0 \le g_2(p) = \sqrt{p_1/p_3} \le 42.0\,(\text{sec}^{-1});$$
$$27.0 \le g_3(p) = \sqrt{p_2/p_4} \le 32.0\,(\text{sec}^{-1})$$

Superior limitation on $g_2(p)$ and $g_3(p)$ are flexible and can vary in some limits. Domain of definition T is formed by joining parallelepiped Π and functional constraints G.

Optimization was conducted by four quality criteria:

$C_1 = X_{1\P}$ (mm) – first mass oscillation amplitude, minimized,

$C_2 = M_1 + M_2$ (kg) – specific quantity of metal in the system, maximized,

$C_3 = X_{1\P}/X_{1st}$ and $C_4 = w/g_2$ (non-dimensional quantities) – dynamic characteristics of the system, where X_{1st} – static deviation of mass M_1 under force P action, both minimized.

Formulate:
Judgment about quality criteria joint behavior on the domain of definition.

2. Task solution method

The visualization method was chosen for solving the considering task. Main idea of visualization method is using the visualization tools bring together the source data and the corresponding static or dynamic graphic interpretation. Obtaining this graphic interpretation, one can make decisions about the source data. Therefore, the method the method is made out of two sequential solutions of tasks. They are a pure visualization and the task of image analysis and interpretation.
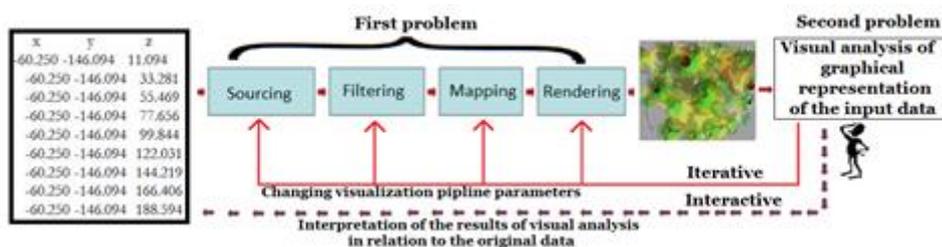

Figure 2. Data analysis task solving using the visualization method.

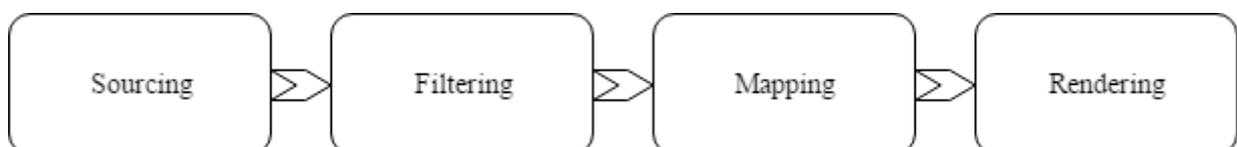Algorithm of obtaining graphic images from the analyzed data consists of the following steps:

Figure 3. Visualization pipeline.

Together those steps, performed one after another, are called visualization pipeline.

2.1. Visualization pipeline

Sourcing is the process of obtaining source data of the visualization pipeline. The source data can be obtained in a variety of ways, such as statistical processing or analytical function assignment.

Filtering is an operation of preprocessing the source data. Filtered data is used in the future steps of pipeline. For example, it could include interpolation and smoothing of the source data. This process is optional.

The spatial scene is built by the filtered data; this process includes describing the geometric model of the scene and its optical specifications. Geometric model of scene consists of geometric primitives and complex geometric objects.

Rendering is a process of generating graphic representations of the obtained scene. 3D rendering has a number of settings:
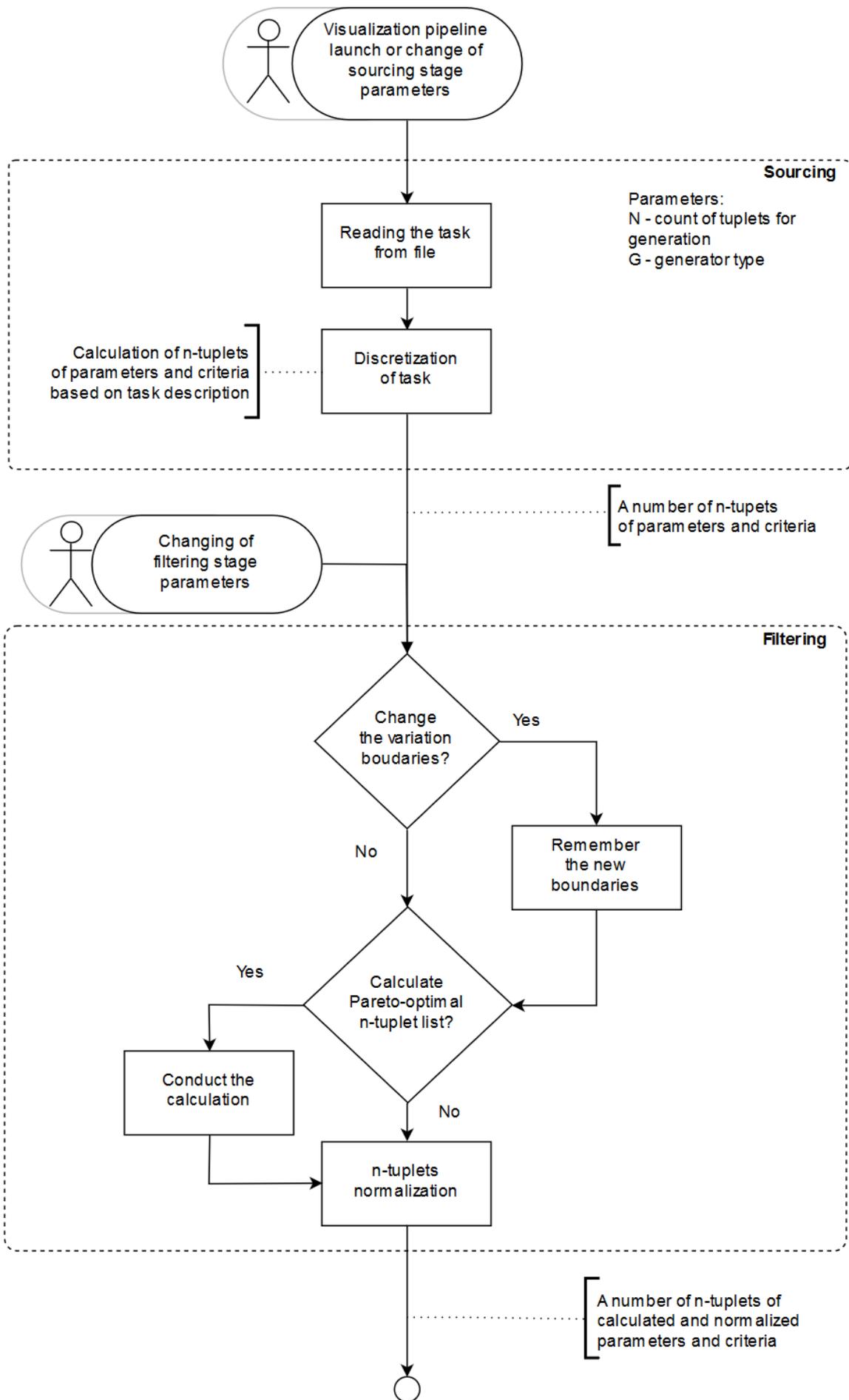
1. Camera
2. Light sources
3. Environment of spatial scene

2.2. Visual analysis

The visualization process cannot be formalized strictly. Some of the scene analysis problems, solved in the process of analysis by human, are: analysis of spatial object's shapes, analysis of their mutual arrangement, and analysis of their optical properties. When obtaining unsatisfactory results, the whole visualization pipeline or its parts are rerun. As the result, the analysis task becomes more complicated, iterative and interactive.

3. Algorithm design

The data analysis visualization method consists of two tasks, executed consequently. The first is data visualization and its visual analysis. Then, the results are interpreted towards the source data. The designed algorithm of visualization pipeline is given below.
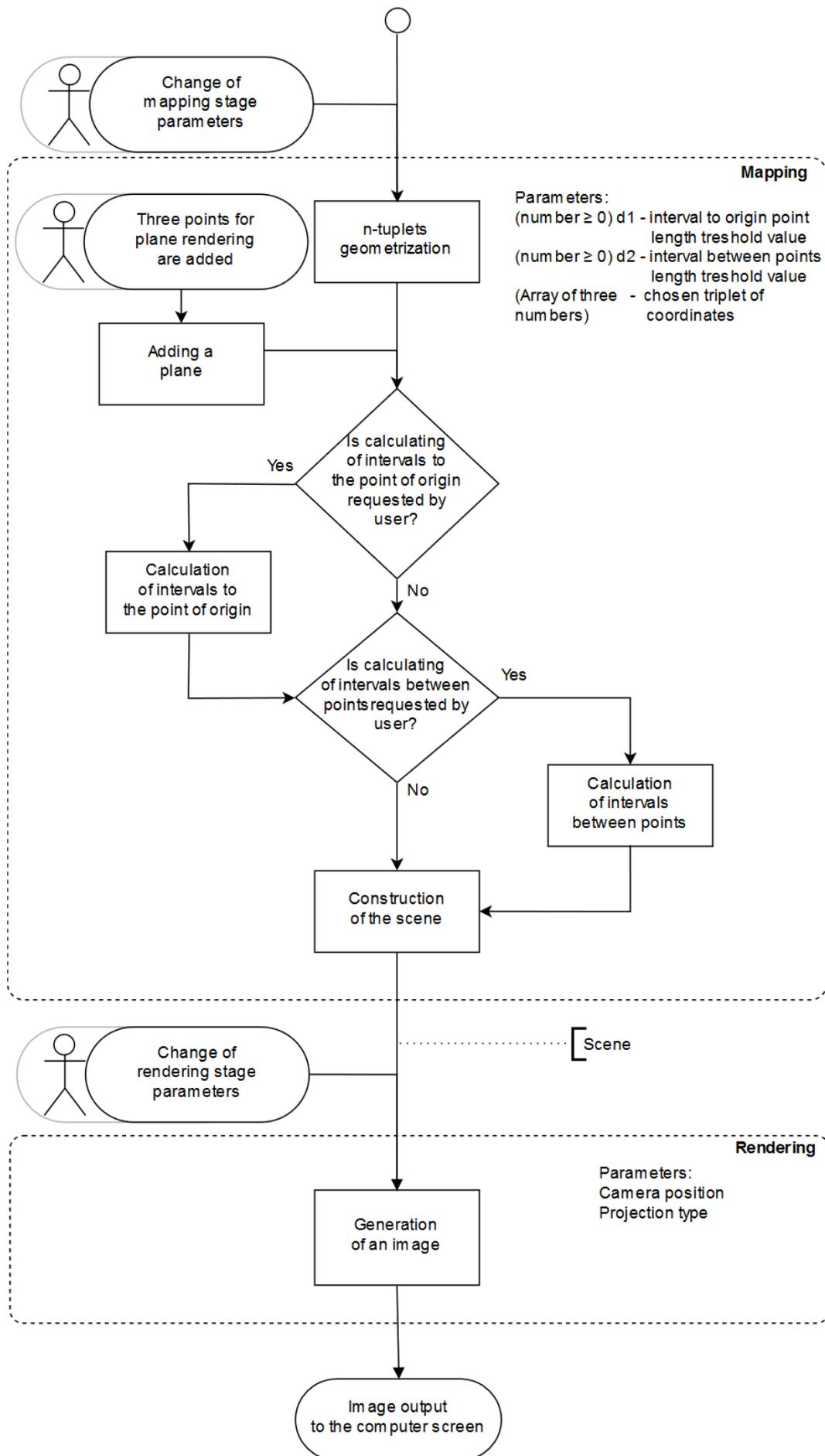
Visualization pipeline launch or change of sourcing stage parameters

Sourcing

Parameters:
N - count of tuplets for generation
G - generator type

Reading the task from file

Calculation of n-tuplets of parameters and criteria based on task description .... Discretization of task

A number of n-tupets of parameters and criteria

Changing of filtering stage parameters

Filtering

Change the variation boudaries?

Yes

No

Remember the new boundaries

Calculate Pareto-optimal n-tuplet list?

Yes

No

Conduct the calculation

n-tuplets normalization

A number of n-tuplets of calculated and normalized parameters and criteria

Change of mapping stage parameters

n-tuplets geometrization

Three points for plane rendering are added

Adding a plane

Parameters:
(number ≥ 0) d1 - interval to origin point length treshold value
(number ≥ 0) d2 - interval between points length treshold value
(Array of three - chosen triplet of numbers) coordinates

Mapping

Is calculating of intervals to the point of origin requested by user?

Yes

No

Calculation of intervals to the point of origin

Is calculating of intervals between points requested by user?

Yes

No

Calculation of intervals between points

Construction of the scene

Scene

Change of rendering stage parameters

Rendering

Parameters:
Camera position
Projection type

Generation of an image

Image output to the computer screen

Figure 4. Algorithm scheme.

The visualization pipeline has four steps: Sourcing, Filtering, Mapping and Rendering. All of them are iterative and interactive, which means it's possible to start the pipeline again from any step. Entry points are marked as a sketchy man icon. Thus, once the image has been obtained, one can restart the pipeline from the Mapping stage using other parameters.
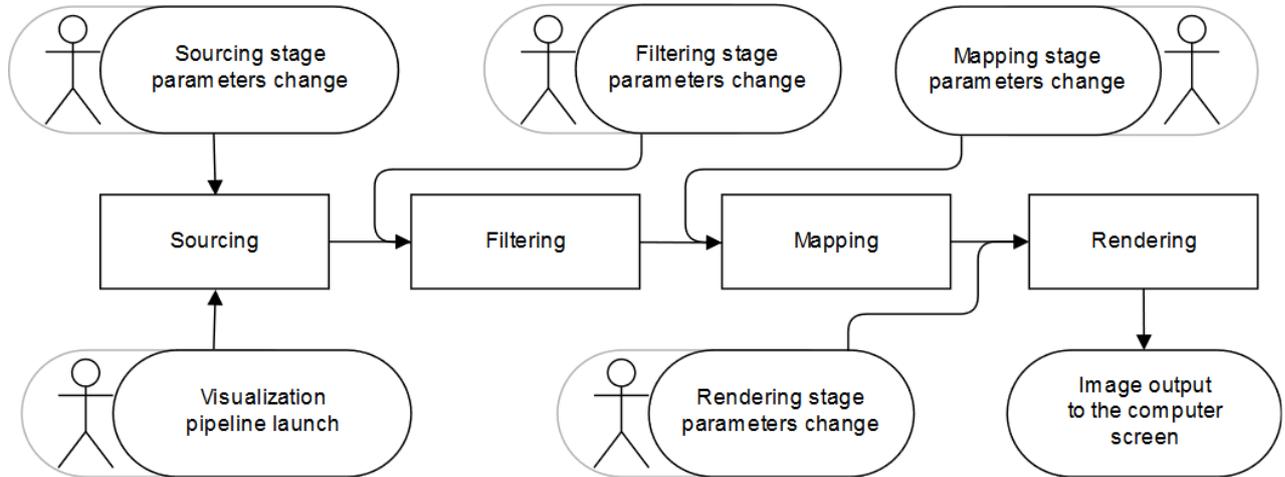

Figure 5. Visualization pipeline scheme.

3.1.1. Sourcing

Source data is obtained on this step. Sourcing can be controlled; one can interactively change the input data. He/she can:

- change the boundaries of the parallelepiped Π;
- choose the method of generation for values of parameters, there are two options: pseudo-random number sampling a uniform distribution and Sobol sequences;
- set the amount of data for computations.

The sourcing stage consists of two parts. First part is loading the problem from a .lua script file with a known structure. Once loaded, the procedurally stated calculation functions inside this file are discretized. It allows obtaining an n-tuplet, an ordered set of n real values of function with corresponding variables.

Digitalization

The user states N – a required number of n-tuplets, which are then generated. Thus, a sequence number is assigned to every n- tuplet, $t = \overline{1, N}$. As every n- tuplet is an ordered set of numbers, the variable value will be p_{it}, with corresponding objective function values c_{zt}, where $i = \overline{1, q}$ and $z = \overline{1, m}$. This way, the value of any variable or objective function is uniquely determined for every n- tuplet.

Two random generation method was implemented. The theory for the Sobol sequences algorithm was proposed by [7].

3.1.2. Filtering

Data, obtained on previous step, is preprocessed on this stage. One can change the domain of definition and start the set of Pareto-optimal objective function values to be calculated. The data is normalized. Output of this step contains a number of n-sets and normalized n-tuplet.

Filtering stage consists of three parts. In the first one G_j are changed, thus changing the domain of definition. The second one is calculation of the Pareto-optimal values (optional). The third one is data normalization.

Normalization

To get a geometric model, obtained data is linearly normalized, so the values of every variable and objective function are contained in [0, 100].

First, the largest and the lowest values of variables and objective functions are calculated from all n-tuplets. New ordered sets of n real numbers of function and variable values are obtained for each n-tuplet:

$\tilde{P} = \{\widetilde{p_1}, \widetilde{p_2}, \ldots, \widetilde{p_q}\}$, where the $\widetilde{p_{it}}$ for each $\widetilde{p_i}$ is calculated as follows:

$$\tilde{p}_{it} = \frac{p_{it} - \min\limits_{t=1,N} p_{it}}{\max\limits_{t=1,N} p_{it} - \min\limits_{t=1,N} p_{it}} * 100, i \in [1 \ldots q].$$

$\tilde{C} = \{\widetilde{c_1}, \widetilde{c_2}, \ldots, \widetilde{c_z}\}$, where the $\widetilde{c_{zt}}$ for each $\widetilde{c_z}$ is calculated as follows:

$$\tilde{c}_{zt} = \frac{c_{zt} - \min\limits_{t=1,N} c_{zt}}{\max\limits_{t=1,N} c_{zt} - \min\limits_{t=1,N} c_{zt}} * 100, z \in [1 \ldots m].$$

3.1.3. Mapping

Completing this stage implies data geometrization and building of a spatial scene. Ordered real n-tuplet are considered as points in n-dimensional space. 3-dimentional subspace of initial n-dimentional space is chosen, it will be used for creation of spatial scene. This approach was introduced in [8].

Two entry points are present: first is on the visualization pipeline initial launch or on control pass from Filtering stage; second is activated when user marks plane points. Right after getting control, geometrization of data in question is implemented. Sets of normalized parameters and criteria are presented as set of multidimensional spaces $\mathbb{P}$ and $\mathfrak{C}$. We name them parameter space and criteria space, respectfully.

Algorithm provides ability to enable function of adding intervals between points in $\mathbb{P}$ and $\mathfrak{C}$ spaces in pairs or to the point of origin, and adding planes.

The last block of this algorithm is building the scene.

Obtained on Filtering stage sets of normalized data are used to form 3-dimentional space $\{x, y, z\}$ as follows: triplet $\{a, b, c\}$ is given by $a, b, c \in [1 \ldots q+m]$; those numbers defines the space in the way, so $x = \begin{cases} \widetilde{p_a}, & a \leq q \\ \widetilde{c_{a-q}}, & a > q \end{cases}$; y and z are defined similarly.

The multidimensional points are projected on $\{x, y, z\}$ as follows: set of triplets (x_i, y_i, z_i) is given by $x_i = \begin{cases} \widetilde{p_{ia}}, & a \leq q \\ \widetilde{c_{ia-q}}, & a > q \end{cases}$; y_i and z_i are defined similarly, $i \in [1, N]$.

Using $\tilde{\mathbb{P}}$ and $\tilde{\mathbb{C}}$, obtained on Filtering stage, describe the following functions:

- $\rho(i,j)$ – function of distance calculation between points i and j,
- $\rho(0,i)$ – function of distance calculation between a point and the point of origin.

Those functions take values as follows:

- All elements of $\{x, y, z\}$ are contained in $\mathbb{P}$:

$$\rho(i,j) = \sqrt{\sum\nolimits_{r=1}^{q} \left(\widetilde{p_{ir}} - \widetilde{p_{jr}}\right)^2},$$

$$\rho(0,i) = \sqrt{\sum\nolimits_{r=1}^{q} (\widetilde{p_{ir}})^2}$$

- All elements of $\{x, y, z\}$ are contained in $\mathfrak{C}$:

$$\rho(i,j) = \sqrt{\sum\nolimits_{r=1}^{k} \left(\widetilde{c_{ir}} - \widetilde{c_{jr}}\right)^2},$$

$$\rho(0,i) = \sqrt{\sum\nolimits_{r=1}^{k} (\widetilde{c_{ir}})^2}.$$

- Otherwise, $\rho(i,j)$ is not described.

First step of building the standard scene is adding a set of lines, limiting the cube [0, 100], three cylinders to describe axis X, Y and Z with corresponding text captions. Points are complemented to spheres, intervals – to cylinders.

A sphere is a spatial object with the following optical and geometrical parameters.

Geometrical parameters:
$$SphG_i = \{p_i(x_i, y_i, z_i), r\},$$
where p_i – sphere center coordinates, r – radius, stated by the user, equals 1 by default.

Optical parameters:
$$SphO_i = \{Red_i, Green_i, Blue_i\}$$
$$Red, Green, Blue \in [0; 255]$$

A cyllinder is a spatial object with the following optical and geometrical parameters.

Geometrical parameters:
$$CylG_i = \{A, B, r\}$$
where A и B - base points of cyllined,
r – radius, stated by the user.

Optical parameters:
$$CylO_i = \{Red_i, Green_i, Blue_i\}$$
$$Red, Green, Blue \in [0; 255]$$
$$Red_i = 255\left(1 - \frac{\rho(A,B)}{d}\right), \quad Green_i = 150\frac{\rho(A,B)}{d}$$
$$Blue_i = 255\frac{\rho(A,B)}{d}$$
where $d = d_1$ in case of intervals connect points of $\mathbb{P}$ or $\mathfrak{C}$ spaces with the point of origin,

$d = d_2$ in case of intervals connect points of $\mathbb{P}$ or $\mathfrak{C}$ spaces with each other.

A plane is a spatial object with the following optical and geometrical parameters.

Geometrical parameters:
$$PlaneG_i = \{A, B, C\}$$
where A, B и C - points for plane construction, stated by user.

Optical parameters:
$$PlaneO_i = \{Red_i, Green_i, Blue_i\}$$
$$Red, Green, Blue \in [0; 255]$$

The result of this stage consists of geometrical and optical models of the scene. Those models are then used to render a graphical representation of the scene on the next stage.

of geometrical and optical models of the scene. Those models are then used to render a graphical representation

3.1.4. Rendering

Rendering stage result is a graphical representation of spatial scene, obtained on Mapping stage. The following parameters of visualization process are given on this step: camera position, positions of light sources, field of view, physical parameters of space. User can change camera position and angle, choose between orthographic and perspective projections.

3.2 Solving visual analysis task and interpretation of the results with respect to the initial data

The result of Mapping stage is a geometrical and an optical models of the spatial scene. We came up with three types of statements, which can be made in the process of spatial scene.

1)   A statement about difference between sphere positions (values of criteria functions):

After marking several close to each other in one subspace spheres with a color and then switching the subspace, one can understand if this closeness remains in the other subspaces.

2)   A statement about difference between sphere position and the point of origin:

- An analyst can make decisions about sphere closeness to the point of origin in the chosen subspace. Changing the subspace, one can make similar decisions in the n-dimensional space. In the stated in the section 1 multidimensional task point of origin is considered as an ideal point.

- Changing the d_1 threshold value, which influences creation of cylinders between the point of origin and spheres, one can make decisions about closeness of spheres to zero. The degree of how close spheres are to the point of origin can be evaluated by the color of cylinders. Blue color means the length of the cylinder is close to the threshold, while red – far from the threshold. Cylinder absence means the distance between points is more, than threshold.

3)   A statement about spheres clusters:

- Changing the d_2 threshold value, which influences creation of cylinders between the points, one can make decisions about closeness of the n-tuplets between each other.

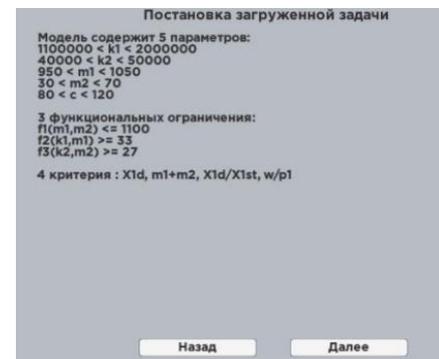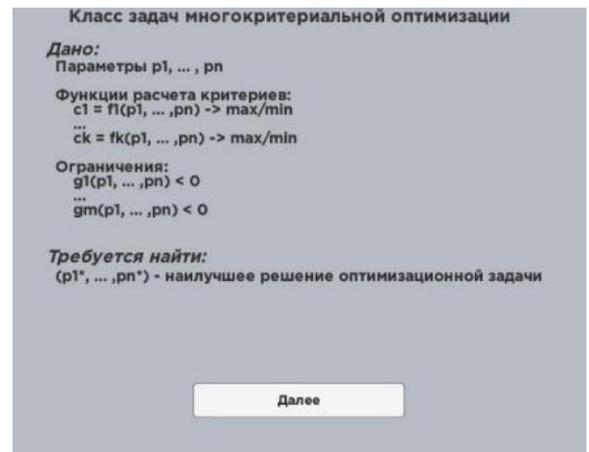Color orientation can be of use in this case, like in the previous one.

- Marking several spheres with color provides to the analyst an ability to make judgements about spheres, which form a cluster in criteria space, while working with criteria space.

4. Program development and the experiment conduction

The development of interactive visual analysis program modules consists of four parts: designing the process of storing and processing obtained multidimensional data, designing the process of obtaining visual representation of data in question, designing the user interface. Visual Analytics Tool has been developed to implement this algorithm. Programming environment is Unity, using scripts on C# and UnityScript, and Lua for problem definition.

The development methodology is iterative. The development team has managed to perform three iterations during the process of making this article. Also, Gitlab repository manager was used during the development process. It provides abilities to not only exchange source code in a fast and handy way, but also helps tracking current tasks, automatic project building and storing current and previous build artifacts.

After program launch, user can see the problem definition, choose the file with task, and proceed to the data generation. The window which a user interact with has three sections. He uses the first one to set discretization parameters and to make a decision about highlining the spheres which correspond to Pareto-optimal solutions. The second one allows to change limits of inequality that describe function limitations Γ. In the third one a user can see generated data in a table.

Класс задач многокритериальной оптимизации

Дано:
Параметры p1, ... , pn

Функции расчета критериев:
c1 = f1(p1, ... ,pn) -> max/min
...
ck = fk(p1, ... ,pn) -> max/min

Ограничения:
g1(p1, ... ,pn) < 0
...
gm(p1, ... ,pn) < 0

Требуется найти:
(p1*, ... ,pn*) - наилучшее решение оптимизационной задачи

Далее

Выбор задачи

Выберите задачу:

Задача 1

Задача 2

...

Задача n

Загрузить файл модели

Назад

Постановка загруженной задачи

Модель содержит 5 параметров:
1100000 < k1 < 2000000
40000 < k2 < 50000
950 < m1 < 1050
30 < m2 < 70
80 < c < 120

3 функциональных ограничения:
f1(m1,m2) <= 1100
f2(k1,m1) >= 33
f3(k2,m2) >= 27

4 критерия : X1d, m1+m2, X1d/X1st, w/p1

Назад   Далее

Генерация многомерных точек

Генерация данных   Редактирование области определения   Полученные данные

Количество точек: 1000   Начать рассчет

Выберите генератор случайных чисел: Случайные ∨

Рассчитать парето-оптимальные точки

Назад

Генерация многомерных точек

Генерация данных | Редактирование области определения | Полученные данные

ID	k1	k2	m1	m2	c	X1d	m1+m2	X1d/X1st	w/p1
0	1273392	47049.41	963.0093	35.30093	82.80376	6.436559	192.2916	4.098131	1.212117
4	1418561	40482.16	1003.685	38.59486	117.7822	5.860565	154.9099	4.156784	1.253153
3	1585963	46806.59	1012.185	52.84835	107.3402	1.944637	160.6637	1.542061	1.319458
1	1863061	40178.71	986.4964	43.4132	82.80376	2.288552	176.9168	2.131856	1.448587
5	1801650	44494.07	1032.705	59.05857	104.6814	1.798002	146.3533	1.619685	1.392277
9	1326768	48014.18	964.961	31.67497	114.6395	5.135379	186.714	3.406728	1.236008
7	1724084	48917.82	1034.253	63.48133	96.4991	1.823722	149.2286	1.572125	1.360958
11	1252542	49971.77	955.887	52.07533	96.85668	5.64187	216.1884	3.533339	1.206623
12	1710407	40193.74	1012.181	39.49469	97.38151	3.435937	147.3138	2.938425	1.370249
15	1290055	47771.68	1007.782	30.96899	85.52554	6.312769	143.1873	4.071911	1.192613

Назад

Figure 6. Windows of the program.

The scene is controlled with several windows, capable of changing d1 и d2 limitations, sphere radiuses, camera projection, also capable of sphere selection and adding limiting planes to the scene. Those panels can be minimized.

4.1. Program results

The program is capable of generating the specified amount of data, visualizing that data and timely reacts to user actions. The process of testing included generation of 100'000 points. During testing, the calculation process never took more than 60 seconds.

Let us note that the design of the user interface allows an analyst to conveniently manipulate with data. He is able to not only change the {x, y, z} subspace, to add the cylinders on scene, but also to move camera and to switch its projections. We should highlight the analyst also has an opportunity to see the generated data in a table.

5. Problem analysis

Let us condust the visual analysis of the spatial scene, described in section 3.2.

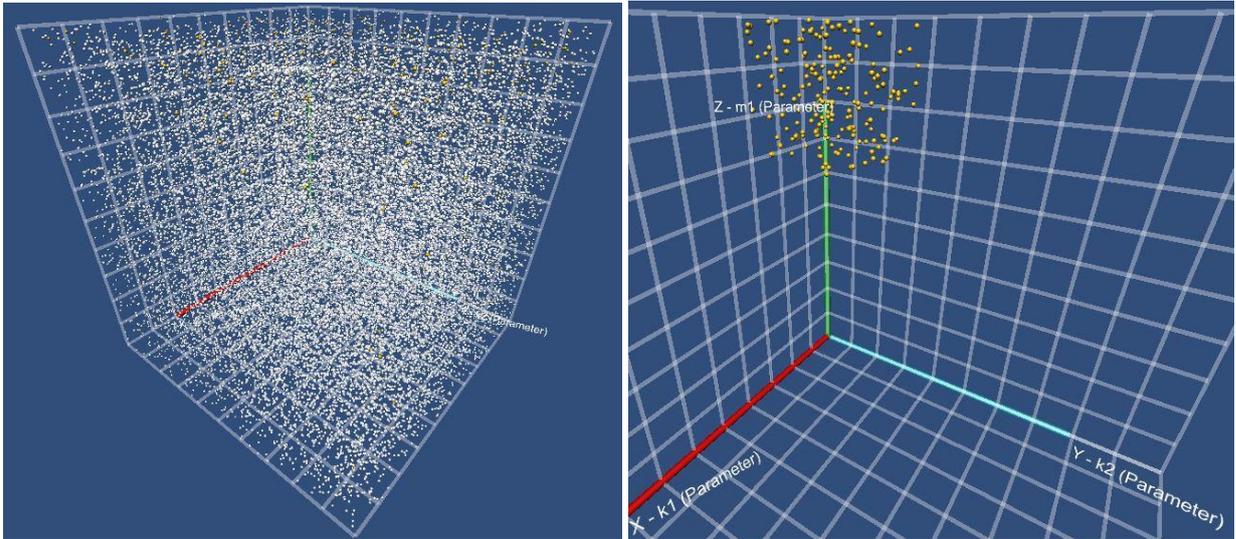A cloud of spheres is marked in the parameter's space:

Figure 7. Spheres cloud

Next ranges of values are considered:
$$1.1 \times 10^6 \leq K_1 \leq 1.37 \times 10^6;$$
$$4.0 \times 10^4 \leq K_2 \leq 4.3 \times 10^4;$$
$$1020 \leq M_1 \leq 1050;$$
$$30 \leq M_2 \leq 58;$$
$$80 \leq C \leq 120.$$

The parameters space is changed to criteria one. The threshold value is assigned and then the spheres from the cloud are grouped by the cylinders. That is to say, the clusters are build up.
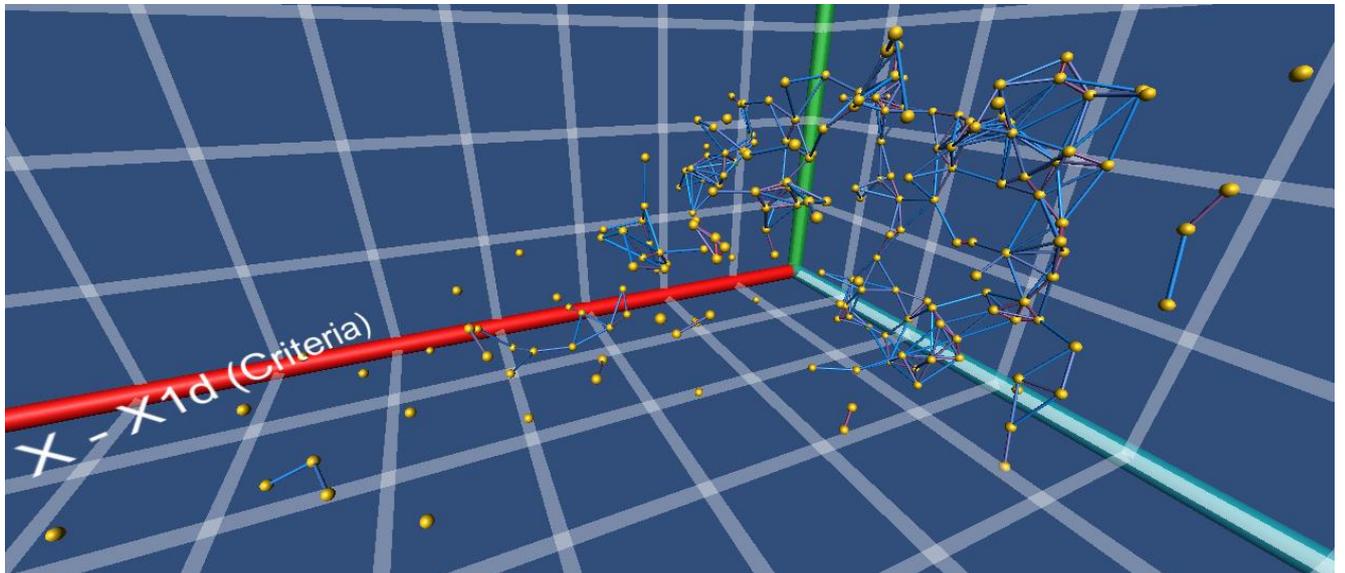

Figure 8. Clusters in the criteria space

The subspace which is formed by the objective functions $X_{1\partial}, m_1 + m_2, \frac{\omega}{p_1}$, according to an abscissa, ordinate and applicate axis is changed by replacing the objective function $\frac{\omega}{p_1}$ to $\frac{X_{1\partial}}{X_{1st}}$.
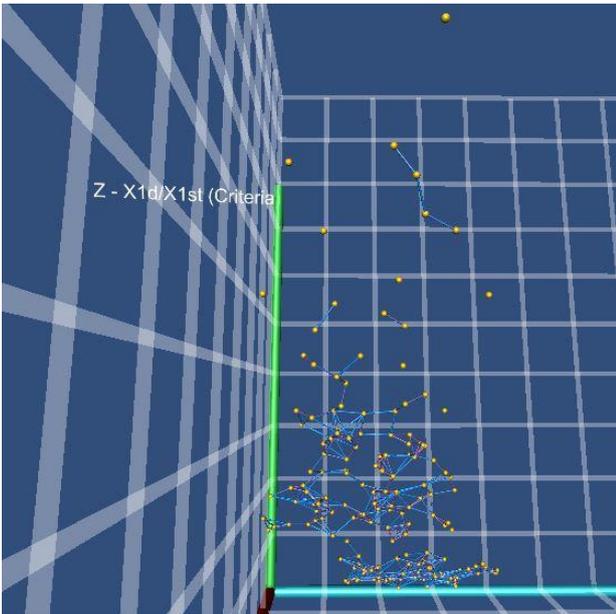
Figure 9. Clusters in the changed sub-space

Let us make a type 3 statement (about sphere clusters) from figure 10. One can observe two pair of spheres, which have the same distance between them in the 3d-space, have different colored cylinders. That is attributable to difference between the other criteria.
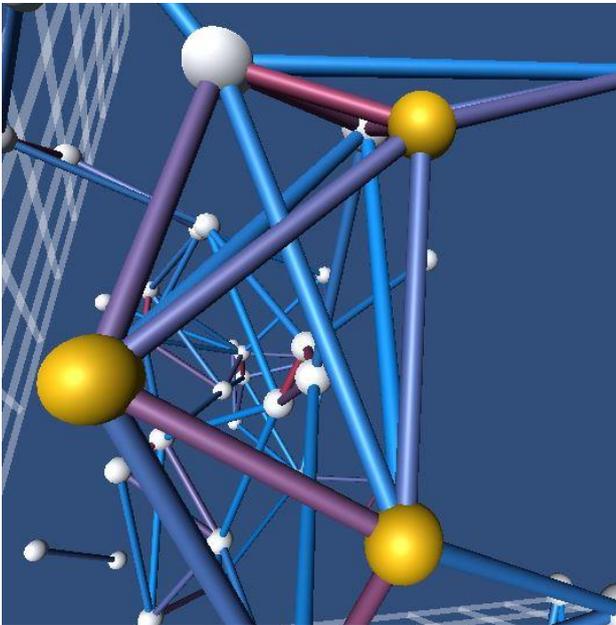
Figure 10. A pair-wise comparison of distance between the spheres

The cluster of spheres which are close to each other is formed by the cloud in the parameters space, while the same spheres are separate in the criteria space. Thus, it may be concluded that a slight change in the pa-

rameter's space leads to considerable change in the criteria space.

Let us make a statement type 2 statement (about difference between the objective functions values and zero. A signification of the statement is argued by the minimization task. The smallest objective functions values are marked by user during the visual analysis.
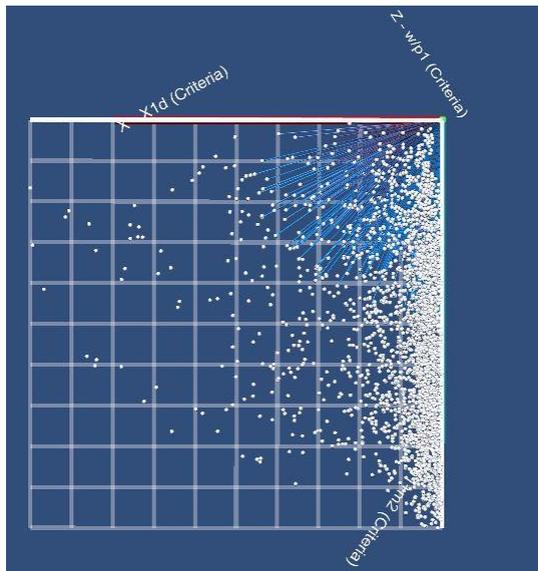
Due to a figure 11:

1. The function values of $X_{1\partial}$ are close to zero - the large cluster is formed by the spheres near OYZ plane.

2. The functions values of $\frac{\omega}{p_1}, m_1 + m_2$ distributed over the entire domain of definition. The objective function values are far from zero, the spheres are distributed across the entire scene in a cross projection.
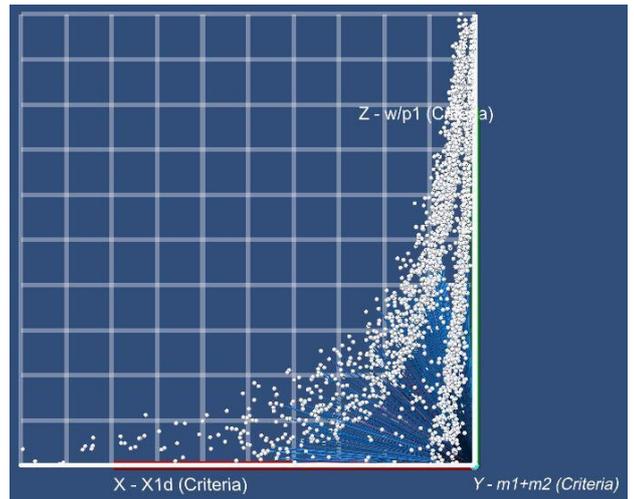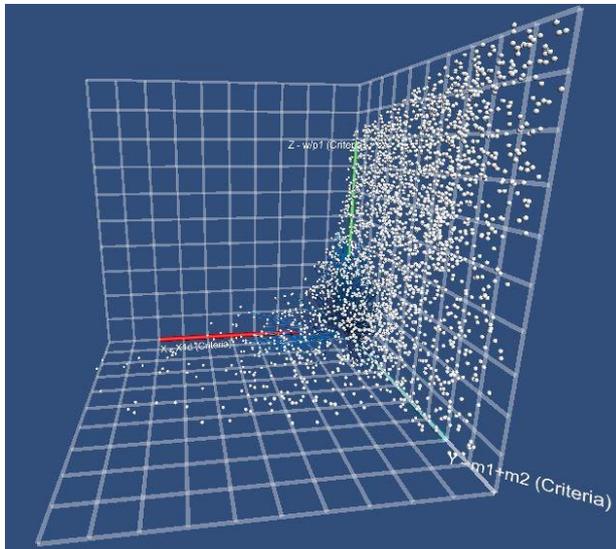

a
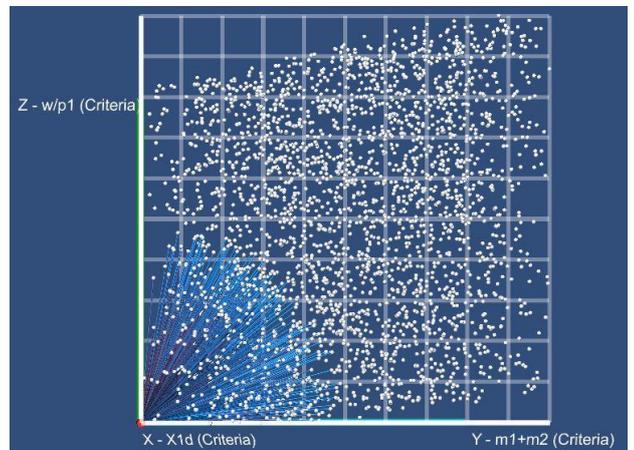
b

c

d

Figure 11. Scene which consists of spheres and cylinders:
a – frontal projection; b – cross projection; c – floor projection; d – projection picture

Let us make a type 1 statement (about difference between sphere positions). The cloud of points is formed by the objective functions' values $\frac{\omega}{p_1}$ and $X_{1\partial}$. A cloud boundary reminds of the shape of a hyperbolic curve. Thus, these functions might have a hyperbolic functional connection.

Functions $\frac{\omega}{p_1}$ and $m_1 + m_2$ do not have an explicit functional connection. The spheres are uniformly distributed.

The values of functions $X_{1\partial}$ and $m_1 + m_2$ are distributed closely to zero. The spheres are distributed along one of the axis.

The closest to the origin point spheres in the criteria space are marked with red color. The ability to render cylinders to the origin point is used.
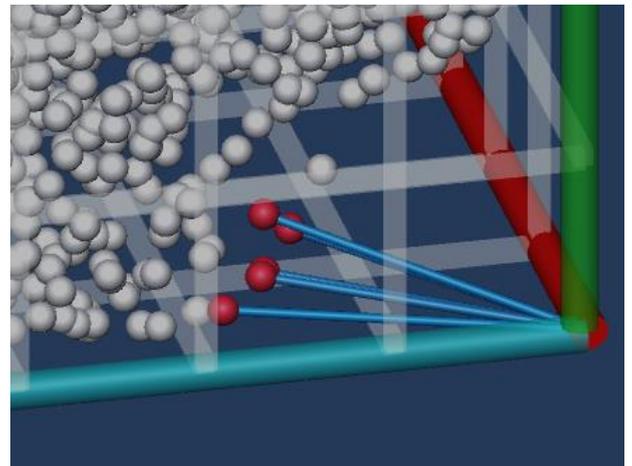

Figure 12. Scene with spheres and cylinders

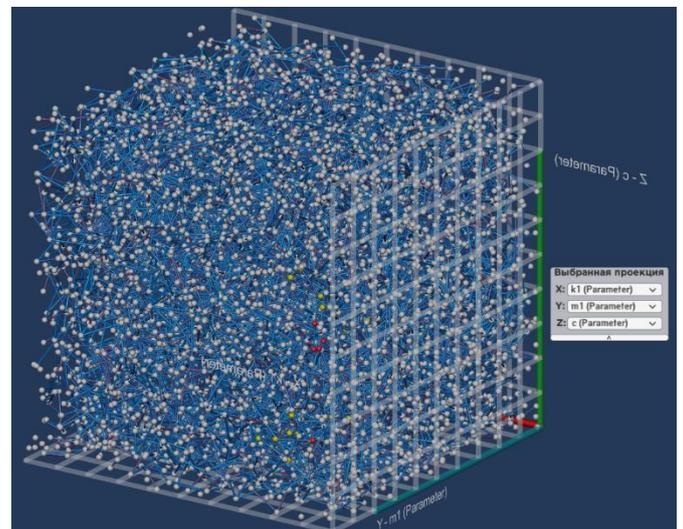Switch to the parameter space and mark yellow spheres closest to the red ones.

Figure 13. Scene with spheres and cylinders
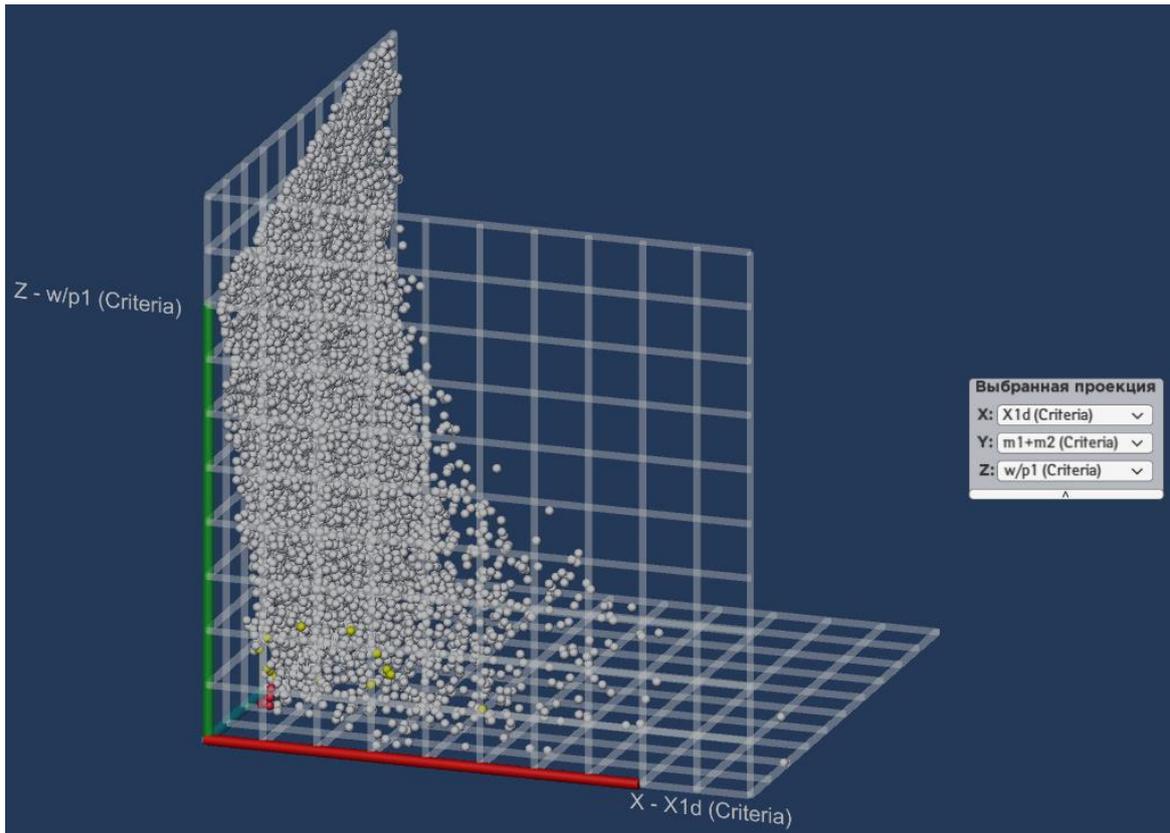
Switch back to the criteria space.

Figure 14. Scene with spheres and cylinders

Let us make a type 1 statement for the figure 14. Spheres, which are close to the initially marked (red) spheres in the parameter space, are not close to them in criteria space. This means, that a slight parameter change leads to considerable change in criteria space.

Conclusion

This article shows the process of visual analytics software development and its future use. Visual analytics was applied to multicriteria optimization task, described at the beginning of the article. The data model and the program algorithm were designed to analyze data of multicriteria optimization task. Using the developed program, the data of twomass dynamic model was analyzed.

The sensitivity analysis of given task was conducted; the solutions are unstable: the slight change of parameters causes significant change in optimization criteria values. The cloud of solutions, optimal by aggregated criteria $\sqrt{\sum_{r=1}^{q}(c_r)^2}$, was visually selected. Judgements about dependence between optimal criteria were formulated.

Future development is planned in various directions. First of all, the improvements in user interactions are planned: working with several 3D-projections at once, additional capabilities of optical parameters change (having spheres on scene colored with different colors), 2D-graphs for barr charts analysis.

Second is adding of numerical methods, which allow to obtain the potentially optimal solutions without using the visualization method.

References

1.   Multidimensional Data Analysis for Multiparametric Optimization Problems Using Visualization / A. Bondarev, V. Galaktionov // Scientific Visualization. – 2012. – Vol. 4 – No. 2 – PP.1-13.

2.   A research of procedures used in the analytic hierarchy process and visuali-

zation in sensitivity analysis / V.V. Kotova // Scientific Visualization. – 2016. – Vol. 8. – No. 2. – PP. 59-84.

3. Using visualization in solving discrete mcda-problem by methods of promethee family / A.Yu. Yakovlichev, I.E. Milman, V.V. Pilyugin // Scientific Visualization. – 2016. – Vol. 8. – No. 3. – PP. 78-84.

4. Scientific visualization as method of scientific data analysis / V. Pilyugin [et al.] // Scientific Visualization. – 2012. – Vol 4. – No. 4. – PP. 56-70.

5. Usage of visualization in the solution of multicriteria choice problems / A.P. Nelyubin [et al.] // Scientific Visualization. – 2017. – Vol. 9. – No. 5. – PP. 59-70.

6. Statnikov R. The Parameter Space Investigation Method Toolkit. / R. Statnikov, A. Statnikov – Artech House. – 2011. – 214 c,

7. Implementing Sobol's Quasirandom Sequence Generator / Stephen Joe, Frances Y. Kuo // ACM Transactions on Mathematical Software, Vol. 29, No. 1, March 2003, PP. 49–57.

8. Data analysis of credit organizations by means of interactive visual analysis of multidimensional data / I.E. Milman [et al.] // Scientific Visualization. – 2015. – Vol. 7. – No. 1. – PP. 45-64.