Situation Assessment Using Results of Objects Parameters Measurements Analyses in IGIS

Andrey Pankin[1], Alexander Vodyaho[2], Nataly Zhukova[1]

[1]Saint-Petersburg Institute for Informatics and Automation of the Russian Academy of Sciences, Research laboratory of object-oriented geo-information systems, Russia

[2]Saint-Petersburg Electrotechnical University, Russia

aivodyaho@mail.ru, {pankin, gna}@oogis.ru

Abstract. The paper describes method for situations assessment based on retrieving information about similar earlier observed conditions. Situation is a set of qualitative and quantitative characteristics that describe states of interrelated objects. Object states are defined by measurement parameters. A method for situation assessment is based on calculation of aggregated indices and their comparison was developed. For calculating aggregated indices it is proposed to use an algorithm for alphabetic description of time series that provide convenient means for their comparison. For situations retrieval it is suggested to use FCA methods. As a case study the results of ocean data analyses for calculating temperature and salinity parameters of water area are presented.

Keywords: situation assessment, measurements analyses, summary indicators

1 Introduction

Nowadays Intelligent Geographic Information Systems (IGIS) are widely used for solving different functional tasks. IGIS incorporates GIS interface as well as various methods of artificial intelligence intended for solving certain intricate problems including problem of decision making support. Decision support systems in IGIS are aimed to provide end users with complex information about the solved problem as well as with reasonable alternative decisions in real time with a pictorial rendition of this information to let it be easily perceived and used.

One of the important tasks, that is solved in decision making support systems, is situation assessment and awareness. Situation assesment is aimed to make situations understandable by users. Situation awareness is the perception of the elements in the environment within a volume of time and space, the comprehension of their meaning, and the projection of their status in the near future [1]. Situation assessment represents analysis of available information in order to get validated estimations of current system state and probable direction and dynamic of its changing. In this context term “system” can have wide interpretation: it can be used to describe dynamic technical or environmental objects, set of interacting objects, analyzed phenomena, entities, or environments.

Problem of situation assessment can be decomposed into two subtasks. The first task is situation recognition and the second is making decision about system state. For situation recognition an approach based on comparing situations to ones that were earlier observed is widely used. To provide this knowledge base of situations a description is formed. Decision about system state is based on knowledge about recognized situations. Various directions of situation development can be considered using modeling tools or expert systems.

By now one of the key means for storing information about systems actual states are measurements instruments that provide information about system parameters in real time. Using these measurements ability to analyze the dynamic of a state evaluation and control a system state is provided. There are several problems related with measurements processing. First problem is great volume of data that has to be processed in limited time. Second problem is that measurements have rather bad quality; they are not coordinated in time and space and are implemented as non-stationary time series. Consequently, highly specialized methods have to be used for measurements processing. Third problem is a necessity to represent measurements as a set of complex characteristics, so that they can be used in methods of situation assessment.

In the paper an approach to situation assessment based on comparison of situations is extended for using measurements of system parameters as one of important information sources and approach to retrieval situations using formal concept analyses (FCA) methods is proposed. In the second section general description of the method for situation assessment based on measures analyses in presented. Following sections provide detailed description of algorithms used in the general method. An algorithm of alphabetic representation of time series given in section 3 is aimed to represent time series in a form that provides easy mechanism for comparing parameters. In section 4 the algorithm of identification of information valuable parameters that allow ranging parameters according to information values is considered. In section 5 algorithm for objects aggregated indicators calculating that takes into account values of parameters measurements is presented. Algorithms for building and comparing graphs that describe situations in terms of objects and their relations are discussed in section 6. In section 7 application of Formal Concept Analysis methods for revealing earlier observed distinguishable situations are considered. As a case study task of ocean parameters estimation using measurements provided by floating hydrographic buoys is described.

2 Main definitions

FCA is a well-established technique in mathematics that is widely used for solving various tasks of intelligent data analyses. Standard FCA definitions are introduced in [2, 3]. Given a formal context $K = (G, M, I)$, where G is called a set of objects, M is called a set of attributes, and the binary relation $I \subseteq G \times M$ specifies which objects have which attribute, the derivation operators $(\cdot)^I$ are defined for $A \subseteq G$ and $B \subseteq M$ as follows:

$$A^I = \{m \in M \mid \forall g \in A : g\, \mathrm{Im}\};$$
$$B^I = \{m \in M \mid \forall m \in B : g\, \mathrm{Im}\}.$$

A^I is the set of attributes common to all objects of A and B^I is the set of objects that share attributes of B. For simplicity operator $(\cdot)'$ is used instead of $(\cdot)^I$. The double application of $(\cdot)'$ is a closure operator; it is extensive, idempotent, and monotonous. Therefore, sets A'' and B'' are closed sets.

A formal concept of the context (G, M, I) is a pair (A, B), where $(A \subseteq G)$, $(B \subseteq M)$, $A = B'$, and $B = A'$. In this case $A = A''$ and $B = B''$. The set A is called the extent and B is called the intent of the concept (A, B). In categorical terms a formal concept is defined by its objects A or its attributes B.

A concept (A, B) is a subconcept of (C, D) and (C, D) is a superconcept of (A, B) if $(A \subseteq C)$ (equivalently, $(D \subseteq B)$). For (A, B) and (C, D) relations $\geq$, $\leq$, $<$, and $>$ are defined and written as usual. (A, B) is a lower neighbor of (C, D) (notation is (A, B) $\prec (C, D)$) and (C, D) is an upper neighbor of (A, B) (notation is $(C, D) \succ (A, B)$) if $(A, B) < (C, D)$ and there is no (E, F): $(A, B) < (E, F) < (C, D)$. The set of all concepts ordered by $<$ forms a concept lattice of the context K, that is denoted by $B(K)$. The relation $\prec$ defines edges in the covering graph of $B(K)$.

For building lattices while solving task of situations analyses formal context as a set of objects G situations are considered, M is a set of situations characteristics, I is an incidence relation between these sets. Each situation s is characterized by a set of relevant objects $E = \{O_i\}_{i=1}^{N}$ and relations between objects $R = \{r_{i,j}\}_{i,j=1}^{N}$, where N is a total number of objects. For each object a set of parameters $e = \{P_i\}_{i=1}^{M}$ that describes objects state is defined.

3 General description of method for situation assessment

Situation assessment is based on comparing current conditions with the previously observed ones. Situation involves objects that can be technical or natural and relation between them. Relations are described for pairs of objects, for each relation its type is defined. All types are to be described a priori in a vocabulary of subject domain. An object state is characterized with a set of parameters; values of parameters are measured using various measurement instruments and are represented as time series.

When solving problem of situation assessment it is necessary to provide an effective and efficient mechanism for situation comparing to retrieve similar ones. For comparing two situations it is necessary to compare list of objects and their states and relations between objects. Objects and relations between objects are reasonable to represent as a graph, where vertexes of the graph are objects and edges of the graph are relations.

For comparing two graphs a wide range of methods is developed. The most convenient algorithm for similar situations retrieval is based on graph edit distance. The main idea of this algorithm is to define difference between graphs using a set of editing operations that are necessary for transforming one graph to the other. This method is tolerant to errors and provides inexact graph matching. Algorithms for graph edit distance calculation are described in [4]. When edges of graph are compared the result is binary – if the relations that corresponds to the edges are equal then result of comparison is '1' else the result is '0'. For comparing vertexes it is necessary to compare objects associated with them. As each object is characterized with a set of parameters to build object description it is necessary to solve two problems –to describe each time series of parameters measurements in such a way that descriptions can be easily compared and to define how to calculate aggregate characteristic of objects using formed descriptions.

For describing parameters measurements it is proposed to use alphabetic representation of time series. To build alphabetic representation method based on Symbolic Aggregate Approximation (SAX) [5] is used. Strings that are composed with SAX-based algorithms can be compared using string Edit Distance that is used in algorithms of string inexact comparing [6].

For solving the second task Aggregated Indices Randomization Method (AIRM) [7] can be applied that is targeting complex objects subjected to multi-criteria estimation under uncertainty. The essence of application of AIRM consists in an aggregation of single characteristics into one complex characteristic that is used for comparing objects. One of the key tasks that is to be solved before ARIM method can be applied is to define weights for objects parameters that are considered as indicators. Taking into account that parameters are characterized with measurement time series for evaluation of time series information value a set of statistical characteristics is used [8].

General description of proposed method for situation assessment is given in Fig.1

Input data. Data base of graphs describing earlier observed situations, description of estimated situation, that includes $E = \{O_i\}_{i=1}^{N}$ is a set of objects, $R = \{r_{i,j}\}_{i,j=1}^{N}$ is a set of objects relations, $O = \{P_i\}_{i=1}^{M}$ is set of objects parameters, $P = \{(t_i, x_i)\}_{i=1}^{H}$ is a time series of parameters measurements, where H is a total number of measurements.

Output data. $S = \{(s_i, q_i)\}_{i=1}^{T}$ is a set of situations s that are similar to a defined situation s^d with similarity degree q.

Algorithm description

A. Building description of estimated situation

Step A1 **build** symbolic representation of objects parameters measurements $\hat{C} = f_{symb}(P)$

Step A2 Building descriptions of objects

calculate weights of parameters $W = \{w_i\}_{i=}^{M}$ according to information value

calculate estimations of aggregated indices for objects $\tilde{Q} = \{\tilde{Q}_i\}_{i=1}^{N}$

B. Building graph for situation description

Step B1 Defining graph vertexes G_V using formalized descriptions of objects

Step B2 Defining graph edges G_D using formalized descriptions of objects relations

C. Situation estimation

Step C1. Reveling similar graphs of situation description in data base

Step C2. Ranging graphs according to degree of similarity $S = \{(s_i, q_i)\}_{i=1}^{T}$

Fig.1 General description of method for situation assessment

4 Algorithm of alphabetic representation of time series

Proposed algorithm of alphabetic representation is based on algorithm of Symbolic Aggregate Approximation (SAX) described in [5]. In SAX for building symbolic representation of time series approach based on application of Piecewise Aggregate Approximation (PAA) is used. According to the algorithm time series are presented as a sequence of segments using window of defined length. For each segment a set of defined statistical characteristics are estimated. PAA can be considered as an attempt to represent a time series in a form of windows line combination. The description of the algorithm is given in Fig. 2. PAA representation of time series is converted into symbolic representation. In SAX it is assumed that analyzed time series have normal distribution, but measurements time series very often doesn't satisfy this criterion. In [9] the description of modification of SAX for time series with various distributions is proposed. The modified procedure assumes, at first, estimation of measurements values interval. To avoid usage of values that contain noise and outliers for determining border median values of K, minimum and maximum values are used. Second, interval of values are split into equal intervals, each part corresponds to one level. Segments, which characteristics correspond to one interval, are the segments of one level and they are described using same symbol from a priori defined alphabet (Fig. 3).

Input data. $P = p_1, ..., p_H$ is an initial time series, where H is a number of segments.

Output data. $\overline{C} = \bar{c}_1, ..., \bar{c}_z$ is aPAA representation of time series.

Algorithm description

Step 1. **calculate** length of one segment $l = \frac{H}{z}$

Step 2 **for** ($i = 1 ... z$)

$\bar{c}_i = f(\{c_j\}_{j=l(j-1)+1}^{l-i})$, where f is a function of calculating segment statistical characteristics

Fig.2 Algorithm for building PAA representation of time series

Input data. $P = p_1, ..., p_H$ is an initial time series, where H is a number of segments, $A = a_1, ..., a_k$ is an alphabet for time series symbolic representation, $B = \beta_1, ..., \beta_{k-1}$ are levels of time series representation.

Output data. $\hat{C} = \hat{c}_1, ..., \hat{c}_z$ is asymbolic representation of time series.

Algorithm description

Step 1. **calculate** $\overline{C}$ using algorithm for PAA representation of time series

Step 2. **calculate** range of time series characteristic values [V_l , V_h], where V_l - low border, V_h - high border

Step 3. **calculate** range of characteristics values for each level β

Step 4. **for (** $i = 1 ... w$ **)**

define alphabet symbol $\hat{c}_i = a_j \Leftrightarrow \beta_{j-1} \leq \overline{c}_j < \beta_j$

Step 5. **concatenate** symbols $\hat{C} = \{\hat{c}_i\}_{i=1}^{z}$

Fig.3 Algorithm for building time series symbolic representation

By now many algorithms that allow to deal with strings, in particular, algorithms of inexact string comparison, based on calculation of Edit Distance are developed. Algorithms of string comparison are applied for qualitative evaluation of time series similarity.

5 Algorithm of information valuable parameters identification

Each object is described by a set of various parameters. Degree of information value of each parameter differs and it is necessary to take it into account when two objects are compared. The degree of parameter information value is used to range parameters in algorithm of calculating objects aggregated indices.

The proposed algorithm of calculating degree of parameter information value is based on using a set of statistical characteristics. Depending on objects characteristics different measures for time series described in [8] can be calculated. Most often the following measures are used: mean, median, variance, standard deviation, interquartile distance, skewness and kurtosis. The algorithm of ranging parameters is based on the idea that most informative are measures that have maximum difference for different objects. So mean distances between measures of parameters time series are calculated and according to them parameters are ranged and preliminary weight coefficients are defined. The proposed algorithm is given in Fig. 4.

Input data. $E = \{O_i\}_{i=1}^{N}$ is set of objects, $O = \{P_i\}_{i=1}^{M}$ is a set of measured objects parameters, where M is a total number of objects parameters, $G = \{g_i\}_{i=1}^{U}$ is a list of time series measures, U is a total number of measures.

Output data. $\overline{P}$ is a sorted set of parameters, $W = \{w_i\}_{i=1}^{M}$ is a list of preliminary weight coefficients for parameters.

Algorithm description

A. Calculating measures for parameters time series

Step A1. **for each** parameter $(i = 1 \ldots M)$

for each object $(j = 1 \ldots N)$

calculate measures $s_{ij} = (s_{ij}^1, ..., s_{ij}^U)$

calculate mean distance $\bar{s}_i = \frac{1}{N} \sum_{k=1}^{M} \sum_{l=1}^{M} \sqrt{(s_{ik} - s_{il})^2}$

B. Ranging parameters

Step B1 **for** $(i = 1 \ldots M)$

define preliminary weights $w_i = \bar{s}_i$

Step B2 **sort** parameters according to preliminary weights $\overline{P} = sort(\{P_i\}_{i=1}^M)$

Fig.4 Algorithm for ranging parameters

6 Algorithm for objects aggregated indicators calculating

To calculate objects aggregated indicators based on set of parameters it is proposed to use indices randomization method. ARIM is used to solve tasks of multiple criteria decision making on the base of poor-quality input information. The main advantage of AIRM is its ability to cope with non-numeric (ordinal), non-exact (interval) and non-complete information. When solving user's tasks information about objects parameters is often incomplete as parameters due to different reasons can't be gathered. Calculated in section 5 preliminary weights of parameters provide approximate estimation of parameters information value and therefor can't be used directly for calculating objects aggregated indicators. Preliminary parameters weights are used to range parameters and thus provide ordinal information about parameters. This information can be effectively used in AIRM.

In ARIM three key steps are executed: i) building vector of single indicators; ii) defining aggregative function; iii) defining weighs coefficients.

Main features of ARIM application for calculating objects indicators using measurements are the following:

1. Results of symbolic representation of time series of parameters measurements build according to algorithm described in section 3 are considered as list of objects characteristics.

2. Single indicators for objects are functions of objects characteristics. They are defined as normalizing power functions of degree one. When characteristic values increase functions also increase.

3. An aggregative indicator is a synthesized function that characterizes each object in general. It depends on weight coefficients and is represented in a form of linear convolution of single indicators functions and weight coefficients.

4. As information I about objects parameters weights is incomplete, weight-vector $w = (w_1, \ldots, w_m)$ is ambiguously determined. In ARIM this vector is deter-

mined with accuracy to within a set $w(I)$ of all admissible weight-vectors. An uncertain choice of a weight-vector from set $w(I)$ is modeled by a random choice of an element of the set according to the concept of Bayesian randomization. Such randomization produces a random weight-vector $w(I) = (w_1(I), \dots, w_m(I))$, which is uniformly distributed on the set $w(I)$. Set $w(I)$ is reduced using ordinal and interval information. Mathematical expectation of random weight coefficient $w_i(I)$ may be used as a numerical estimation of particular indicator q_i significance. Then randomized weight-vector can be defined as $\tilde{w}(I) = (\tilde{w}_1(I), \dots, \tilde{w}_M(I))$. The precision of this estimation is measured by standard deviation of the corresponding random variable.

The algorithm for objects summary indicators calculating is given in Fig.5.

Input data. $E = \{O_i\}_{i=1}^{N}$ is a set of objects, $O = \{\hat{C}_i\}_{i=1}^{M}$ is a set of symbolic representation of measured objects parameters, where M is a total number of objects parameters, $\hat{C} = \{\hat{c}_i\}_{i=1}^{z}$ is a symbolic representation of parameter, where z is a length of symbolic representation.

Output data. $\tilde{Q} = \{\tilde{Q}_i\}_{i=1}^{N}$ are estimations of objects aggregated indicators.

Algorithm description

Step 1. **for each** object $(1, \dots, N)$

define $\hat{C} = (\hat{c}_1, \dots, \hat{c}_M)$ as set of initial characteristics

calculate vector of single indicators $q = (q_1, \dots, q_M)$,

$$q_j = q_j(\hat{c}_j) = \begin{cases} 0, & \hat{c}_j \le MIN_j, \\ \left(\dfrac{\hat{c}_j - MIN_j}{MAX_j - MIN_j}\right), & MIN_j < \hat{c}_j \le MAX_j, \\ 1, & \hat{c}_j > MAX_j; \end{cases}$$

MIN_j and MAX_j - minimum and maximum values of characteristic

calculate randomized weight-vector for characteristics $\tilde{w}_i = (\tilde{w}_1, \dots, \tilde{w}_M)$

calculate aggregated indicator

$$\tilde{Q}\ (q; I) = Q(q; \tilde{w}(I)) = Q(q_1, \dots, q_m; \tilde{w}_1(I), \dots, \tilde{w}_m(I)) = \sum_{i=1}^{m} q_i \tilde{w}_i(I)$$

calculate estimation of aggregated indicator

$$\overline{Q}\ (I) = E\tilde{Q}(I)$$

Step 2. **form** vector of aggregated indicators estimations $\tilde{Q} = (\tilde{Q}_1, \dots, \tilde{Q}_N)$

Fig.5 Algorithm for objects aggregated indicators calculating

7 Algorithms for building and comparing situation graphs

A situation graph contains information about objects, a set of characteristic that are sufficient for objects description, and relations between objects. Building a situation graph assumes following main steps: i) making a list of objects that are significant for situation description; ii) defining set of objects characteristics; iii) defining set of admissible relations between objects; iv) building structure of the graph. All tasks are enumerated but the last one is solved by experts manually. A set of objects characteristics contains aggregated characteristics of measured parameters that are defined in section 5 and it may also contain one or several additional characteristics. Usually, as additional characteristics, time and earth coordinates of parameters measurements are considered. The algorithm for building situation graph is given in Fig. 6.

Input data. $E=\{O_i\}_{i=1}^{N}$ is a set of objects, $R=\{r_{i,j}\}_{i,j=1}^{N}$ is a set of objects relations, $F=\{f_i\}_{i=1}^{Y}$ is a set of object characteristics, where Y is a total number of object characteristics.

Output data. $G=<G_V, G_D>$ is a situation graph, G_V are graph vertexes and G_D aregraph edges.

Algorithm description

Step 1. **define** empty graph $G_V \leftarrow []$, $G_D \leftarrow []$

Step 2. **create** vertexes from objects $G_V \leftarrow E$

Step 3. **create** edges for related objects $G_D \leftarrow R$

Step 4 **for each** vertex $v_i \in G_V$ $(i=1,...,N)$

  define attributes $A_{vi}=a_v(O_i)$ according to characteristics of object O_i

Step 5. **for each** edge $d_{i,j}=d(O_i,O_j)$, $i=1,...,N$, $j=1,...,N$

  if (d_i **exists**)

    define attributes $A_{di}=a_d(r_{i,j})$ according to defined relation between objects O_i, O_j

---

Fig.6 Algorithm for building situation graphs

Widely used methods for comparing graphs are based on calculation of graph edit distance [1, 4]. The main idea of these methods is to find minimum number of graph editing operations (edit path) that will allow the transformation one compared graph to another. Edit distance d for graphs G_1 and G_2 can be defined as: $d(G_1,G_2)=\min\limits_{(e_1,...,e_k)\in \Upsilon(G_1,G_2)} \sum_{i=1}^{k} c(e_i)$, where Υ are all possible edit paths, e is a graph editing operation, $c(e)$ is the cost of operation e. The key advantage of these methods is their flexibility as methods are able to deal with any graphs and any types of vertex and edge attributes. The standard set of graphs operations include following operations: adding, removing and modifying elements.

The described group of methods allows finding optimal solution, but is complicated from computational point of view. Due to this fact if situation description contains considerable number of object and relations, it is proposed to use suboptimal methods for graph comparison [10, 11]. According to these methods graph is decomposed into a set of sub graphs. Each sub graph contains one vertex and edges that are related to the vertex. The task of comparing two graphs is substituted by the task of comparing sets of sub graphs.

The alternative approach for suboptimal graph comparing is based on using Hungarian method [12, 13]. It assumes searching optimal matching of vertexes and their local structure using approximation of graph Edit Distance.

In case if a priori knowledge about objects and their relations for different types of situations is available, complexity of comparing graphs methods can be significantly reduced.

8 Algorithm for revealing situations using FCA

The approach for situations assessment based on building and comparing graphs supposes that a data base of situations is created a priori. The task of creation of a universal mechanism for distinguishable situations retrieval is highly complicated as situations are often rather similar; they have a number of equal characteristics, relations and involved objects. Since there are many situations and each situation is described by huge volume of heterogeneous data it is proposed to use Formal Concept Analysis methods [2] for revealing equal and different features of situations, interconnected situations, and groups of similar situations.

To build lattices formal context K is defined using a set of defined situations and their characteristics. Characteristics can be binary, quantitative or qualitative. Binary characteristics can be used directly for building a context. Qualitative characteristics can be considered as a set of adjusted characteristics, where each of characteristics values correspond to one adjusted characteristic. For representation of quantitative characteristics in binary form nominal scales can be used. This approach is rather flexible as it allows user to modify scales manually. It is also possible to build lattices using multivalued contexts that are defined as $K = (G, M, W, I)$, where W is a set of situations characteristics values, I is a ternary relation, $I \subset G \times M \times W \times I$, where process of scaling is automated. Approaches for building lattices using multivalued contexts are described in [14, 15].

The algorithm for revealing situations using FCA supposes executing of three main stages. The first stage assumes building formal context for representation situations and their characteristics. As objects of formal context a preliminary list of situations defined by experts is used. A list of context features contains set of three characteristics for each involved subject domain object. A set of used characteristics is equal to the set that is used for building graphs. Each object is characterized by i) its name or id, ii) its location in space and, if necessary, in time and iii) aggregated indicators. All characteristics are represented in binary form. For building nominal scale for aggregated indicators, ranges for values are defined using entropy based methods, in particular, Gini [16] evaluation measure. At the second stage FCA methods are applied to build concept lattice [17]. At the third stage formal concepts are analyzed by experts that modify the preliminary list of situations and, in separate cases, the list of features using obtained results. The algorithm for revealing situation using FCA is given in Fig. 7.

Input data. $E=\{O_i\}_{i=1}^{N}$ isa set of objects, $F=\{f_i\}_{i=1}^{Y}$ is a set of object characteristics, where Y is a total number of object characteristics.

Output data. $S=\{s_i\}_{i=1}^{K}$ is a set of situations, where K is a number of revealed situations.

Algorithm description

Step 1 **define** preliminary list of situations $S=\{s_i\}_{i=1}^{K}$

Step 2 **calculate** characteristics of situation
 for each situation $s_i \in S$
  for each object $e_j \in E$ involved in s_i
   calculate object characteristics F_{ij}
   convert characteristics to binary form $F_{ij} \to F_{ij}^{B}$

Step 3 **build** formal context K
 define formal objects $G \leftarrow S$
 define formal objects features $M \leftarrow \{E, F^{B}\}$
 define relations I

Step 4 **build** lattice

Step 5 **improve** set of situations S

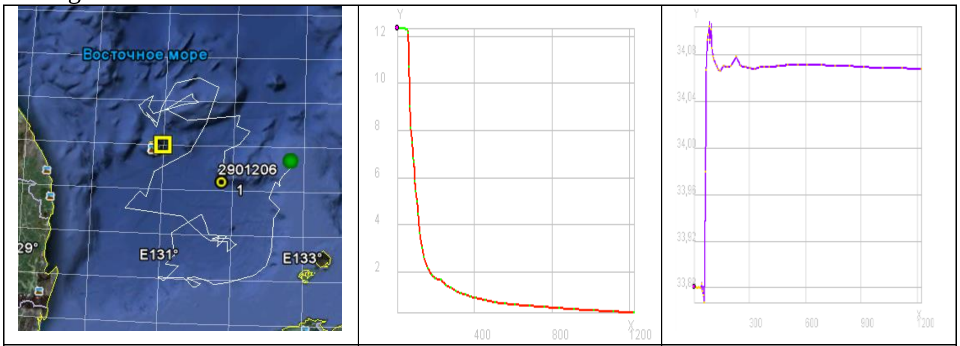
Fig.7 Algorithm for revealing situations using FCA

9 Case study

The proposed approach for situation assessment was used for solving task of providing operational information about ocean temperature and salinity parameters for hydroacoustics calculations that use sound speed of water area as one of parameters. Regular grids of parameters values are usually used as a source for information about water area state. Performing processing and analysis of available oceanographic data in order to build regular data grids includes two main steps: data verification and data regularization. The main purpose of data verification step is systematic storage, analysis and processing of data in order to prepare it for solving problem of building data grids [18, 19]. The main objective of regularization stage is to build a regular grid using methods of objective analyses and estimate the accuracy of gridded data [20]. Regular grids are usually updated and provided to end-users twice a year. It is possible to organize grid recalculation each time new measurements are acquired in systems that include components for oceanographic data processing. Algorithms for grids recalculation assumes that the whole grid is processed. The recalculation takes much time, besides new data is processed equally to historical data, though it is much more important for estimation of actual water area parameters.

The experiments on operational estimation of water area parameters were made using measurements received from Argo float drifts [21]. The objective of Argo program is to operate and manage a set of floats distributed in all oceans. An Argo float drifts for a number of years in the ocean. It continuously performs measurement cycles. Each cycle lasts about 10 days and can be divided into 4 phases: a descent from surface to a defined pressure (e.g., 1500 decibars), a subsurface drift (e.g., 10 days), an ascending profile with measurements (e.g., pressure, temperature, salinity), a surface drift with data transmission to a communication satellite.

An example of Argo float trajectory, temperature and salinity profiles are given in Fig.8.



a) trajectory	b) temperature profile	c) salinity profile

Fig.8 An example of an Argo float trajectory and profiles with measurements

For operational estimation of ocean parameters each Argo buoy was considered as a system that was characterized by trajectory and a set of profiles with measurements. Each point where data transmission was fulfilled was defined as objects. For neighboring objects according to the trajectory relations were set. List of possible relations contained two types of relations: ‘measured before’, ‘measured after’. Each object was characterized by a vector of characteristics listed in table 1 and by a vector of measured parameters. The parameters were described in the form presented in table 2.

Table 1 Objects characteristics

Name	*Definition*	*Comment*
PLATFORM_NUMBER	char PLATFORM_NUMBER(N_PROF, STRING8); PLATFORM_NUMBER:long_name = "Float unique identifier"; PLATFORM_NUMBER:conventions = "WMO float identifier : A9IIIII"; PLATFORM_NUMBER:_FillValue = " ";	WMO float identifier. WMO is the World Meteorological Organization. This platform number is unique. Example : 6900045
JULD	double JULD(N_PROF); JULD:long_name = "Julian day (UTC) of the station relative to REFERENCE_DATE_TIME"; JULD:units = "days since 1950-01-01 00:00:00 UTC"; JULD:conventions = "Relative julian days with decimal part (as parts of day)"; JULD:_FillValue = 999999.;	Julian day of the profile1. The integer part represents the day, the decimal part represents the time of the profile. Date and time are in universal time coordinates. Example : 18833.8013889885 : July 25 2001 19:14:00
LATITUDE	double LATITUDE(N_PROF); LATITUDE:long_name = "Latitude of the station, best estimate"; LATITUDE:units = "degree_north"; LATITUDE:_FillValue = 99999.; LATITUDE:valid_min = -90.;	Latitude of the profile. Unit : degree north. Example : 44.4991 : 44° 29’ 56.76’’ N

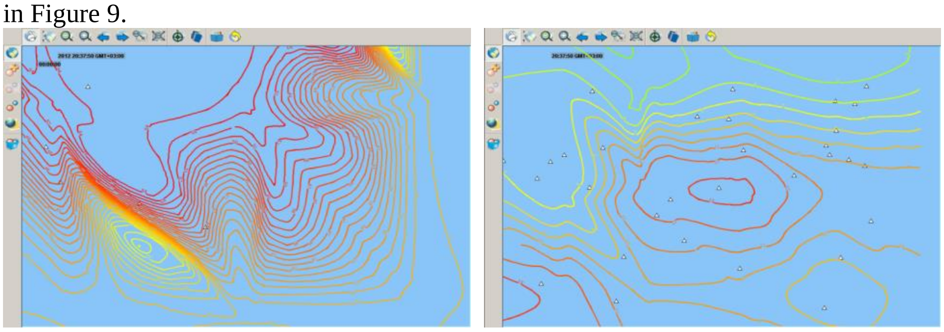
	LATITUDE:valid_max = 90.;	
LONGITUDE	double LONGITUDE(N_PROF); LONGITUDE:long_name = "Longitude of the station, best estimate"; LONGITUDE:units = "degree_east"; LONGITUDE:_FillValue = 99999.; LONGITUDE:valid_min = -180.; LONGITUDE:valid_max = 180.;	Longitude of the profile. Unit : degree east. Example : 16.7222 : 16° 43' 19.92'' E

Table 2 The description of parameters measurements

Name	*Definition*	*Comment*
<PARAM>	float <PARAM>(N_PROF, N_LEVELS); <PARAM>:long_name = "<X>"; <PARAM>:_FillValue = <X>; <PARAM>:units = "<X>"; <PARAM>:valid_min = <X>; <PARAM>:valid_max = <X>; <PARAM>:comment = "<X>"; <PARAM>:resolution = <X>;	<PARAM> contains the original values of a parameter <X> format of values representation

To provide end-users with actual information based on results of new measurements, regular grids were rebuilt for the region where new data was received. Identification of ocean regions borders can be made manually by experts of subject domain or using algorithms of cluster analyzes. Algorithms for building gridded data were extended by a preliminary step that assumed assessment of observable situation. Buoys with similar or partly similar trajectories that have close measurements values were found using algorithms for building and comparing situation graphs. Depending on distances between the analyzed and similar situations weight coefficients were assigned to measurements. The highest values were assigned to newly received measurements. When rebuilding grid weight of measurements are considered. It allows calculating ocean parameters estimations based on new data and take into account tendencies that were observed in similar situations. As not all grid is rebuild, but only region of interest, processing is executed enough fast to meet users requirements.

Examples of results of ocean data processing using proposed approach are given in Figure 9.



a) Temperature b) Salinity

Fig. 9. Measuring facilities and ocean parameters regular grids

The evaluation of the results was carried out by comparing measurements from a test set that contained 5000 temperature and salinity values for various depths measured by instruments and calculated values for the same parameters at the points with the same coordinates. The result of the comparison showed that the accuracy of calculated parameters values has increased up to 5% in some regions and in average in about 2-3%.

10 Conclusion

The application of the proposed method for situation assessment allows to take into account results of objects parameters measurements received from different sources. Recognition of situations and revealing similar situations provides possibility to obtain additional information about observed situation including tendencies and dynamics of its development. The approach to describe and compare situations using graphs provides high speed of calculations. Thus, we can say that the presented method can solve all problems considered in the paper.

Our future research is connected with developing algorithms that will allow using information about dependencies between parameters and their mutual influence.

References

1. Endsley, M.: Toward a Theory of Situation Awareness in Dynamic Systems. Human Factors, vol. 37, no 1, 32 – 64 (995)

2. Ganter, B., Wille, R.: Formal Concept Analysis: Mathematical Foundations, Springer (1999)

3. Kuznetsov, S.O.: Mathematical aspects of concept analysis. Journal of Mathematical Science, vol. 80, issue 2, 1654-1698 (1996)

4. Bunke, H., Allermann, G.: Inexact graph matching for structural pattern recognition. Pattern Recognition Letters, no. 1, 245–253 (1983)

5. Lin, J., Keogh, E., Wei L., Lonardi, S.: Experiencing SAX: a novel symbolic representation of time series. Data Mining and knowledge discovery, vol. 15, no. 2, 107‑144 (2007)

6. Lin, J., Keogh, E., Lonardi, S., Chiu, B.: A Symbolic Representation of Time Series with Implications for Streaming Algorithms. In: 8th ACM SIGMOD Workshop on Research Issues in Data Mining and Knowledge Discovery, pp. 2–11. ACM Press (2003)

7. Fedotov, Y., Hovanov, N. Complex Production Systems' Performance Measurement: Methods of Estimation Aggregate Indices. Discussion Paper #25(R)–2006. Institute of Management, Saint Petersburg State University, St. Petersburg (2006)

8. Kugiumtzis, D., Tsimpiris, A. Measures of Analysis of Time Series (MATS):A MATLAB Toolkit for Computation of Multiple Measures on Time Series Data Bases. Journal of Statistical Software, vol. 33. issue 5, 1-30 (2010)

9. Sokolov, I. S.: Method for building graph model for group telemetric signal. In: Scientific session of National Research Nuclear University MEPhI, pp. 77‑78. MEPhI, Moscow (2011)

10. Eshera, M., Fu, K.: A graph distance measure for image analysis. IEEE Transactions on Systems, Man, and Cybernetics (Part B), vol. 14, no. 3, 398–408 (1984)

11. Eshera M., Fu, K.: A similarity measure between attributed relational graphs for image analysis. In: 7th International Conference on Pattern Recognition, pp. 75–77. Springer, Heidelberg (1984)
12. Riesen, K., Neuhaus, M., Bunke, H.: Bipartite graph matching for computing the edit distance of graphs. In: 6th International Workshop on Graph Based Representations in Pattern Recognition. LNCS, vol. 2726, pp. 1–12. Springer, Heidelberg (2006)
13. Riesen, K., Bunke, H.: Approximate graph edit distance computation by means of bipartite graph matching. Image and Vision Computing, vol. 27, no. 7, 950–959 (2009)
14. Wille, R.: Conceptual structures of multicontexts. In: ICCS, Eklund, P. W., Ellis G., Mann, G. (eds.). Lecture Notes in Computer Science Series, vol. 1115, pp. 23-39. Springer, Heidelberg (1996)
15. Ganter, B., Kuznetsov, S. O.: Pattern structures and their projections. In: ICCS, Delugach H. S., Stumme G., (eds). Lecture Notes in Computer Science Series, vol. 2120, pp. 129-142. Springer, Heidelberg (2001)
16. Witten, I., Frank, E., Hall, M.: Data Mining: Practical Machine Learning Tools and Techniques. Morgan Kaufmann, third edition, San Francisco (2011)
17. Kuznetsov, S.O., Obiedkov, S.A.: Comparing Performance of Algorithms for Generating Concept Lattices. Journal of Experimental and Theoretical Artificial Intelligence, vol. 14, no. 2-3, 189-216 (2002)
18. Korablev, A. A., Pnushkov, A. V., Smirnov, A. V.: Compilation of the oceanographic database for the Nordic Seas. Journal of Arctic and Antarctic Research Institute, vol. 447, 85-108 (2007)
19. Boyer, T., Levitus, S., Garcia, H., Locarnini, R. A., Stephens, C., Antonov, J.: Objective analyses of annual, seasonal, and monthly temperature and salinity for the World Ocean on a 0.25° grid. Int. J. Climatol., vol. 25, 931–945 (2005)
20. Zhuang, S. Y., Fu, W. W., She, J.: A pre-operational three Dimensional variational data assimilation system in the North/Baltic Sea. Ocean Sci., vol. 7, 771-781 (2011)
21. http:// www.argo.ucsd.edu/