

Recognition and Analysis of Complex Structured Objects

Tatiana Kosovskaya

Abstract. Approaches to solving mathematical problems that arise in the recognition and analysis of complex structured objects are described. An approach based on the application of the predicate calculus language is considered. A question is raised related to the possibility of using the hypergraph language.

Introduction

While investigation properties of complex structured objects, which are a set of elements that have some specified properties and are in specified relationships, a convenient tool is to describe them using predicate calculus formulas [1, 2, 3].

Since the problems that arise with this approach are NP-hard, a method for constructing a multilevel description of classes of objects is described [4].

The main mathematical concept in constructing such a description is the concept of isomorphism of elementary conjunctions of predicate formulas, the verification of which is polynomially equivalent to the «open» problem *Graph Isomorphism* [5].

The notion of isomorphism of elementary conjunctions of predicate formulas allows to solve the following problems:

- construction of a multi-level description of classes [4];
 - building a fuzzy logic-predicate network [6, 7];
 - creation of ontologies in the set of CSO [8]
- and many others.

While using the predicate formulas, the relations between the elements are set by atomic predicate formulas in which the order of arguments is strictly defined. But if the relation does not depend on the order of the arguments (for example, it is commutative like the relation "to be brothers"), then it becomes necessary to write down atomic formulas with all possible orders of arguments.

It is even more difficult to write down such a property of a group of elements in which the number of elements for different groups of the same type is different. For example, a group of elements "family" can include from two up to a sufficiently large number of elements.

The use of hypergraphs for the description and analysis of such objects is discussed below.

1. Main definitions

Definition. A complex structured object (CSO) is defined as an object $\omega = \{\omega_1, \dots, \omega_n\}$, whose elements have given properties and are in specified relationships defined by predicates p_1, \dots, p_k .

Definition. A description of the complex structured object ω is an elementary conjunction $S(\omega)$ of atomic formulas with predicates p_1, \dots, p_k , which contains the maximum number of literals and is true for ω .

Let the set of all CSOs Ω be divided into disjoint subsets $\Omega_1, \dots, \Omega_M$.

Definition. A description $S(\Omega_j)$ of the set Ω_j is a disjunction of elementary conjunctions with variables for arguments

$$S(\Omega_j) = A_j^1(\bar{x}_j^1) \vee \dots \vee A_j^{k_j}(\bar{x}_j^{k_j}),$$

which is true for every object from Ω_j and only for them.

Definition. Two elementary conjunctions of atomic predicate formulas $F1(a_1, \dots, a_n)$ and $F2(b_1, \dots, b_n)$ are called isomorphic

$$F1(a_1, \dots, a_n) \sim F2(b_1, \dots, b_n),$$

if there is such an elementary conjunction $R(x_1, \dots, x_n)$ and substitutions of arguments a_{i_1}, \dots, a_{i_n} and b_{j_1}, \dots, b_{j_n} of formulas $F1(a_1, \dots, a_n)$ and $F2(b_1, \dots, b_n)$ respectively, instead of all occurrences of variables x_1, \dots, x_n of the formula $R(x_1, \dots, x_n)$, that the results of these substitutions $R(a_{i_1}, \dots, a_{i_n})$ and $R(b_{j_1}, \dots, b_{j_n})$ coincide up to the order of literals with the formulas $F1(a_1, \dots, a_n)$ and $F2(b_1, \dots, b_n)$, respectively.

The resulting substitutions $\lambda 1 = \{x_1 : a_{i_1}, \dots, x_n : a_{i_n}\}$ and $\lambda 2 = \{x_1 : b_{j_1}, \dots, x_n : b_{j_n}\}$ are called unifiers of formulas $F1(a_1, \dots, a_n)$ and $F2(b_1, \dots, b_n)$ with the formula $R(x_1, \dots, x_n)$ respectively.

Definition. An elementary conjunction that does not contain constants is called a common subformula of two elementary conjunctions if it is isomorphic to some subformulas of each the initial ones.

Definition. An elementary conjunction that does not contain constants is called a maximal common subformula if it is their common subformula with the largest number of literals.

2. Construction of a multi-level description of classes

If the description of the class of objects has the form

$$S(\Omega_j) = A_j^1(\bar{x}_j^1) \vee \dots \vee A_j^{k_j}(\bar{x}_j^{k_j}),$$

then the recognition of a CSO with the description $S(\omega)$ consists in the checking the logical sequence

$$S(\omega) \Rightarrow \exists \bar{x}_{\neq} A(\bar{x}), \quad 1$$

where $A(\bar{x})$ is a disjunctive term of the classes descriptions.

Computational complexity of such a checking is exponential under the length of the right-hand part [9]. Construction of multi-level description of classes consists in pairwise extraction of maximal common subformulas from elementary conjunctions $A_j^l(\bar{x}_j^l)$ and $A_j^r(\bar{x}_j^r)$. This procedure is repeated with the obtained elementary conjunctions until the newly received ones do not have common subformulas. The process of recognition starts from checking of (1) for the shortest of the received [4].

Computational complexity remains exponential but with essentially lower exponent.

3. Discussion of using predicate calculus formulas disadvantages

The use of predicate calculus formulas has such disadvantages that

- in cases that the relations are commutative (for example, «to be friends»), you have to explicitly specify both $p(x, y)$ and $p(y, x)$,
- if the relation can be multi-place without specifying the number of arguments (for example, «be members of the same family»), then you need to enter several predicates like $p^j(x_1, \dots, x_j)$ for each number of arguments. Question: what is the maximum number of such arguments?

The possibility of using hypergraphs to eliminate these disadvantages is suggested below.

4. Main definitions related to the use of special type hypergraphs

Definition. A set $\omega = \{\omega_1, \dots, \omega_n\}$, on which the properties of these objects and the relationships between them are set is called a complex object (CO).

For every element of a complex object ω it is known what properties are fulfilled and in what relationships these elements are.

The vertices of a hypergraph of relations are

- vertices with the names of properties and relations, which will be called **predicate vertices**;
- vertices with the names of the elements of ω , which will be called **objective vertices**.

Definition. A hypergraph of relations is such a hypergraph that its vertices are divided into two parts:

Tatiana Kosovskaya

- one part (predicate vertices) contains the names of the properties of objects and the relationships between them, the in-degree of these nodes are 0;
- the other one (objective vertices) contains the names of the elements of an object.

The edges are defined as follows:

- for a property p the edge has the form

$$p \rightarrow \omega_i ,$$

- for a noncommutative k -ary relation

$$p \rightarrow (\omega_{i_1} \rightarrow \cdots \rightarrow \omega_{i_k}) ,$$

- for a commutative relation of arbitrary dimension

$$p \rightarrow \{\omega_{i_1}, \cdots, \omega_{i_k}\} ,$$

- for a noncommutative k -ary relation, which can have one of the elements of a set specific to that position

$$p \rightarrow \{\{X_1\}, \cdots, \{X_k\}\} .$$

Definition. A description of $CO \omega$ is a hypergraph of relations, the names of properties and relations defined on ω are in one part of which, and the names of all objects are in the other part.

Let the set of all objects Ω be divided into disjoint subsets $\Omega_1, \dots, \Omega_M$.

Definition. The description $S(\Omega_j)$ of the set Ω_j is such a collection of hypergraphs with variables as the names of the vertices of the second part that there is a hypergraph isomorphic to the subgraph defining the description of the object for each object of the class in this collection.

The problem of checking whether a recognized object belongs to a given class is to check whether the class description contains a hypergraph isomorphic to a subgraph of the hypergraph that defines the description of the object.

To solve this problem, it is necessary to give a precise definition of hypergraphs of the required type, as well as to define the concept of isomorphism of such hypergraphs.

5. Example of descriptions of an object and a class of objects

Let an object be a family of Johnson. It contains 5 members: father John, mother Mary, two sons Peter and Basil, daughter Katherine.

Two properties and one relation are defined as

$m(x)$ – "x is a man",

$w(x)$ – "x is a woman",

$p(\{X\}, \{Y\})$ – "the element of the set $\{X\}$ is the parent of the element of the set $\{Y\}$ ".

In such a case, the description of the Johnson family can be represented by a bipartite hypergraph, in one part of which there are vertices m, w, p , and in the other part vertices with the names of family members J, M, P, B, K .

Short Paper Title

The edges have the form $(m, J), (m, P), (m, B), (w, M), (w, K), (p, (\{J, M\}, \{P, B, K\}))$, the last edge means that every element of the set $\{J, M\}$ is the parent of every element of the set $\{P, B, K\}$.

Description of the object class "families with brothers"¹ corresponds to the fact that the formula $\exists x x_1 x_2 (p(x, \{x_1, x_2\}) \& m(x_1) \& m(x_2))$ follows from the description of a particular family.

This formula (without quantifiers of existence) corresponds to a hypergraph with variables x, x_1, x_2 as vertex names and with edges $(m, x_1), (m, x_2), p, (x, \{x_1, x_2\})$.

It is obvious that it is a subgraph of a hypergraph defining a description of the Johnson family with the following unifiers (values of variables): $x = J, x_1 = P, x_2 = B$.

Conclusion

The article states the main provisions of the logic-predicate approach to the recognition and analysis of complex structured objects.

The disadvantages of this approach are given for the relations between the elements of objects, which may not depend on the order of the elements or on the number of elements in this relationship.

To overcome these disadvantages, it is proposed to use special-type hypergraphs, namely bipartite hypergraphs, in one part of which are the names of properties and relationships between objects, and in the other part the names of the elements of objects.

A model example of using the proposed type of bipartite hypergraphs is given to check whether an object belongs to a class with a given property.

The question remains open about the development of an algorithm for checking the isomorphism of the proposed type of bipartite hypergraphs, as well as the allocation of the largest common up to the names of the vertices of the second part of the subgraph. Such algorithms make it possible to solve the problems mentioned in the introduction.

References

- [1] D.Z. Du and K.I. Ko. "Theory of Computational Complexity." A Wiley-Interscience Publication. John Wiley & Sons, Inc. 2000.
- [2] N. Nilson, "Problem-solving methods in artificial intelligence." New York, McGRAW-HILL BOOK COMPANY Press. Publ., 1971, 280 p.
- [3] S.J. Russel and P. Norvig, "Artificial Intelligence. A Modern Approach." Pearson Education, Inc. 2003.

¹Below, the letters x, y, z are used for the names of variables for objects (perhaps with indexes), and $\{X\}, \{Y\}, \{Z\}$ (perhaps with indexes) are used for the names of sets of objects.

- [4] T.M. Kosovskaya. "An approach to the construction of a level description of classes by means of a predicate calculus language", SPIIRAS Proceedings, **3(34)**, 204 – 217 (2014). (In Russian)
- [5] T. M. Kosovskaya, and N. N. Kosovskii, Polynomial Equivalence of the Problems Predicate Formulas Isomorphism and Graph Isomorphism // VESTNIK ST. PETERSBURG UNIVERSITY, MATHEMATICS Vol. 52 No. 3, 2019. pp. 286 – 292. (In Russian)
- [6] Tatiana Kosovskaya. Fuzzy logic-predicate network. // Proceedings of the 11th Conference of the International Fuzzy Systems Association and the European Society for Fuzzy Logic and Technology (EUSFLAT 2019). Editors Martin Stepnicka. Part of series ASUM (Atlanties Studies in Uncertainty Modeling), V. 1. pp. 9 - 13.
- [7] Tatiana Kosovskaya. Fuzzy Neural Networks that Change their Configuration. // ISSN 0361-7688, Programming and Computer Software, Springer Nature, 2024, Vol. 50, Suppl.1, pp. S10 – S17.
- [8] Kosovskaya T.M., Kosovskii N.N. Extraction common properties of objects for creating logical ontologies. // Vestnik of St. Petersburg State University. Applied mathematics. Computer science. Control processes.2022. Vol. 18, issue. 1, pp. 37 — 51.
- [9] T.M. Kosovskaya Proofs of estimates of the number of steps for solving some image recognition problems with logical descriptions // Vestnik of St. Petersburg university. Mathematics. 2007, issue 4. Pp. 82 – 90.

Tatiana Kosovskaya
Faculty of Mathematics and Mechanics
St. Petersburg State University
St. Petersburg, Russia
e-mail: kosovtm@gmail.com