

H.A. PIDNEBESNA, Junior Researcher,
International Research and Training Centre
for Information Technologies and Systems of the NAS of Ukraine,
Glushkov ave., 40, Kyiv, 03187, Ukraine,
pidnebesna@ukr.net

A.V. PAVLOV, PhD (Eng), Researcher,
International Research and Training Centre
for Information Technologies and Systems of the NAS of Ukraine,
Glushkov ave., 40, Kyiv, 03187, Ukraine,
andriypavlove@gmail.com

V.S. STEPASHKO, DSci (Eng), Professor,
International Research and Training Centre
for Information Technologies and Systems of the NAS of Ukraine,
Glushkov ave., 40, Kyiv, 03187, Ukraine,
stepashko@irtc.org.ua

ONTOLOGY APPLICATION TO CONSTRUCTING INDUCTIVE MODELING TOOLS WITH INTELLIGENT INTERFACE

This paper is devoted to the analysis of sources in the field of development and building intelligent user interfaces. Particular attention is paid to presenting an ontology-based approach to constructing the architecture of the interface, the tasks arising during the development, and ways for solving them. An example of construction of the intelligent user interface is given for software tools of inductive modeling based on the detailed analysis of knowledge structures in this domain.

Keywords: *intelligent user interface; inductive modelling tools; GMDH; ontology approach; domain metamodel, human-computer interaction; user modeling.*

Introduction

The problem of constructing and implementing various types of intelligent interface between users and software means has long history, see e.g. [1–3]. A series of International Conferences on Intelligent User Interface (IUI) is held yearly; see the home page [4]. As it is noted in [5], an IUI involves the computer side having sophisticated knowledge of the domain, and/or a model of the user. These allow the interface to better understand the user's needs and personalize or guide the interaction. Another interpretation of the IUI may be done as

follows: a program that has an intelligent interface and uses intelligent techniques when interacting with the user. The IUI might be based on some user models, or on knowledge of the system functionality, or on procedures helping the user [6].

Generally, there is an idea in most sources that an intelligent user-computer interface should predict the users' intentions and present information based on that meeting their current needs. Besides, the following IUI features are eligible: adaptability to the needs of a particular user and the ability to learn new concepts and techniques.

The generation of interfaces, called "intelligent", is designed to provide the user with additional capabilities, including adaptability, customization and assistance in solving problems. The implementation of the intelligent user interface should be an intelligent agent or intermediary between the user and a particular computer application that implements approaches and methods to support communication with the user [7].

As it was mentioned in [8], there is an intention to implement the following more broad characteristics of the intelligent user interface when designing the intelligent modeling system: user-system interactive mode that tracks and supports all stages the modeling process; acquisition and active usage of the user's knowledge when performing the process; permanent monitoring, testing and correcting all decisions made by user at any stage; training the user during interaction with the system; and others.

Under the user-system interactive mode we mean the spectrum of possible mutual actions starting from the fully automated mode for a novice user to the possibility of planning the whole modeling process by a skilled user (expert). This is possible only when the intelligent interface have means for the evaluating the level of user's skills and automatic adaptation to them [8].

This paper offers a way of "intellectualization" of the inductive modeling software tools by applying an ontological approach to representing the knowledge of the domain to design a knowledge base, computing tools, and intelligent user interface.

The aims of this research is to make a survey of sources on the state-of the art in intelligent user interface (IUI) field (Section II), to explain our ontology-based approach to the IUI construction (Section III), to apply this approach to the domain of inductive modeling (Section IV), to give an example demonstrating the consequent decision making as an IUI-based modeling process (Section V) and to present relevant conclusive remarks (Section VI).

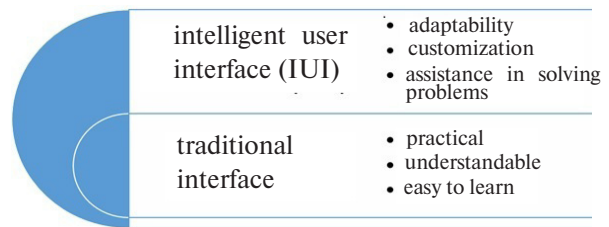


Fig. 1. IUI Characteristics

Intelligent User Interface: Characteristics and Tasks

The user interface combines all the elements and components of the program that can affect the user's interaction with the software. These include:

- a set of user tasks that he solves using the system;
- system control elements;
- navigation between system units;
- visual (and not only) design of program screens;
- means of displaying information;
- devices and technologies of data input/output;
- the procedure for using the program and documentation for it, etc.

Like traditional interfaces, intelligent user interface should be easy to learn, practical and understandable. But, in addition, intelligent interfaces promise to provide some additional features [9]:

- recognition of inaccurate, ambiguous and/or partial multimodal input of information (acquiring and processing information in the system using various devices such as mouse, keyboard, microphone);
- creating a coherent, unified and understandable multimodal representation;
- fully or partially automatic problem solving;
- interaction management (problem-solving, customization, interface adaptation) through representation, inference, and use models of user, domain, task and context.

Fig. 1 illustrates main differences between IUI and traditional interfaces.

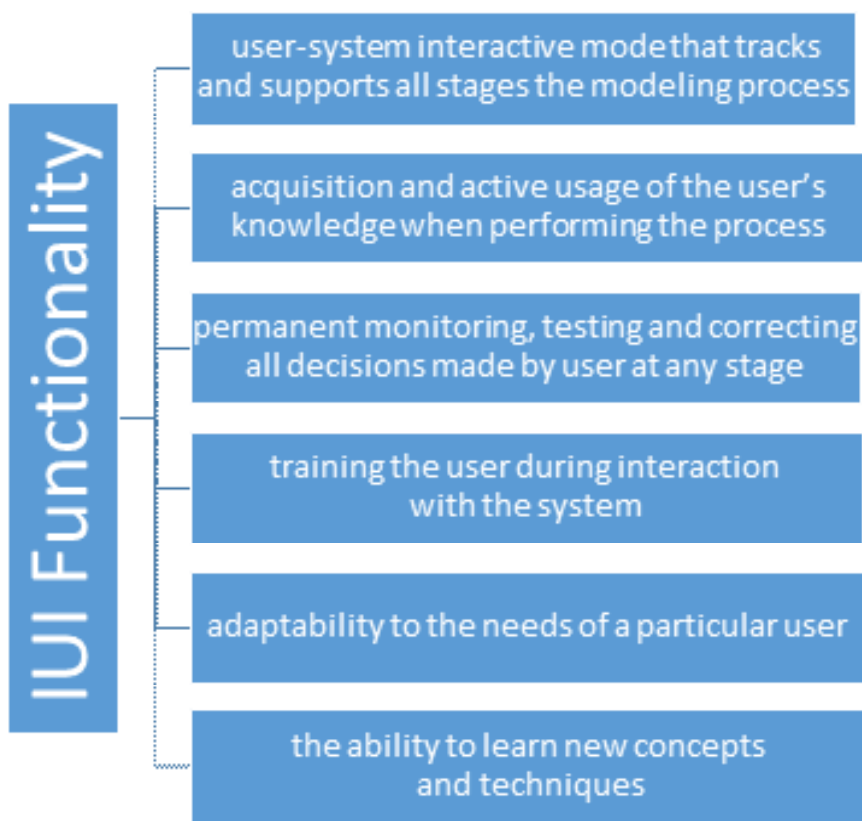


Fig. 2. IUI Functionality

In other words, "intelligence" of the user interface can be interpreted as some subroutine-agent (Fig. 2). On one hand, it monitors the user's actions, analyzes them and helps the user by suggesting options for action. On the other hand, it connects to a given software application and calls its commands according to the model of dialogue with the user.

Interaction management is the core of the IUI and is responsible for:

- interpretation and integration of recognized information from different sources based on a data transmission model;
- creating and adjusting a real-time dialogue model, advancing the dialogue with the user according to this model based on the domain model, task model, and user model;
- recognition and formation of an action plan based on the model of dialogue and past actions of the user (direct input by the user or tracking his actions);

- presentation of results or formation of clarifying questions for the purpose of elimination of ambiguities in the course of dialogue with the user.

An important characteristic of IUI is the ability to adapt the system response to the level of understanding of a particular user. This ability is based on the construction and use of the user model and inference in the direction of appropriate adaptations of the interface. The IUI adaptability includes adaptability both to the user and to the context. Adaptability to the user requires the construction of a user model, and adaptability to the context - the model of the domain, the model of the task and the model of dialogue. Ontologies [10] and rules [11] can be used to build a domain model and task models.

As well as ontologies cover conventional static knowledge of a particular field, they are also valuable for other areas of research. In natural language

applications, ontologies can be used for natural language processing or for automatic extraction of knowledge from (scientific) texts.

The initiative in the interaction between the IUI and the user can be given to the user, the system, or be mixed (fig. 3) [12]. Mixed-initiative systems support the natural alternation of user contributions and systems aimed at solving tasks. At the moment, systems with mixed-initiative are considered the most promising.

This approach declares a significant improvement in human-machine interaction, allowing the computer to behave more like a partner who is able to work with users to develop a common understanding of goals and contribute to problem-solving in the most appropriate way. Creating such complex systems requires a number of time-consuming tasks. In particular, it is necessary to develop mechanisms for collecting information, formal presentation of knowledge of the domain, drawing logical conclusions about the intentions of the user, his attention, and his abilities in conditions of uncertainty.

A dialogue is built between the user and the agent, who pursue a common goal that must be accomplished over the application. The IUI acts as the driving force behind this dialogue. Its role is to maintain and monitor the state of the dialogue and to offer the following options for the development of events moving along a certain tree of the plan, consisting of actions necessary to achieve the goal or sub-goal. The state of dialogue in the system consists of a stack of goals and an action plan in the form of an ontological model. The ontological model of the user's task specifies the objects of activity, defines the classes of terms, characteristics, and areas of possible values, states, and necessary to perform the tasks of linking objects. This is necessary to work out the functionality of applications, to build a sequence of actions that solve specific types of tasks,

Note that IUI is abstracted from the specific ways of transmitting the information, domain, software, task and the user. One of the modern approaches that allow solving the above problems and reducing the effort required for implementation is Model-Based User Interface Development

(MBUID) [11]. The purpose of this approach is to identify high-level models that allow you to define and analyze software from a more semantically oriented level. This allows you to create a complete specification of the entire interface model, with the division into components, modules, reuse, as well as allow the implementation of automatic coding of components according to the built specification of the user interface.

Principles of the Ontology-Based Approach

The use of ontologies can facilitate the description of the IUI design problem as a part of complex systems from components and implement a program that makes such a configuration independent of the product and the components itself, makes it possible to reuse.

The advantage of an ontological approach is that ontology defines a conceptual structured environment in which the process of constructing a model of an automation object takes place.

Ontology is the exact specification of some field that contains a glossary of terms and a set of domain links describing relations between these terms. It actually is a hierarchical conceptual skeleton of domain.

Formal ontology model (O) is an ordered triplet

$$O = \langle T, R, F \rangle,$$

where T is finite set of terms of the domain being described by the ontology O ; R is a finite set of relations between terms of the given domain; F is a finite set of the interpretation functions given on the terms and/or relations of the ontology O .

Additional information about classes and their relationship to each other, as well as restrictions on the value of attributes for each class are fixed in axioms. Axioms are sometimes segregated in a separate (fourth) set A , then the ontology is given by the quadruple $O = \langle T, R, F, A \rangle$. In the classical sense, they are part of a set of interpretation functions.

As a rule, ontologies are classified by levels of generality [10]:

- Top-level ontologies (meta-ontologies, representational ontologies) do not link to any specific

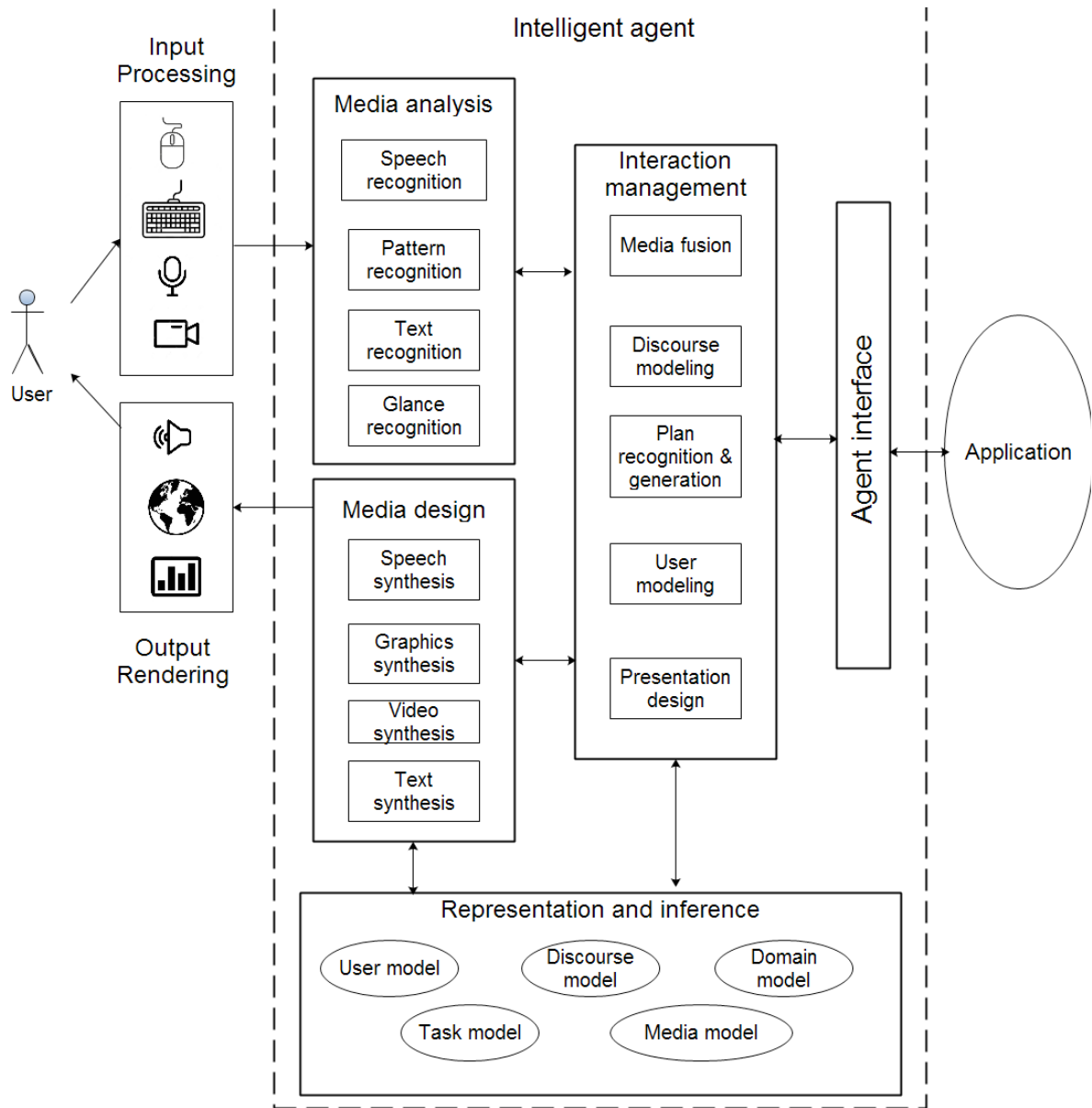


Fig. 3. IUI Architecture

domain, operate with general concepts and relationships that do not depend on the subject area. Such ontologies provide representative entities without specifying what should be represented.

- Domain ontologies cover the knowledge that applies to a particular type of domain, consists of concepts that describe a particular subject area, significant relationships in it, the set of these concepts and relationships. If ontologies operate in several

domains, they are called generic or core ontologies, as well as super theories.

- Application ontologies (task ontology, applied ontology) contain all the necessary knowledge to model a specific process (usually a combination of domain and method ontologies), where the concepts are the types of problems to be solved and the relationships are the decomposition of these tasks into subtasks.

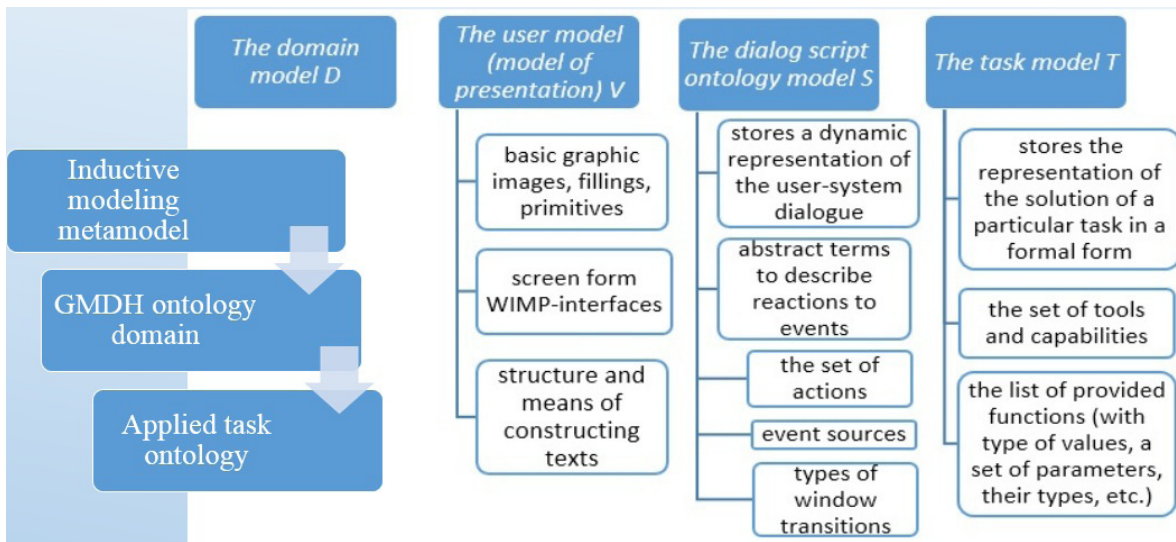


Fig. 4. Types of ontological models used for IUI

The main principles of the ontological approach to designing the user interface are [13]:

1. To separate the development and modification of the user interface from the application under the rules of their interaction.

2. To combine content-homogeneous interface information into separate components. For example, you can select several basic systems of concepts in the user interface:

- a system of user concepts through which the user interacts with the system;
- developer concept system, which should contain concepts to describe the structure and means to display information in the interface, describe the scenario of the dialogue and the implementation of the interaction between the application and the user interface.

3. To present the components of the user interface in the form of declarative models formed on the basis of universal ontologies describing each component.

The user interface is the part of the software package that, given the wide range of potential users, can often adapt, taking into account different levels of training, a wide range of tasks, and personal preferences. Implementation of the user-

friendly interface, meeting these requirements, is possible by building layouts of the future interface, based on the use of universal ontological models of the user interface.

There are several types of ontological models, and each type plays a different role in the process of software interaction with the user (Fig. 4).

The IUI architecture includes next ontological models:

- *The domain model D* that represents the characteristics of the specific area, defines concepts, characteristics, and areas of possible values.

- *The user model V (ontology of presentation)* that collects, stores and updates knowledge about the user so that the IUI effectively meets his requirements.

- *The dialog script ontology model S* that:

1. stores a dynamic representation of the user-system dialogue;
2. defines: abstract terms to describe reactions to events; the set of actions that occur when events occur; event sources; types of window transitions; and how to select window instances etc.

- *The task model T* that:

1. stores the representation of the solution of a particular task in a formal form;

2. determines: the set of tools and capabilities provided by the application; the list of provided functions that are characterized by the type of values, a set of parameters, their types, etc.

Then, the generation of the interface model component is reduced to setting values of the corresponding model concepts of universal ontology $\{D, V, S, T\}$. This representation allows implementing one of the most important features of the IUI, namely adaptability.

Ontology for the Domain of the GMDH-based Inductive Modeling

The task model and the domain model of the system can be developed using ontologies of different levels. The automation of tasks for intelligent modeling systems requires simulation of the modeling process, which is a meta-modeling. A metamodel is a model that describes the structure and principles of other models' operation.

A metamodel provides a logical level of the domain and is interpreted dynamically at the application level. This adds additional flexibility to the system, since the domain logic can be changed without modifying the code. To allow or disallow a type of communication at the logical level, it is enough to assign it only to the meta-model formal terms.

In fact, the metamodel is defined by a high-level ontology, in terms of concepts of solution methods, key stages, and constraints. The ontological model of the domain at the lower level describes the algorithmic components of each particular modeling method in more details. To solve a practical problem, an ontological model of the task is used. This model has its own parameters, specific characteristics and tolerance ranges.

Designing an ontology for a specific domain requires in-depth analysis, identification of relevant concepts, attributes, relationships, constraints, instances and axioms of this domain. Such analysis of knowledge leads to systematization, construction of a hierarchy of concepts with their attributes, values and relations.

The results of the analysis and structuring of the inductive modeling area help to design the GMDH

meta-ontology [14]. It contains the basic components of the modelling process. The main principles of generating its characteristics are determined.

When formalizing knowledge, different types of ontologies are used to solve problems – these are the so-called ontologies of methods and tasks ([15]). Ontologies of problems defining specific problems contain terms and relations for specific tasks, and ontologies of methods do terms that are specific to specific methods of solving problems. Ontologies of methods and tasks allow to clearly defining the interaction between problem solving and knowledge of domains.

Inductive modeling is a model generation process based on the analysis and generalization of given statistical data obtained through observations or experiments. The inductive modeling methods provide building mathematical models of objects and processes to solve a range of tasks:

- time series forecasting;
- classification (supervised learning);
- clusterization (unsupervised learning or self-training: identification of effective features, forms and rules of distinction); this problem is called “Objective Computer Cauterization” (OCC) in GMDH field;
- determining the set of independent (inputs), dependent (outputs), and irrelevant (uninformative) variables among the measured ones with the aim to build an adequate model of the system; this problem is called “Objective System Analysis” (OSA) in GMDH field.

A number of methods are used to solve different types of problems. In cases where it is necessary to model objects, parametric GMDH algorithms have proved to be effective. In cases of modeling objects with fuzzy characteristics, it is more effective to use nonparametric GMDH algorithms, in which polynomial models are replaced by a sample of data divided into intervals or clusters. Algorithms of this type completely solve the problem of eliminating the bias of coefficient estimates [16].

The ontology of methods specifies a set of GMDH algorithms.

Parametric:

- the basic *Combinatorial* (COMBI) algorithm is based on full or reduced sorting-out of

gradually complicated models and evaluation of them by external criterion on separate part of data sample;

- *Multilayered Iteration* algorithm (MIA) uses at each layer of sorting procedure the same partial description (iteration rule). It should be used when it is needed to handle a big number of variables;

- *Objective System Analysis* (OSA) algorithm. The key feature of it is that it examine not single equations but systems of algebraic or difference equations obtained by implicit templates (without goal function). An advantage of the algorithm is that the number of regressors is increased consequently, the information embedded in the data sample is utilized better;

- *Two-level* (ARIMAD) algorithm for modeling of long-term cyclic (such as stock or weather) processes. There are used systems of polynomial or difference equations for identification of models on two time scales and then choice of the best pair of models by external criterion value.

Non-parametric algorithms are exemplified by:

- *Objective Computer Clusterization* (OCC) algorithm that operates with pairs of closely spaced sample points [5]. It finds physical clusterization that would as possible be the same on two sub-samples;

- *“Pointing Finger”* (PF) algorithm for the search of physical clusterization. It is implemented by construction of two hierarchical clustering trees and estimation by the balance criterion;

- *Analogues Complexing* (AC) algorithm, which use the set of analogues instead of models and clusterizations [8]. It is recommended for the fuzziest objects.

Inductive modeling is a process of directed model generation and selection, consisting of a finite set of successive stages. All methods of inductive modeling have standard components. Those can be used as a basis to form a metamodel of inductive modeling.

The principles of algorithmic modules formation for solving a specific problem can be determined as a result of the domain structuring. Depending on the task type, a relevant method of solving it can be selected.

Structuring the Successive Process of Inductive Modeling

A step-by-step approach is applied to solve problems of modeling processes and objects from the given data set. This approach is a consecutive selection of «the best» solutions from a set of possible options under condition of existence of the target solution. An algorithm builds a tree of actions and puts questions based on the knowledge about the current action and a given goal. The peculiarity of the algorithm is that it does not produce the whole plan of modeling in detail but only a part of it. The plan is formed in more details in the course of the modeling process (Fig. 5).

The process of any real-world problem solution can be characterized by the following stages:

1. *Data preparation*: creation and maintenance of data files or databases, and their pre-processing (smoothing, gaps handling, outlier filtering, scaling, centering, normalization).

2. *Task preparation*: determination of the problem type (static data or real-time data modeling), the purpose of modeling (approximation, interpolation, extrapolation, trend selection, etc.), and primary data analysis (determining statistical characteristics, correlation, trend and oscillatory properties).

3. *Task definition*: determination of the object class (linear or nonlinear, static or dynamic, stationary or nonstationary etc.) and selection of the basic functions class (polynomials, trigonometric series, difference equations etc.).

4. *Modeling problem statement*: choosing the model quality criterion, the type of the model structure generator, and the parameter estimating method.

5. *Solving the stated problem*: the actual modeling procedure is performed based on the given data.

Inductive modeling is a process of sequential decision making consisting of a certain set of successive stages. Therefore, we can decompose the solution of the relevant modeling method selection problem into certain sub-problems to solve them individually in a sequential manner (Fig. 6) [17].

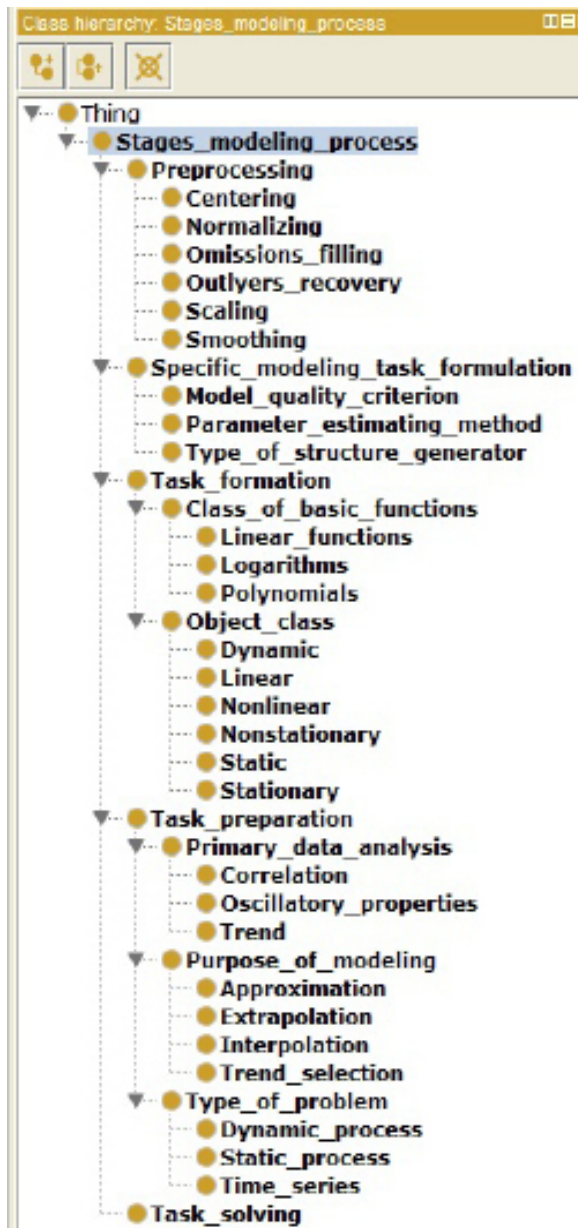


Fig. 5. Fragment of the ontology describing main stages and methods used throughout the inductive modeling process

Suppose, for example, that the data is pre-processed and does not contain gaps, outliers, and so on. Then, the appropriate sequence of decision-making stages, taking into account the results of this report, will be as follows:

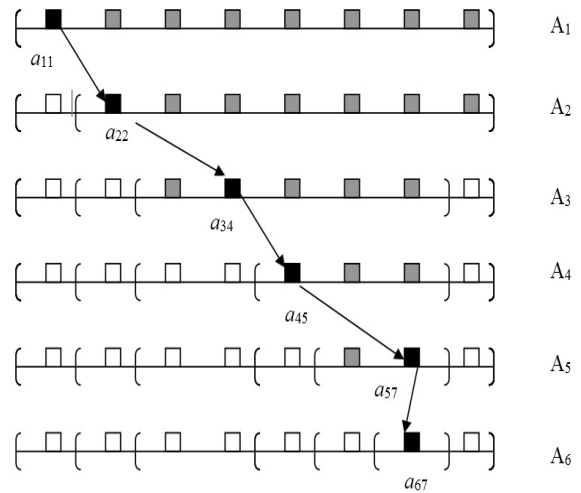


Fig. 6. Decomposition of the process of sequential decision making

1. Choice of the *modeling purpose* (approximation, interpolation, extrapolation, trend determination, construction of an input-output model etc.).
2. Determination of the *process type* (linear static object, non-linear static object, linear time series, non-linear time series, linear dynamic process, nonlinear dynamic process).
3. Determination of *process stationarity* (stationary, with increasing, decreasing, or oscillatory trend, sum of trends).
4. Choice of the *model class* (linear regression, autoregression, autoregression with the trend, harmonic, logarithmic, polynomial or exponential functions of time, difference equations etc.).
5. Choice of the *external criterion* for the model selection (Fisher criterion, Akaike criterion, Mallows C_p -statistics, jack-knife, unbiasedness, regularity criterion etc.).
6. Choice of the *parameter estimation* method (LSM, LMM, ridge regression etc.).
7. Choice of the *structure generation* method (a given structure, nested structures, inclusion, exclusion, stepwise, branch and bound, exhaustive search, combinatorial, combinatorial-selective, multilayered, relaxation, genetic etc.).

8. The *model validation* (Fisher statistics, R^2 criterion, accuracy on a validation data set etc.)

At each of these stages, one of a number of possible decisions (done by experts, based on the literature or experience) is made, following which one can move to the next stage. If one uses the decomposition of the proposed stages, then, depending on the decisions made at the previous stages, the set of potential solutions will be significantly reduced further. This is because the decision at each stage imposes implicit restrictions on the application of certain methods required at the subsequent stages.

In general, it is clear that the proposed decomposition of the process into stages and, in turn, the stages into basic and auxiliary solutions are also the structure of an intellectual dialogue. Its application in the IUI decision-making system is a condition that the user could get a model of the studied object using the modeling system, regardless of the level of his/her training, a priori knowledge of the object and the initial data. Evidently, the more prepared is the user and the more information he/she holds on the object or process, the more adequate model will be built.

It might be noted that even a user who firstly try to solve a modeling task using such system can always get maybe a simple but still satisfactory model applying the automatic mode of the system.

Conclusion

In recent decades, we can distinguish two different views on the progress and development of intelligent user interfaces. The first group of scientists is passionate with the development of methods of presentation and mechanisms of logical inference evaluation and prediction of the attention and intentions of the user. The second group of researchers focuses on the development of automated tools and models aimed at improving the user's ability to work directly with the application. We believe that their symbiosis can offer a significantly new level of use of systems, which is characterized by deeper and more natural human-computer cooperation.

The paper presents the principles of building a user interface based on ontology models in the development of inductive modeling systems. This approach makes it possible to simplify the design of efficient, user-friendly interfaces of different levels of training, which have the characteristics of an intelligent interface. We believe that the proposed approach provides an advantage due to the high level of the formalism of ontologies, their independence from languages and programming tools, the ability to make changes in the structure of the interface independently of other parts of an applied program, automatic coding based on ontologies.

REFERENCES

1. *Intelligent Interfaces Theory, Research, and Design*. P.A. Hancock and M.H. Chignell (eds), North Holland, New York, 1989.
2. *Kolski, C., Strugeon, E.Le.*, 1998. A review of "intelligent" human-machine interfaces in the light of the ARCH model, *Int. J. of Human-Computer Interaction*, 10 (3), pp. 193–231.
3. *Rogers, Y., Sharp, H., Preece, J.*, 2011. *Interaction Design: Beyond Human-Computer Interaction* (3rd ed), Wiley, Chichester.
4. *Welcome to ACM IUI 2020!* [online] Available at: <<https://iui.acm.org/2020/>> [Accessed 20 Feb. 2020].
5. *Sonntag, D.*, 2012. Collaborative Multimodality, *Kunstliche Intelligenz*, Springer, 26 (2), pp. 161–168, DOI: 10.1007/s13218-012-0169-4, <http://iui.acm.org/2018/> Last accessed 2019/19/22.
6. *Intelligent User Interfaces*. [online] Available at: <<https://web.cs.wpi.edu/Research/airg/IntInt/intint-outline.html>> [Accessed 29 Dec 2019].

7. *Introduction to Model-Based User Interfaces*. [online] Available at: <<https://www.w3.org/TR/2014/NOTE-mbui-intro-20140107>> [Accessed 22 March 2020].
8. *Stepashko, V.*, 2019. On the Self-organizing Induction-Based Intelligent Modeling, *Advances in Intelligent Systems and Computing III* / N. Shakhovska, M.O. Medykovsky, Editors, AISC book series, Cham: Springer, 871, pp. 433–448.
9. *Intelligent User Interfaces: An Introduction*, Morgan Kaufmann, RUIU, San Francisco, 1998, pp. 1–13.
10. *Gruber, T.*, 1995. Toward principles for the design of ontologies used for knowledge sharing, *Int. J. Human-Computer Studies*, 43(5-6), pp. 907–928.
11. *Ahmad, A.-R., Basir, O., Hassanein, Kh.*, 2004. Adaptive User Interfaces for Intelligent E-Learning: Issues and Trends, *Proc. ICEB'2004*, pp. 925–934.
12. *M. Maybury*, 1998, *Intelligent User Interfaces: An Introduction*, RUIU, San Francisco: Morgan Kaufmann, pp. 1- 13.
13. *Pidnebesna, H.*, 2014. An ontological approach to user interface design for inductive modeling systems, *Inductive modeling of complex systems*, Kyiv: IRTC ITS NASU, 6, pp. 117–125 (In Ukrainian).
14. *Pidnebesna, H., Stepashko, V.*, 2018. On Construction of Inductive Modeling Ontology as a Metamodel of the Subject Field, *Int. Conf. Advanced Computer Information Technologies (ACIT-2018)*, Ceske Budejovice, University of South Bohemia, pp. 71–74.
15. *Studer, R., Benjamins, V.R., Fensel, D.*, 1998. Knowledge Engineering: Principles and methods, *Data & Knowledge Engineering*, 25, pp. 161–197.
16. *Spectrum of GMDH algorithms*. [online] Available at: <http://www.gmdh.net/GMDH_alg.htm> [Accessed 22 Sept. 2020].
17. *Stepashko, V.S., Zvorygina, T.F.* 2003. On an approach to the decision inference problem in an intricate task, *CSC № 6*, С. 82-87 (In Russian).

Received 17.08.2020

Г.А. Піднебесна, молодший науковий співробітник,
Міжнародний науково-навчальний центр
інформаційних технологій та систем НАН та МОН України,
просп. Глушкова, 40, Київ 03187, Україна,
pidnebesna@ukr.net

А.В. Павлов, к.т.н., науковий співробітник,
Міжнародний науково-навчальний центр
інформаційних технологій та систем НАН та МОН України,
просп. Глушкова, 40, Київ 03187, Україна,
andriypavlove@gmail.com

В.С. Степашко, д.т.н., професор,
Міжнародний науково-навчальний центр
інформаційних технологій та систем НАН та МОН України,
просп. Глушкова, 40, Київ 03187, Україна,
stepashko@irtc.org.ua

ОНТОЛОГІЧНИЙ ПІДХІД ДО ПОБУДОВИ ІНТЕЛЕКТУАЛЬНОГО ІНТЕРФЕЙСУ КОРИСТУВАЧА В ІНСТРУМЕНТАЛЬНИХ ЗАСОБАХ ІНДУКТИВНОГО МОДЕЛЮВАННЯ

Вступ. Генерація інтерфейсів, яка називається "інтелектуальною", призначена для надання користувачеві додаткових можливостей, включаючи адаптивність, налаштування на конкретного користувача та інтерактивну допомогу у вирішенні проблем. Інтелектуальний інтерфейс користувача має бути інтелектуальним агентом або посередником між користувачем і певним комп'ютерним додатком, що реалізує підходи та методи підтримки комунікації з користувачем.

Мета статті. У роботі пропонується підхід до "інтелектуалізації" програмних засобів індуктивного моделювання на основі онтологічних концепцій подання знань про предметну галузь для проектування бази знань, обчислювальних інструментів та інтелектуального інтерфейсу користувача. Цілі цього дослідження: виконати огляд джерел про сучасні засади конструювання інтелектуального інтерфейсу користувача (ІІК), пояснити наш підхід до побудови ІІК, заснований на онтології, застосувати цей підхід до галузі індуктивного моделювання, а також навести приклад, що демонструє подальше прийняття рішень як процес моделювання на основі ІІК та подати відповідні прикінцеві зауваження.

Результати. Розглянуто сучасні підходи до організації інтелектуального користувацького інтерфейсу. Описано характеристики і завдання ІІК, принципи онтологічного підходу до його конструювання та функціонування.

Стан діалогу в системі визначається набором цілей та планом дій у формі онтологічної моделі завдань користувача, яка визначає об'єкти діяльності, класи термінів і характеристик та області їхніх можливих значень і станів, необхідних для виконання завдань зв'язування об'єктів. Це необхідно для відпрацювання функціональних можливостей додатків, побудови послідовності дій, що вирішують конкретні типи завдань. Важливо брати до уваги, що *IUI* абстрагується від конкретних способів передачі інформації, домену, програмного забезпечення, завдання та користувача.

Результати аналізу і структурування знань загалом у галузі індуктивного моделювання є базисом для розроблення метаонтології конкретної предметної галузі індуктивного моделювання на основі МГУА. Така онтологія містить основні компоненти процесу моделювання. Визначено основні принципи формування його характеристик. Представлено фрагмент онтології домену індуктивного моделювання.

Показано, що розкладання процесу розв'язання будь-якої реальної задачі на етапи і, у свою чергу, етапів на основні та допоміжні рішення сприяє формуванню структури інтелектуального діалогу. Застосування його в системі прийняття рішень ІІК є умовою того, щоб користувач міг отримати модель об'єкта дослідження за допомогою відповідної системи моделювання, незалежно від рівня його/її підготовки, апріорних знань про об'єкт та початкових даних.

Висновки. У статті представлено принципи побудови інтерфейсу користувача на основі онтологічних моделей при конструюванні систем індуктивного моделювання. Такий підхід дозволяє спростити розроблення ефективних і зручних інтерфейсів для різних рівнів підготовки користувача, які мають інтелектуальні характеристики. Ми вважаємо, що запропонований підхід забезпечує перевагу завдяки високому рівню формалізму онтологій, незалежності від мов і засобів програмування, можливості вносити зміни в структуру інтерфейсу незалежно від інших частин прикладних програм, автоматичного кодування на основі онтологій.

Ключові слова: *інтелектуальний інтерфейс користувача; засоби індуктивного моделювання; МГУА; онтологічний підхід; метамодель предметної галузі, взаємодія людина-комп'ютер.*