Modelling of the integrated waste management systems using object-oriented methodology

KRZYSZTOF GASKA, KRZYSZTOF PIKOŃ
Chair of Technologies and Installations for Waste Management
Silesian University of Technology
ul. Konarskiego 18   44-100 Gliwice
POLAND
http://www.polsl.pl

Abstract

The paper presents the adaptive, reactive computer assistance system design methodology, making modelling, analysis, and testing possible of the integrated waste management systems. The presented model is defined in a multilayer open architecture, i.e.-, it consists of the logical spaces of the constituent models (submodels) of the object classes and real processes of the waste management system, implemented as the program code developed in the high level programming languages (RAD). The waste management system model was developed using a state-of-the-art approach to the design of algorithms for modelling, analysis, optimization, and testing of the real elementary processes of the integrated systems. Topics covered by the paper include elements of a literature review on the modelling of systems and waste management processes, review of the programming environments employed, as well as a description of the object-oriented waste management systems design methodology. An example of one of the developed subsystems is also included at the end of the text. Computer simulation results are also presented, carried out for one of the processes for the purposes of the model calibration and validation. The simulation – confirmed a high conformity of the model with the real processes.

Key-Words: waste management, object-oriented methodology, validation, environment, modeling of elementary processes.

1   Introduction

Applying the object modelling methodology to real waste management systems allows computer decision support systems to be designed for use in the planning of integrated waste management systems [11][12]. Because of the complexity of models describing the various elements of the systems connected with waste disposal, and also with analysis of their effect on all natural environment elements (soil environment, surface and underground waters, atmospheric air, acoustic climate) due to the migration chemical and microbiological pollutants from technological processes, the object-oriented design methodology was adopted, based on high level programming languages (Rapid Development Development – RAD) C++, Object Pascal with elements of the symbollic machine language (Assembler). The essence of these programming environments is the use of object types (classes) to define the complex data structures, characteristic for the modelled systems.The design presented in the paper is characteristic of a different attitude to modelling and analysis of waste management systems, using novel methods to design algorithms for the simulation, optimisation [5], testing, validation, and verification of object-oriented models of

the integrated management systems [15]. One of the new developments is the method of multivariant analysis of waste management systems solutions using graph algorithms [2] and genetic algorithms. Another new solution is the algorithm for transforming the graphical part of the digital map into the graph structure used in the optimisation processes [1]. The new algorithms made it possible to develop many solutions in the area of object-oriented design and analysis of waste management systems.

2   Theory of the waste management systems

A waste management system is defined as a set of elements (objects, processes) linked by relationships. Figure 1 presents the schematic diagram of the integrated municipal, industrial and hazardous wastes management system decomposed into subsystems. The list of the elementary processes of one of the subsystems is presented as an example. Waste type was assumed as the criterion of system decomposition. Each subsystem is subjected to further decomposition taking into account

its constituent processes (waste acquisition and storage, waste processing and disposal, logistic processes of waste transportation). The decomposition of the globals waste management system is aimed at separation the so-called subsystems, which – following the assumption that they are not linked by equations of constraints – makes it possible to selectively analyze the particular elementary processes defined within these subsystems. Moreover, separation of the integrated system into its constituent elements makes their analysis easier and is of key importance during the design of the mathematical model and – in consequence – also of the numerical one.

- predictive (forecasting changes of the shape of processes, predicting changes of the parameters of processes, forecasting changes in waste generation),
- analysis of migration of pollutants from the technological processes of waste processing and disposal,
- analysis, identification and classification of environmental hazards,
- modelling of processes, elementary objects,
- modelling of the geo-environmental data.

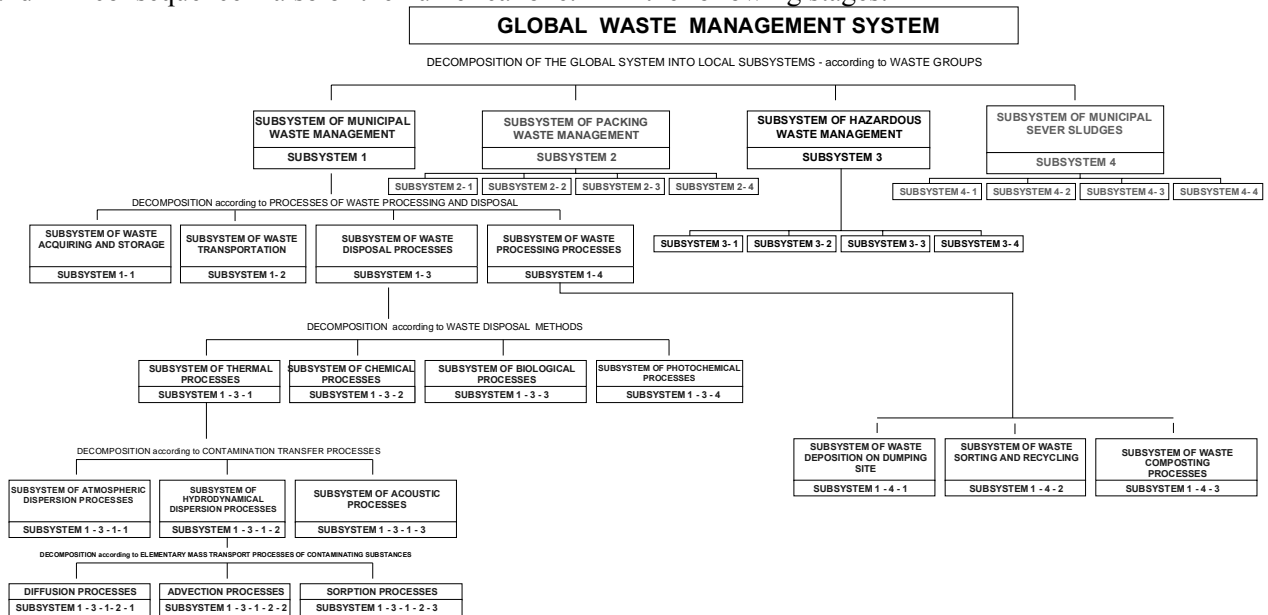The general systems modelling problem encompasses the following stages:

Fig.1. Decomposition schema of the integrated waste management system

3 Theory of the modelling of waste management systems

The modelling of waste management systems is connected with simulation of the real elementary processes of which waste management is composed. Because of the complexity of the subject, dedicated software packages are in use nowadays featuring decision support systems for solving many complex problems in the area of design, analysis and optimization of single processes.

Modelling an integrated waste management system involves the following problems[4]:

- planning of the management systems variants for wastes of all groups,
- management of processes, systems, subsystems,
- analysis of the constituent processes of systems,
- optimization of the elementary processes of the waste management system (process control, scheduling of tasks),
- investigation of correlations among the particular system elements and processes,
- identification of data structures identifying the system elements or processes,

1. Development of the conceptual model
2. Development of the continuous mathematical model,
3. Development of the discrete mathematical model, deterministic (with known functional relationships by discretization of the ordinary/partial differential equations defined in the continuous model)[13]
4. Development of the numerical model (basic one), structural
5. Development of the reactive, adaptive object-oriented model
6. Verification of the model
7. Calibration of the model parameters (estimation of parameters)[7]
8. Validation of the model (evaluation of conformity of the model with the real system/process)
9. Testing of the object model
10. Analysis of the model stability and sensitivity.

4 Model of the waste management system

Simulation of real waste management systems, encompassing all constituent processes consists in representation of the occurring physical phenomena as

mathematical relationships describing roughly the nature of the particular processes [10]. Because of the complex nature of the analysed problems the numerical model was designed using an object-oriented methodology, as a computer implementation of data sets and mathematical equations, in software code developed with high-level languages. The definition of the object model is preceded by a number of operations connected with the specification of the general assumptions (conceptual model) and formulation of the model topology, the way in which it interacts with the environment, system dynamics, state variables, etc. The following chapters present a description of the elemets of the project of an object-oriented model of a real waste management system .

and Object Pascal high level languages implementing most of the object paradigms[6]: definition of classes, data encapsulation, inheritance (multiple inheritance), and implementations of the polymorphic methods. The open topology of the model makes modification of the software code possible within the particular classes defined within the system framework, realizing the specialisation process of the model modules. The model topology (Fig.2) encompasses the logical space of models of classes of the particular elements and processes of the waste management systems.

Computer implementation consists in formulating the equations as software code in the C++, Object Pascal (Delphi 5 environment) and Assembler languages. The developed model contains the definition of the optimization algorithms based on the multicriteria
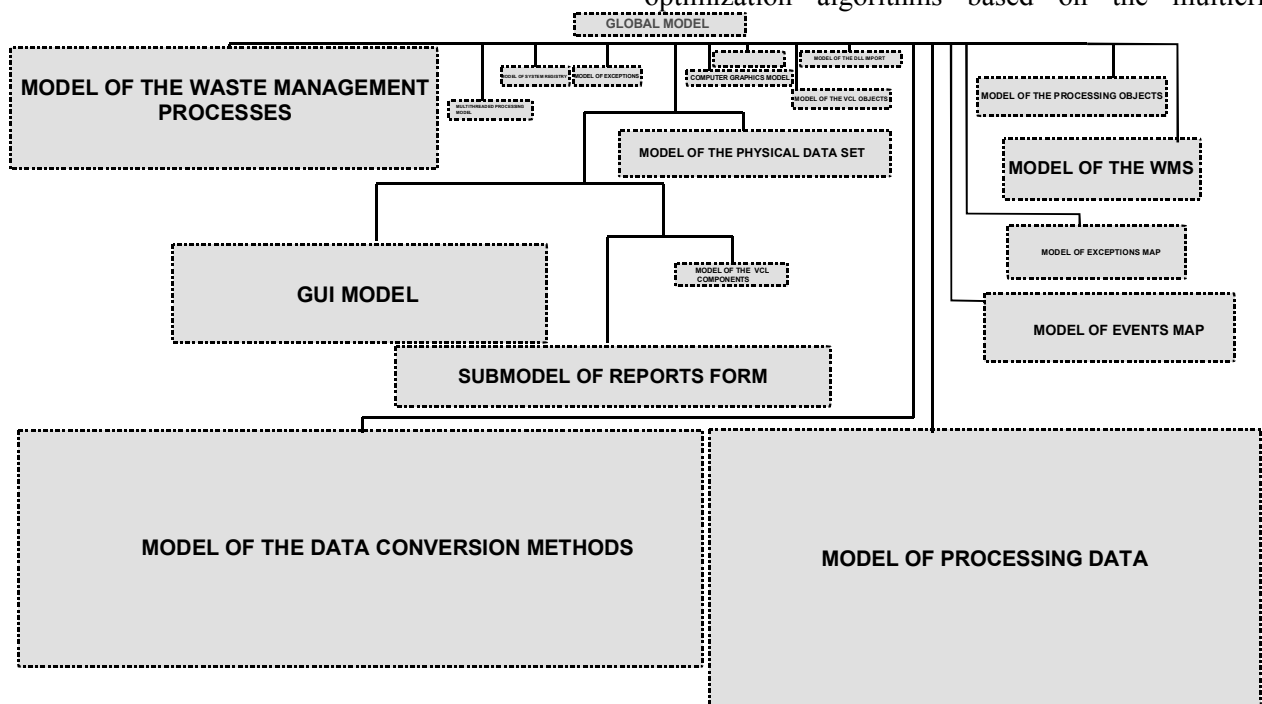
Fig.2. General schema of the model of waste management system.

4.1   Conceptual model

The concept of the model is based on using the object-oriented design and analysis methodology simulate the real elementary (constituent) processes of waste management. The global waste management system was decomposed into the so-called subsystems corresponding to the analysed processes and groups of the generated waste. The conceptual mode of waste management includes the global model definition, developed using the object-oriented methodololy, based on the RAD (Rapid Application Development) programming environments, and submodels obtained by decomposition of the general model, taking into account the specific featires of the modelled problems. The object-oriented model was designed in the multilayer open architecture in the RAD technology using the C++

method [9] of searching the space of feasible solutions, making possible the planning of the variants of solutions of the waste management system by means of cost-benefit analysis, but also the ecological analysis of the effect of the particular processes of waste processing and disposal on all elements of the natural environment, by analysis of distribution of the contamination with chemical and biological substances from point, linear, and area emission sources, as well as the distribution of the acoustic field. Therefore, the formulated criteria encompass the definition of the global goal function, defined as the net present value ratio (NPVR) evaluated based on the money flow, and local functions, defined as the admissible concentrations of the contamination substances in the atmospheric air, soil environment, surface and ground waters. The optimized model

parameters were formulated as: localisation coordinates of the particular elements-objects in the system, coordinates of the nodes of the paths determining the routes of the means of transport (travelling salesman algorithm, Floyd and Dijkstra algorithm)[2], elementary processes control parameters (task queuing and allocation, allocation control taking transport into account), parameters of technological processes affecting the emission amount of the polluting substances, parameters of the load distribution (using the greedy approximation algorithm implementing the so-called vertex cover problem in a graph).

4.2 Mathematical model

The mathematical model of the integrated waste management system was decomposed into classes of submodels featuring the representation of the elementary processes of waste processing and disposal, according to Fig. 1. The definition of submodels includes the class of:
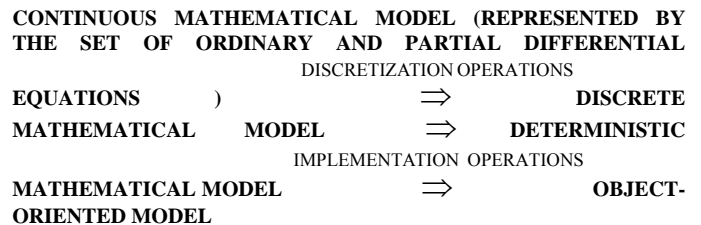- submodels of the waste acquisition and storage process,
- submodels of the process of waste deposition on the dumping site,
- submodels of the waste sorting and recycling process,
- submodels of the waste composting process,
- submodels of the process of waste disposal methods (thermal-, biological-, chemical-, photo-chemical methods, etc),
- submodels of the logistic process of waste transportation,

and the class of the so-called micromodels associated with the processes above:
- submodel of the atmospheric dispersion of pollution,
- submodel of the hydro-dynamical dispersion allowing for the diffusion, advection, and sorption processes,
- submodels of the contaminating substances migration in soil, air, ground and surface waters.

Definition of each of the above-mentioned submodels of the global model of the waste management system includes the set of data structures uniquely identifying the particular model and the set of methods in the form of a data processing algorithms. In the case of the continuous models/submodels the ordinary (partial) differential equations are used to formulate the methods; whereas, in the case of the discrete models, difference equations are used, obtained by discretization of time and space. The data structure sets describing the particular elements of the waste management system were defined separately for each of the submodels, taking into account the specific features of the described processes. Matrices were used (as the base of the static and dynamic tables, specific for the relational database) in view of investigating into the modelling issues using

the GIS geographic information systems represented by digital maps (consisting of the graphical and database parts), in which all data describing the system elements, localised in the graphical part of the map, were represented as georelational database tables. Employment of graph structures to describe particular models or submodels was dictated by the use of discrete algorithms in the optimization processes and in the design methodology of the numerical object models.

4.3 Object-oriented model (global)

The discrete, deterministic mathematical model is the base for the development of the object model, formulated using the known functional relationships, obtained by discretization of the ordinary and partial differential equations of the continuous model. The object model is the result of the following sequence of transformations:

CONTINUOUS MATHEMATICAL MODEL (REPRESENTED BY THE SET OF ORDINARY AND PARTIAL DIFFERENTIAL DISCRETIZATION OPERATIONS **EQUATIONS)** $\Rightarrow$ **DISCRETE MATHEMATICAL MODEL** $\Rightarrow$ **DETERMINISTIC** IMPLEMENTATION OPERATIONS **MATHEMATICAL MODEL** $\Rightarrow$ **OBJECT-ORIENTED MODEL**

The global (object) system model encompasses the general definitions of the base classes and of the base methods forming the base for defining the derived classes. Figure 3 presents the topology of the object-oriented waste management system allowing for the elements of hierarchical inheritance of fields and properties. The presented schematic diagram of classes is the result of specialization of the general model of the waste management system (Fig.2). The system model topology consists of three logically linked parts, featuring the representation of the analytical methods (the abstract part of the model), graphical user interface, and the relational database. The definition of an acyclic graph of classes (Fig.3) was used to represent the hierarchical structure of the model classes, their relationships, and inheritance structure. A description of the models of classes of the multilayer open architecture of the waste management system is presented in Table 2. The presented schema of classes (the acyclic graph of the model classes) has an open architecture, resulting in the possibility of specialization of the particular classes by modification of the class code, or by creating derived classes, thus expanding the base class library with elements making it possible to enhance the possibilities of the model.
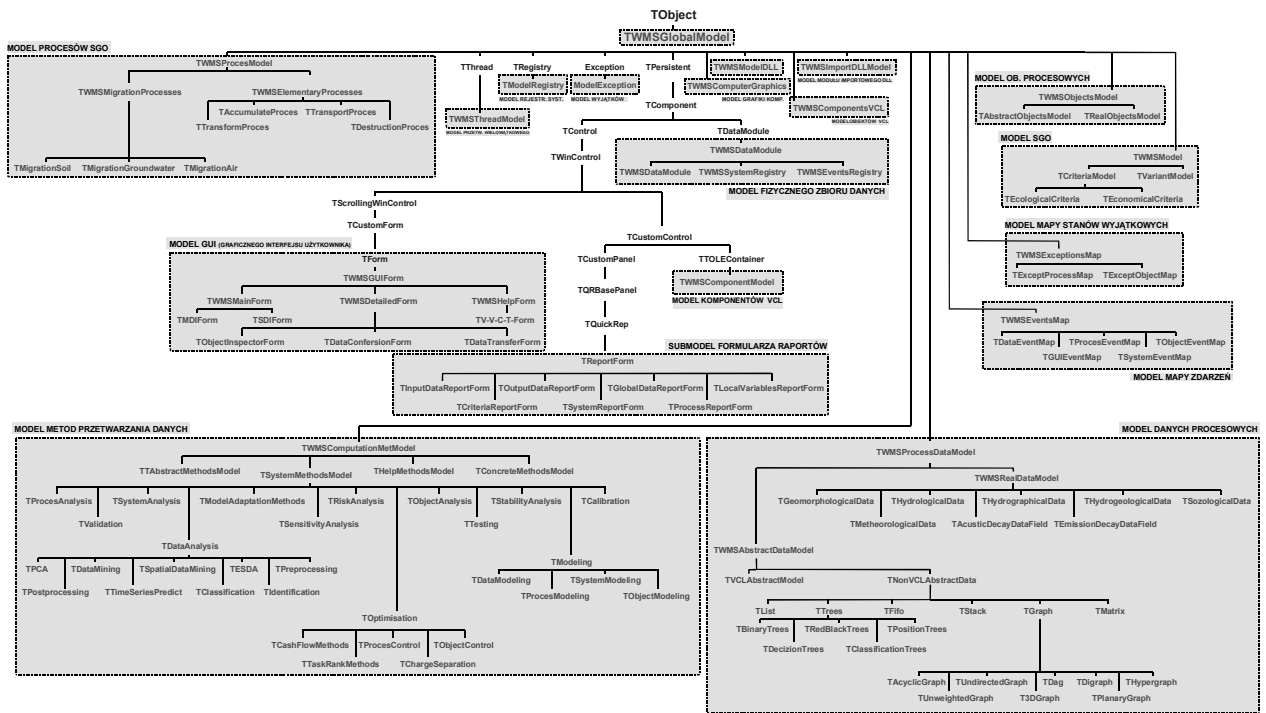
Fig. 3. Acyclic class graph of the object-oriented waste management system

4.3.1 Description of the model

The model is an adaptive, reactive (event driven) system and was defined in a multilayer open architecture, i.e., it consists of the logical spaces of the constituent models (submodels) of the object classes and real processes of the waste management system, implemented digitally as the program code developed in the environments of the high level programming languages (RAD). The core of the model are the modules whose software implementation includes grouping of definitions of data types and methods for the particular submodels or micromodels Because of the object nature of the model a number of methods were defined making it possible to carry out many complex operations on the model data structures. The model has a modular structure, with the following elements:

- Module with the definition of abstract classes (high and low level of data abstraction), data models (model of the geological data, model of the hydrogeological-, hydrological-, hydrographical-, and meteorological data, data model of pollution emission, data model describin the system objects and elementary processes),
- Module with the classes definition of the models of processes (contamination migration process, waste deposition process on the dumping site, thermal waste transformation process, etc.)
- Module with the definition of classes of th real system objects (class of the objects models – waste generation sources, class of the object models of waste disposal, etc.),
- Module with the definition of classes for the transformation methods models of the abstract data

structures (class of the exploratory data analysis methods, class of the exploratory spatial data analysis methods, etc.)
- Module with the definition of the classes of the models of testing and verification methods, as well as of the object system stability analysis,
- Module with the definition of the user interface model classes,
- Module with the definition of classes for the visualisation and simulation of process models,
- Module with the definition of classes of the state map model of the particular objects and processes.

4.3.2 General data structures

Data types were defined based on data sets describing the discrete, deterministic mathematical model of the particular submodels of the elementary processes and system elements. The general data model on which definitions are based of the specialized templates included the definition of simple types (simple, string, pointer, variant, procedural) and complex ones – structural (set, file, interface, class reference, array, record, class), as well as of the polymorphous and abstract ones. Delphi environment includes the predefined data types based on which the data types are defined characterizing the analyzed constituent processes of the waste management system.

4.3.3 Models of the general (standard) data structures processing methods

Processing methods of data structures

Data processing methods feature the direct implementation of the algebraic expressions defined in the mathematical model, in the form of the source code of the Object Pascal language (Delphi environment). Their main classification includes: static methods, virtual and dynamic methods, abstract methods, constructors, destructors, event methods, exception methods and class methods. A detailed classification was carried out within the methods presented above, taking into account the specific operations performed on data sets of particular simulation models of processes, i.e.:

- Data processing methods (data access methods, data type conversion methods, assignment methods, data update and testing methods, methods predefined in Math module),
- Methods used to perform operations on the physical data set (Dataset methods) and Input/Output methods.

Data management methods

Data management from the level of the model developed in Delphi (C++ Builder) environments is performed through methods defined in the TDataset base class and derived classes, making possible many operations on data. The database design was done under the Microsoft Access program, and all operations on the physical data file are carried out on the software code level of the Delphi application though the database alias using the ODBC standard. The set of model database tables created under Microsoft Access, is represented by the TDataModule class type, making it possible to group the non-visual elements of the data type edition (TSession, TSessionList, TDatabase, TTable, TQuery, TStoredProc, and TBatchMove). This solution was chosen because of the necessity to store the data structures in the mass memory of the computer system. Data access methods are realized using the predefined methods in the TDataSet class. Saving the data and results to the database table is carried out by the method – procedure InsertRecord (list of variables).

Visualisation methods od data structures elements

Data structures describing the model were defined using data types predefined in the Delphi environment. Data sets representing models of the real processes were defined based on structural types (class and record types, types of static and dynamic tables), and simple types. Classes of the particular submodels are polymorphous types serving as templates for the derived, specialized classes. The object data model includes definitions of:

1. Global variables, representing components of the abstract objects, nonvisual (classes of the processes models), as well as components of the visual objects of the GUI graphical user interface in the source code module of the main application form,
2. Local and auxiliary variables defined in the constituent program modules,

3. Variables representing access components to data saved in the physical database of the application model.

Visualisation is connected with elements of presentation of the relational database data structures, storing the results of the simulation of real processes. Access to data is made possible by methods predefined in the TDataset class of the integrated Delphi environment, which is the base class of all software components, responsible for data management processes.

4.3.4 Design methodology of the object-oriented model

The methodology connected with the development of a model in the object technology includes the following stages:

- Development of the concept and of the detailed design requirements of the system model (conceptual model),
- Analysis and verification of the model assumptions,
- Verification and analysis of correctness of the formulated model assumptions,
- Prototyping the source code of classes of models/submodels (processes and data),
- Definition of class templates,
- Definition of the event system (event maps),
- Definition of the event handling methods,
- Defintion of the exception maps,
- Definition of exception handling methods,
- Definition of method templates,
- Specialization of method templates,
- Definition of the global and local variables of the application model,
- Initialization methods of class objects (constructors and destructors),
- Definition of the application modules,
- Definition of the data modules – interface to the model database,
- Relational database design,
- Definition of the user graphical interface,
- Definition of the main form of the application model,
- Definition of the dialogue windows forms,
- Analysis of the time complexity,
- Analysis of model stability,
- Analysis of model sensitivity,
- Verification of the software code,
- Calibration of the model parameters (estimation of model parameters),
- Validation of the model,
- Testing of the object model,

The model was developed in the modular technology consisting in grouping the source code of the object of the particular classes in separate modules, according to the Delphi environment convention, i.e.:

- module with the definition of the visual and nonvisual elements of the main application form,
- modules corresponding to the constituent forms, reports,
- data modules, correlated with the physical database of the system model,

modules with the definition of the nonvisual elements, i.e., abstract classes performing operations on the data structures of the models (submodels).

5 Design of models

Design of models or submodels of the constituent elements of the global model of the waste management system includes in each case stages specified in chapter 4.3. that is from the stage of formulating the deterministic mathematical model, through the computer implementation stages to the object-oriented model. The base class for all defined submodels is the TWMSGlobalModel object type, which in turn inherits all elements of the superior Tobject class, predefined in the standard Object Pascal environment class library (the Delphi package).

5.1 Abstract models of data processing methods

Abstract, general models of data structures processing methods constitute the so-called auxiliary models base supporting operations on data structures of complex models, i.e. models of processes and,- objects, and include:

- Models of calibration methods of the model parameters,
- Models of verification methods of software code,
- Models of validation methods of models/submodels,
- Models of exploratory data analysis,
- Data Mining models,
- Models of the spatial data mining methods,
- Model of data preparation (preprocessing model),
- Graph models (with definition of the graph transformation and plotting methods)[3]
- Models of geometrical transformations,
- Model of calibration of the raster images of the digital maps.

The model of the abstract methods of data structures processing is represented by the TWMSComputationMetModel class type, which is anderived type with respect to the TWMSGlobalModel global class (Fig3). The presented diagram is a specialization of the class template by a set of types with definitions of the derived models.

5.2 Data model (submodel)

The data submodel was defined based on the sets of parameters describing the individual constituent processes of the waste management system. It is represented by the TWMSProcessDataModel class type and includes definitions of abstract data structures like graphs, trees, and definitions of the real data, characterizing the real elementary processes of the waste management system. The data models were formulated using the standards assumed in the digital maps (GIS) design. The following data models were defined in the waste management model:

- Model of the atmospherical data
- Model of the geomorphological data
- Model of the hydrological data
- Model of the hydrographical data
- Model of the acoustic field distribution data
- Model of the sozologic data

Each model is represented by a separate class type containing the definition of fields and properties describing parameters of the particular model and the set of methods making it possible to perform the main operations on the fields of the class object. The particular class types were defined as polymorphous structures making it possible to define derived types, extending in this way the base class library with new elements, specific for the defined derived types.

5.3 Models of the Graphical User Interface (GUI)

The graphical user interface model was designed using the visual components of the Delphi environment (GUI) so that it would represent logically the structure of classes and data types defined in the program. Models of the real processes are represented by graphical elements of the so-called main forms of the application.The main application form includes sections (so-called tabs) corresponding to the particular constituent models and each tab of the main application form is represented by a separated class type. The tabs in turn represent groups of a number of visual data edition components (editing windows) corresponding to the particular structural fields defined in the object model classes for the particular processes. Therefore, each tab features the specific visualisation of the class type representing the particular models/submodels.

5.4 Models of processes

Models of the elementary processes, constituents of waste management systems were differentiated into:

- Model of waste transportation process
- Model of the waste processing processes
- Model of the waste disposal processes

The design of the particular models includes the

definition of classes (separately for each process model - submodel) grouped in the separate modules of the computer program. Because of the necessity of analysing processes connected with, e.g., migration of the chemical substances contamination occurring during the technological processes, the mechanisms were employed enabling the modification of classes definitions as transfer methods – the import of functions and procedures defined in the DLL modules of programming environments (Neurosolution, Microstation GeoEngineering) performing the elementary operations on the data structures of the models of contamination dispersion in soils, waters, and in the air.

5.5   The application model

The application model consists (according to notation assumed in Delphi environment) of a number of functional parts called modules (Units).   Modules contain definition of data structures types and methods of processing this data in Object Pascal language code. Grouping of methods and data in separate modules allows source code to be analyzed in more detail and modified more easily. Software modules were logically divided into three classes:

1. Class of abstract modules (not correlated with the GUI graphical user interface forms) – containing definitions of data and methods of the simulated processes models,
2. Class of modules correlated with the GUI elements,

Class of modules containing the definition of the model database management methods.

6   Example of waste processing modelling

The model of the fuel component preparation process

definition of mass transfer algebraic equations, definition of modified simplex optimization algorithm).

Separated combustible species (plastics, paper, rubber, textiles, wood), as well as those obtained from agricultural processing (straw, sunflower waste, sugar beets and their derivatives, other biological and animal-originated waste), together with fossil fuels (hard coal and its low-quality fractions, brown coal, peat) are mixed in appropriate proportions to acquire fuel usable in the production of energy and cement. However, the mixing process must be carried out in a way to obtain a product with required and beforehand established, technical parameters such as specified gross calorific value, yet with the lowest volume of noxious elements possible (chlorine, sulphur, metals).

The optimization problem of the fuel formation process from selected combustible fractions of waste consists of the maximization of the objective function which is defined (formulated) in standard form [8][14] :

$$\max_{u \in S} W_d^T u , \quad S = \left\{ u \in R^n : Au \le b_{max}, Au \ge b_{min}, u \ge 0 \right\}$$

$$(1)$$

where: W_d^T - denotes $|j|$ dimension vector (number of fractions which take part in fuel component formation process) of components (w_j), denoting lower calorific values of formed fuel components. Therefore optimization process resolves to linear programming problem (linear objective function and linear limitations) with limitations Eq.(1) imposed on decision variables determined by vector U, of components (u_j), determining ipso facto the set of permissible solutions. Because the solution of the linear programming problem is located only at the top of the set of permissible solutions, the searching method should revise the tops (in space of permissible solutions). However this is not a
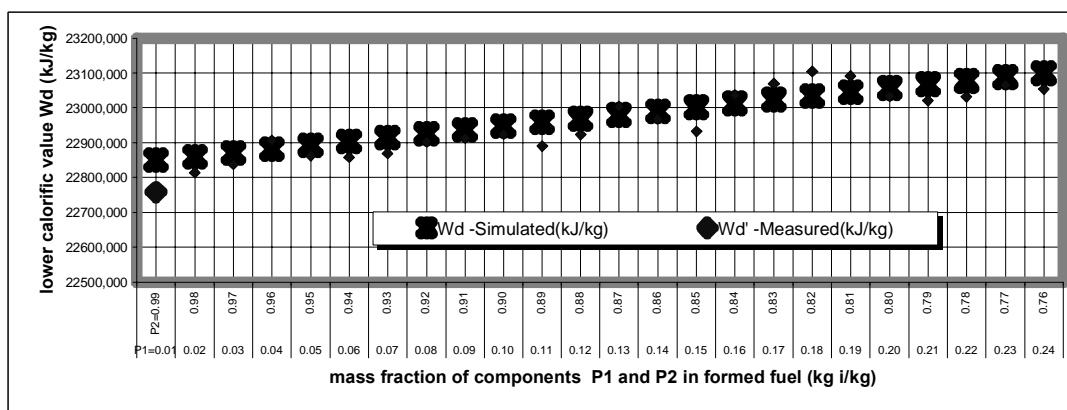

Fig.4. Comparison between measured and simulated data of lower calorific value Wd.

consists of data sets (parameters) uniquely identifying the process which define the model, and data conversion methods (definition of variables describing the process,

complete revision, but a greedy strategy in which a modified algorithm of simplexes is used. The calculation procedures outlined above are embodied in a numerical model programmed in the Object Pascal (Borland Delphi

5.0) high-level object-oriented language, with a graphical Windows-based user interface. In order to verify numerical model (described in the paper) and to

Table 1. Mass composition of components P1 and P2

Fuel components	Mass fractions in components P1 and P2 (kg i/kg)							
	c	s	h	o	n	w	p	cl
P1 (coal)	0.60350	0.00270	0.04100	0.08970	0.00010	0.07140	0.18420	0.00740
P2 (biomass)	0.52010	0.00720	0.06960	0.21860	0.13580	0.00000	0.04710	0.00160

Table 2. Constraints for optimization procedure

	Mass fraction in formed fuel (kg i/kg)							
	c	s	h	o	n	w	p	cl
Criterium		<0.02						<0.003

validate its computer code, sample problems were simulated and the results were compared with the analytical solutions. Validation of the model wass carried out with the following assumptions:

1. For the fuel formation process two components are used: dark coal and biomass of the chemical composition given in table 1.

2. Optimisation constraints imposed on fuel formation process are used as given in table 2.

Results of laboratory tests of the formed fuel and its components as well as results of computer simulations and the comparisons between simulated results and measured data are presented in Fig. 4.

As a result of computer simulations and laboratory experiments the model gives the results that are very close to the obtained from the real process. The acceptable agreement between our simulation results and the measured data demonstrates that the model can reasonably simulate the real process of fuel formation.

7 Conclusion

The model presented in this paper was developed as a reactive and adaptive system with a multilayer open architecture, making it possible to modify the source code of the class definitions and methods representing the submodels of the decomposed global model. The model presented in this work was developed based on the object design and analysis methods introducing elements of the object analysis technique of systems with very complex data structures displaying the following properties:

1. the system is modelled and developed as a set of objects communicating with each other, implemented in software by special object types whose definition includes both data and mtheods allowing many complex operations to be carried out (based on given algorithms),

2. separation of the program into objects simplifies its analysis: it is a fact that "inside" of an object one still

deals with the classical imperative model; however, from the "outside" it is to handle the objects in such a way that the program structure will resemble the structure of functional program ,

3. the design procedure assumes declarative features by using the dynamical properties of the object structures, i.e., polymorphism, encapsulation, and impementation of methods in the object structure,

4. using objects as elements of the system model ensures that its modification will be easier changes to data (fields) are mostly local within one object class

5. the so-called specialization of classes is possible by inheriting fields and methods, and declaring new properties in the derived objects.

Object representation of the waste management system required the following notations to be used:

1. objects of similar properties form a so-called class of objects, which is also a certain object structure, and the object itself is its realisation,

2. data sets forming the object structure are called fields; procedures and functions are called methods,

3. the dynamic properties of the system are realized by the methods of object interaction, i.e., methods (procedures) carrying out posting messages (defined by variables) and methods responding to the messages sent to the object. Messages are generated by the so-called events induced , for example, -by change of the object state, that is by change of the values of the object's fields, and are identified by integer constants (sent in records with the general predefined type included in the Object Pascal programming environment used in the present work). Implementation of the waste management system presented in this paper was carried out based on the open architecture.

References:

[1] Abou Najm M. , El-Fadel M. ,2004. Computer-based interface for an integrated solid waste management optimization model. Environmental Modelling & Software 19, 1151–1164

[2] Cormen Thomas H., Leiserson Charles E., Rivest Ronald L., 1999. Introduction to algorithms. WNT, Warszawa.

[3] DiBatista G., 1999. Advances in the theory and

practice of graph drawing. Theoretical Computer Science 217, 235-254

[4] Dijkema G.P.J. , Reuter M.A. , Verhoef E.V., 2000. A new paradigm for waste management. Waste Management 20, 633-638

[5] Duan Q. Y. , Gupta V. K. , Sorooshian S., 1993. Schufled Complex Evolution Approach for Effective and Efficient Global Minimisation. Journal of optimization theory and applications 76 (3)

[6] Dumnicki R., Kasprzyk A., Kozłowski M., 1998. Object Oriented Analysis and Design. Helion. Gliwice.

[7] Finley J.R., Pint´er J.D., Satish M.G. ,1998. Automatic model Calibration Applying Global Optimisation Techniques. Environmental Modeling and Assessment 3, 117–126

[8] Gaska, K., Wandrasz, A.J., Mathematical modeling of biomass fuels formation process, Waste Management(2007),
doi:10.1016/j.wasman.2007.03.025

[9] Hokkanen, J., Salminen, P., 1997. Choosing a solid waste management system using multicriteria decision analysis. European Journal of Operational Research(98) 19–36.

[10] MacDonald, M., 1996a. Solid Waste Management models: a state of the art review. Journal of solid waste technology and management 23 (2),73–83.

[11] McDougall, F., White, P., Franke, M., Hindle, P., 2001. Integrated Solid Waste Management: A Life Cycle Inventory. Blackwell Science,London.

[12] Morrissey A.J. , Browne J., 2004. Waste Management Models and Their Application to Sustainable Waste Management. Waste Management 24, 297–308

[13] Ross Keneth A., Wright Charles R. B., Discrete Mathematics, PWN, Warszawa 1999,

[14] Sysło M. M., 1999. Discrete Optimization Methods. PWN. Warszawa.

[15] White, P., Franke, M., Hindle, P., 1995. Integrated Solid Waste Management. Blackie Academic & Professional, London.