PhD THESES

KAPOSVAR UNIVERSITY

FACULTY OF ECONOMIC SCIENCES

DOCTORAL SCHOOL OF

ECONOMIC AND REGIONAL SCIENCES

Department of Information Technology

Head of Doctoral School

PROF. DR. UDOVECZ GÁBOR

Doctor of Hungarian Academy of Sciences

Supervisor

DR. HABIL. CSUKÁS BÉLA

Associate Professor

ECONOMIC OPTIMIZATION OF
SUSTAINABLE COMPLEX PROCESSES

Author:

MÓNIKA VARGA

KAPOSVÁR

2009

1. BACKGROUND AND OBJECTIVES

Computer model based process design is motivated by the increasing demand for quality products, by the strong competition in the market and by the strict environmental regulations.

One of the major difficulties in the computer aided process optimization is that the cost parameters (specific costs, prices, demand, and supply), determining the economic goal functions are uncertain variables. The usual paradoxical situation is that the technological data are better known, than the economic parameters. In the course of optimal process design, we have to choose the possibly "best" solutions from the alternatives of various structures and parameters, but the evaluation of these best solutions essentially depends on the uncertain parameters. In the present economic situation it is almost impossible to fix these parameters. That's why new optimization under uncertainty approaches are required.

The following objectives have been determined for my study:

• A methodology has to be developed for testing of the technological alternatives, generated by the optimizing algorithm, for the prescribed uncertain ranges of the economic parameters. Accordingly, a two criteria optimization has to be solved, when the two objectives tend to select the maximal contribution margin, and the solutions, that are insensitive for the uncertain parameters, respectively.

• The optimizing methodology of our research group, combining the generic simulator and the genetic algorithm, has to be extended by the uncertainty testing module.

• An extended GraphViz based expert and user interface has to be developed for the declaration of the parameters, input data and local program codes, associated with the building elements. In this development

the description file of the open source Graphviz has to be supplied for the description of data structures and brief programs, that can be interpreted by the general Prolog kernel of the simulator. The planned solution has to be support the automatic generation of the expert and user modules.

- The previously mentioned development has to be tested in the example of the economic optimalization of a complex recycling system, involving food industrial and agricultural processes. The example is based on the existing and planned system of the Kaposvar Sugar Factory of Hungarian Sugar Ltd.

2. MATERIALS AND METHODS

In the present dissertation new methods are developed and investigated, that can extend with new capabilities of some existing methods and programs in a general way. These existing programs are the following:
- the dynamic simulator based on Direct Computer Mapping,
- the multiobjective, discrete/continuous genetic algorithm, and
- the open source GraphViz graphical structure representation package.

In the engineering research the methodological development cannot be based on simplified examples only. In contrary, the methodological research can definitely be supported by the investigation of a complex enough, realistic, or at least a reality based case study. Considering this, the optimization of a sustainable complex recycling process, involving food industrial and agricultural components, focusing on the utilization of sugar beet slice, has been chosen for testing. The actual background of this choice was the preparation for a respective proposal. Although, the planned work

hasn't been realized, the approximate analysis of the given realistic processes gives an alive example.

3. RESULTS AND DISCUSSION

3.1 Owerview of the studied class of the problems and of the developed methods

3.1.1 Hierarchic levels of the investigated task

The hierarchic levels of the investigated task are illustrated in the scheme on Fig. 3.1.1-1. The gray color highlighted layer refers to the investigation of the complex technological alternatives (B). The possible solutions can be generated from the upper level possibility space (superstructure C), while the economic optimalization is determined by the consideration of the uncertain economic parameters and functionalities of outer layer (D).

The upper level stoichiometric models of layer B are determined by the detailed model of the given subprocesses, consequently in the case of keynote elements these detailed models also have to be analyzed.

3.1.2 Overview of the methodological developments

A new methodology for optimization under uncertain economic parameters has been elaborated, where all of the candidate alternatives, generated by the optimization tool, are tested with another algorithm, which selects those "good enough" solutions that are insensitive for the uncertain economic data. The new methodology has been implemented in the collaborating system of dynamic simulator and genetic algorithm, elaborated by our research group.

Figure 3.1.1-1: Hierarchic levels of the investigated task

Figure 3.1.2-1: Schematic illustration of the new developments in the optimizing tool

In the present work the available tools have been combined by a new extension of the open source GraphViz package, developed in this work. Accordingly, the usual structure generating capabilities of the GraphViz have been extended by the Prolog-like declarations, making possible the automatic generation of the simulation models. The major developments of the present dissertation are highlighted by blue colour in Fig. 3.1.2-1.

3.2 Detailed description of the developed general methods

3.2.1 Method for the consideration of the uncertain economic parameters

To solve the optimization under uncertainty problems the collaborating system of the simulator and optimator has been extended with an uncertainty testing module.

The functioning of the uncertainty testing module is based on the previously described ranges of the uncertain economic parameters. The simulation program randomly selects from these ranges, and the calculation of the actual alternatives is based on the calculation of the average contribution margin and on the standard deviation of the contribution margins, calculated from the randomly chosen uncertain parameter sets. The uncertainty management can be followed by the circled numbers in Fig. 3.1.2-1. First, with the knowledge of the possibility space, the genetic algorithm proposes various technological alternatives for the simulator (1). Next the uncertainty testing module multiplies the simulation with various, randomly selected sets of economic parameters, chosen from the user defined ranges (2). For example, a given alternative is calculated 20 times, with different, randomly selected economic data.

The contribution margin is calculated for every simulation. Having finished the simulation and evaluation of the given alternatives with various

parameters, the average value, as well as the standard deviation of the contribution margin is calculated (3). The two-criteria genetic evolution tends to select the good enough solutions, characterized by high average and low standard deviation of the contribution margin (4).

The new methodology for the optimization under uncertainty differs from the published methods, that are based on the calculation of the uncertain model. In contrary, in the present work the deterministic model is perturbed with the consideration of the continuous or discrete uncertain economic parameters.

The contribution margin (F) is calculated by the expression

$$F = \sum_{i=1}^{n} (A_i - K_{v_i}) \tag{1}$$

where:

A_i – is the revenue of the ith technological partprocess in the actual alternative, Ft/simulated period;

K_{vi} – is the concerning variable cost, Ft/simulated period.

It is to be noted, that all of the costs associated with the new partprocesses, embedded into the process, are considered as concerning variable cost.

The optimization is controlled by two objectives. One of them is the average contribution margin of the uncertainly evaluated set of perturbed variants ($\bar{\bar{F}}$):

$$\bar{\bar{F}} = \frac{\sum_{j=1}^{n} F_j}{n} \tag{2}$$

where:

F_j – the contribution margin of the jth perturbed alternative, Ft/simulated period

n – the cardinality of the set (e.g. 20).

The second objective is the standard deviation of the contribution margins (s_F):

$$s_F = \sqrt{\frac{\Sigma(F_j - \bar{F})^2}{n-1}} \qquad (3)$$

where:

F_j – the contribution margin of the jth perturbed alternative, Ft/simulated period

$\bar{F}$ – the average contribution margin, Ft/simulated period

n – the cardinality of the alternatives.

3.2.2 Extension of the GraphViz description language with the declarations of the model generating data and programs

In the applied modeling methodology the structure of the various processes is described by the state and transition elements, as well as by the connections between them. There are two kinds of connections that can be distinguished by the nodes, connected with the directed edges. Namely, the state (ellipse) → transition (rectangle) edges determine reading and modifying, while the transition (rectangle) → state (ellipse) edges refer to modifying only. The respective connections are specified in the structure describing text file.

The GraphViz program, in line with the graphical representation of the process graphs, prepares also an XML-like text file. This file can be

extended with the textual description of the model generating information. The contribution of the model generating declaration of Prolog syntax doesn't cause interpretation problems in the graphical representation of the text file. At the same time, the local variables and the relative input/output flow patterns in the unification of the declarative logical program make possible the free local description of the actual brief program codes, communicating via the standardized list of functors in the dynamic partitions. In the present work, the GraphViz description has been supplemented by the Prolog-like, typical declarations, making possible the description of the input data, conditions, program codes and consequences. This extended text file makes possible the automatic generation and execution of the complex process models by the kernel of the dynamic simulator.

To understand the model generating capabilities of the extended GraphViz language, a typical example can be seen in Fig. 3.2.2.-1. In the scheme the conventional elements of the GraphViz language cannot be seen.

The general declarations, containing the model generating information are the followings:

In the declaration of the **transition elements** (e.g. a_i in Fig. 3.2.2-1) the *prototype* determines the type of the given state element. Defining prototypes makes easier the description of the similar standardized elements, because the reference to the prototype replaces for the detailed description of the model generating *data*, *program*, *input* and *output*.

The flag *yn* defines the existence of the given state element in the genetic development. The simple transcription of the flag in the generated database makes possible the selection between the possible alternatives.
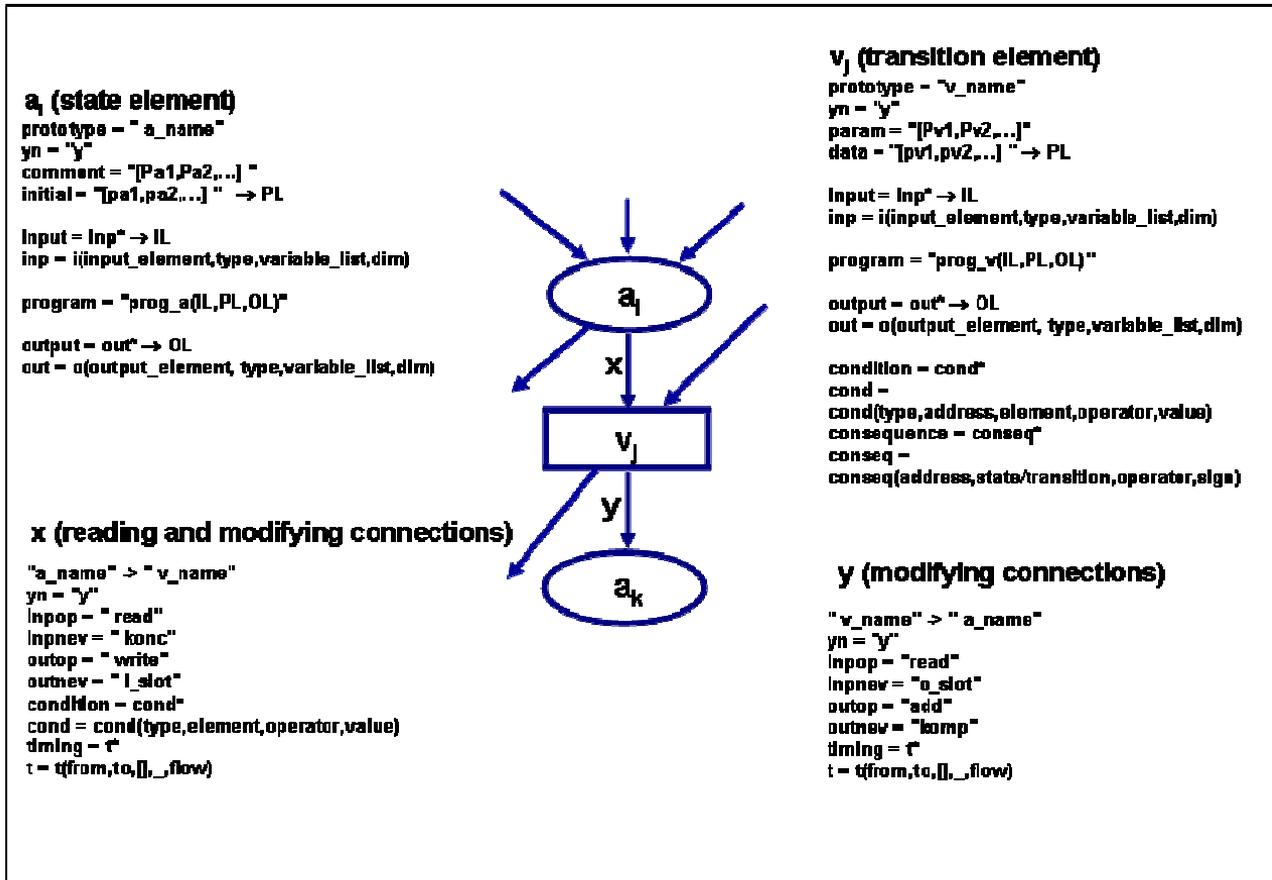
a_l (state element)

prototype = " a_name"
yn = "y"
comment = "[Pa1,Pa2,...] "
initial = "[pa1,pa2...] " → PL

input = inp" → IL
inp = i(input_element,type,variable_list,dim)

program = "prog_a(IL,PL,OL)"

output = out" → OL
out = o(output_element, type,variable_list,dim)

v_j (transition element)

prototype = "v_name"
yn = "y"
param = "[Pv1,Pv2,...]"
data = "[pv1,pv2,...] " → PL

input = inp" → IL
inp = i(input_element,type,variable_list,dim)

program = "prog_v(IL,PL,OL)"

output = out" → OL
out = o(output_element, type,variable_list,dim)

condition = cond"
cond =
cond(type,address,element,operator,value)
consequence = conseq"
conseq =
conseq(address,state/transition,operator,sign)

x (reading and modifying connections)

"a_name" -> " v_name"
yn = "y"
inpop = " read"
inpnev = " konc"
outop = " write"
outnev = " l_slot"
condition = cond"
cond = cond(type,element,operator,value)
timing = t"
t = t(from,to,[],_,flow)

y (modifying connections)

"v_name" -> " a_name"
yn = "y"
inpop = "read"
inpnev = "o_slot"
outop = "add"
outnev = "komp"
timing = t"
t = t(from,to,[],_,flow)

Figure 3.2.2-1: Illustration of the model generating declarations

The *comment* helps to identify the content of the state element for the expert and user, while the *initial* row contains the initial values of the data, enumerated in the *comment*.

The *input* and *output* declarations determine the patterns of the input and output variables. In the present work, the state elements are characterized by conservational measures and, accordingly, the input and output lists contain i(komp,l,IL,t) and o(konc,l,OL,kg_t) functors, respectively. These lists of functors unify in the Prolog kernel of the simulator. For the input, the i() functor describes the name, type, quantity list of components and dimension of the individual variables. Similarly, for the output, the o() functor describes the name, type, the list of calculated concentrations and dimension of the individual variables.

The *program* declaration describes, how the output variables are calculated from the input variables with the knowledge of the parameters. In present work the program of the state variables is determined by the "measure" prototype, that calculates intensive characteristics from the extensive properties by means of the respective general clause, embedded in the kernel.

In the declaration of the **transition elements** (e.g. v_j in Fig. 3.2.2-1) the role of *prototype* and *yn* is the same, as for the state elements. In the rows *param* and *data* the names of parameters, as well as their actual values are described in a prescribed order. The parameters refer to the economic (prices, costs) and technological (stoichiometries) data, determined by the expert and user.

In the *input* declaration standardized, comma separated i() functors are described between list parentheses. The i() functors determine the input variables, necessary to the calculation of the given transition.

In the *output* declaration standardized, comma separated o() functors are described between list parentheses. The o() functors determine the increase and decrease of the various quantities, calculated by the program.

Accordingly, the *program* defines, how can we calculate the local output variables from the local input values, with the knowledge of the local and global parameters. In addition, the program contains also the data and functionalities for the calculation of economic evaluation. Here are described the ranges of the uncertain economic parameters, too. The declarations *condition* and *consequence* are Prolog-like lists, containing the conditions and consequences to be taken into consideration during the execution.

The state → transition connections refer to the edges, determining the reading/modifying channels. The *yn* row characterizes the existence of the connection. The *inpop* declaration determines the type of the operator, used in the reading of the content from the state element, declared in the row *inpnev*. This reference activates the link between the local models, according to the structure of the process graph.

Similarly, the rows of *output* and *outnev* describe the execution of the output links.

Declaration of *condition* makes possible to define the actual conditions, determining the existence of the given connection.

Timing describes the constrains, determining the temporal execution of the given connection. The first and second data refer to the starting and ending time. For the conservational processes the remaining data help to describe the qualitative and quantitative properties of the transportations (e.g. flows).

The transition → state connections refer to the edges, determining the modifying channels. The respective model generating information

correspond to the formerly described state → transition connection, *mutatis mutandis*.

3.2.3 Method for the consideration of conditions and consequences, controlling the simulation of the time-varied structures

The additional complexity of the time-varied processes needs control actions to determine the sequence of the various transitions. The dead-lock and hazard situations can be avoided by the built-in conditions and consequences. The dead-lock situation means, that two partprocesses mutually detain starting of each other. In the hazard situation unexpected transitions can start or stop. Both case can be controlled by the conscious prescription of the sequential execution. The declaration of the conditions and consequences contains

$$condition = \text{cond*}$$
$$\text{cond} = \text{cond(type,addresse,state/transition,operator,value)},$$

and

$$consequence = \text{conseq*}$$
$$\text{consequence} = \text{conseq(address,state/transition,operator,sign)}$$

list of functors, respectively.

The listed elementary conditions are interpreted according to the AND logics. The **type** of the conditions may refer to a given quantity, to the existence of the given element, or to the consideration of any of the consequences. The **address** means the identifying name of the state or transition elements, waiting for the message. The global messages are addressed by the "any" value. The addressed messages are deleted at the execution of the given element, while the global messages are deleted by the

kernel, having accomplished the time step. The **state/transition** corresponds to the elementary state or transition, being modified according to the given condition. The **operator** depends on the type, for example in the case of quantity type we can use operators "more" or "less", while the operators of the existence or works type are the "yes" or "no" logical values. The **value** refers to the limitation of the amount for the quantity types.

In the listed elementary consequences the **address** determines the identifying name of the state or transition elements, that the message is sent for (the general messages are addressed by "any"). The **state/transition** identifies the name of the element, that will consider the message. The **operator** is used to sign whether the given element, sending the message does or doesn't work (type = works) or need (type = needs). Accordingly, the corresponding signs are logical values "yes" or "no".

3.2.4 Derivation of the upper level stoichiometries from the lower level models

The stoichiometric matrixes, describing the transformation of the components in the upper level model, can be determined in two ways:

a) The stoichiometric data can be calculated from the results of the detailed lower level models. Fig. 3.2.4-1 shows an example, how from the input and output data of the lower level model one can calculate the upper level stoichiometric ratios. In accordance with the example, in present work the detailed model based estimation of the stoichiometries has been studied for the anaerobic digestion.

b) In the lack of the detailed models, the stoichiometric matrixes can be estimated by the experts or from the literature.
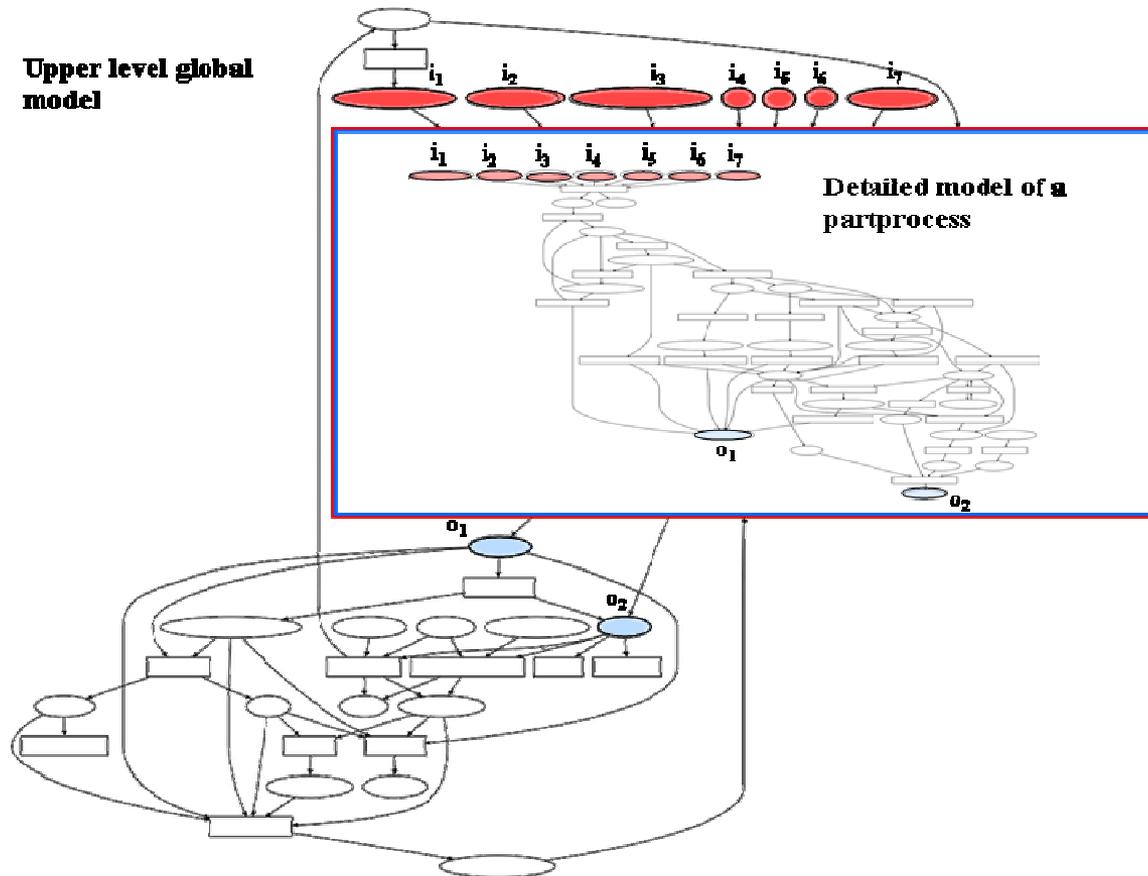
Upper level global model

Detailed model of a partprocess

Figure 3.2.4-1: Relation between the upper level global model and the detailed mode

4. CONCLUSIONS

The major conclusions of present work can be summarized as follows:

The optimalization methodology, based on the feedback system of the generic simulator and genetic algorithm, can effectively be extended by the new solution for the consideration of the uncertain economic parameters. The essence of the suggested method is, that all of the candidate alternatives generated by the optimization tool, are tested with another algorithm, which tries the given variant by a set of simulations, with randomly selected economic parameters. Having finished this set of simulation, the average evaluation, as well as the standard deviation of the evaluations are fed back to the genetic algorithm. The advantage of the method is, that it can be parameterized freely, and executed automatically. The disadvantage is the increase of the computational demand, that can be managed by macrogranularly parallel execution of the variants (e.g. in a PC cluster).

For the investigated class of optimal process design, characterized by changing structures and parameters, the evaluation of the alternatives can effectively be solved by the contribution margin. In the calculation of the contribution margin the objective function contains those, and only those costs, that may change for the various structures and parameters, suggested by the optimization algorithm.

According to the study, the process optimization under uncertainty can advantageously be carried out by a two-criteria evaluation, considering the average value and the standard deviation of the objective values, obtained from the simulations with randomly changed uncertain cost parameters. These two objectives can conflict with each other, however, having defined the uncertain ranges appropriately, the evolutionary method tends to develop feasible compromise solutions. Of course, it is presumed, that the effect of

the uncertain economic parameters must not dominate over the effect, caused by the possible alternatives. This fact has to be considered in the determination of the possible discrete and continuous properties of the variants. On the other hand, the unnecessarily wide ranges of the uncertain economic parameters have to be avoided, too.

The extension of the GraphViz description language, introduced in the present work, proved to support the flexible declaration of the complex process models for the generic simulator, based on Direct Computer Mapping. In the automatic generation of the expert and user modules, the enhanced GraphViz interface can effectively collaborate with the kernel of simulator (written in declarative logical language). The developed interfacing helps the expert to describe the brief local program codes, determining the relationships between the input and output variables of the state and transition elements of complex processes, in an easily editable text form. At the same time, the user can modify the structure and the parameters of the process models, combining graphical representation with the textual data. The experiences, obtained in the first case study, can be utilized in the further development of the more sophisticated future expert and user interfaces.

In the preparation of hybrid dynamic process models with time-varied structure (as in the studied example), the sequential execution has to be controlled by the effective and appropriate set of conditions and consequences, associated with the elementary transitions. According to my experiences, the possible dead-lock and hazard situations, coming from the sequential execution of model elements, can be avoided by these controlling measures.

The fast enough simulation of the large space- and time-scale problems needs relatively coarse time steps. The disadvantageous side effect

of the advantageously faster calculation is, that the calculated results fluctuate unfeasibly.

5. NEW SCIENTIFIC RESULTS

The new scientific results of my work are the followings:

1. I have elaborated a new methodology for the consideration of the uncertain economic parameters in the detailed model based optimalization (suboptimalization) of the complex technological processes. The essence of the methodology is, that the contribution margin based evaluation of the alternative solutions, synthesized from the possible discrete or continuous properties, is replaced for a two-objective evaluation, where the second objective tends to evolve those alternatives, that are insensitive for the specified uncertainty of the economic parameters (supply, demand, prices, costs, etc.).

2. I have extended the description language of the GraphViz package, to support the experts and users in describing the complex process models, studied by the generic simulator, based on Direct Computer Mapping. The extended model generating capabilities collaborate with the declarative logical language of the simulating kernel. The elaborated solution makes possible the expert the easily editable textual declaration of data and parameters, as well as of the brief program codes describing the relationship between the input and output variables, associated with the state and transition elements of the complex processes. On the other hand, the user can combine the graphical representation of the structure with the textual description of the elementary data, in the actual configuration of the model.

3. I have extended the definition of the transitions with a generally usable declaration of conditions and consequences, that can avoid the dead-lock and hazard situations, coming from the sequential execution of the elements for the hybrid dynamic processes of time-varied structures.

4. I have prepared and studied an example model for the macro-level simulation based optimal design of a complex, recycling technological system, involving food industrial and agricultural processes with utilization of byproducts and wastes. Based on the systematic analysis of the technological and uncertain economic parameters, I have determined the possibility space and the evaluating objectives for the investigated case study.

5. I have embedded the elaborated new methods in the general purpose optimizing package, applying the evaluation feedback between the generic simulator and genetic algorithm, developed by our research group. Having analyzed the case study for the suboptimalization of a macro level simulation model, I have concluded, that the extended methodology can effectively be applied for the process optimization under economic uncertainties. Based on the analysis of the model, I have showed, that the precise technical and economic evaluation needs the more detailed modeling of the keynote anaerobic digestion.

6. I have built the detailed dynamic simulation model of an experimental anaerobic fermentor of concentrated parameters. Based on the literature and on the available pilot-scale measurements, I have identified the *'a priori'* parameters for the first draft version of the model.

6. RECOMMENDATIONS

Based on discussion of the results my recommendations are the followings.

Further detailed investigations ought to be carried out, to determine the relation between the measure of the domain, determining the possible technological changes, and the uncertainty ranges of the economic parameters, for more precise outlining the feasibility limits of the elaborated methodology.

Further studies ought to be planned to analyze, whether the effectivity of the optimalization can be improved by other evaluations, applied besides or instead of average value and standard deviation of the values (e.g. the minimal and/or maximal value obtained for the set of variants prepared by the uncertainty generator).

The extended GraphViz interface ought to be enhanced by user friendly options, for type checking and for controlling of the variable bindings.

In case of the practical application of the process development for the investigated complex technological system, the data set ought to be refined and additional measurements ought to be carried out to increase the reliability of the simulations.

Considering the more precise analysis of the case study, the detailed dynamic model of anaerobic fermentor ought to be refined. This would need additional knowledge about the anaerobic digestion of the alternative raw materials. It would be important also to model the co-digestion of sugar beet slice with various alternative raw materials of different quantity. The co-digestion of the additional resources with the stored sugar beet slice seems

to be more likely, than the full change of the raw materials in the subsequent periods.

7. PUBLICATIONS IN THE FIELD OF THE DISSERTATION

Scientific publications in foreign language

Varga M.: Economic Optimization of Sustainable Complex Processes under Uncertain Cost Parameters. Regional and Business Studies, 2009 (in press).

Varga M.: Natural Computational Methodology for Optimization under Uncertain Economic Parameters. Submitted to Journal of Information Technology in Agriculture.

Scientific publications in Hungarian language

Varga M.: Egy nagyobb méretű metabolikus modell kifejlesztése. Modell leírás és előzetes eredmények. *Acta Agraria Kaposváriensis*, 10 (3), 2006, pp. 153-161.

Varga M., Balogh S., Boldizsárné Sinkó I., Csukás B., Hantos G., Katonáné Tóbiás E.: Az enzimkoncentráció meghatározása az aktivitásmérés identifikált modellje alapján. Műszaki Kémiai Napok'07, Veszprém, 2007, ISBN 978-963-9696-15-0. pp. 293-296.

Domonkos D., Bélteky L., Barthó I., **Varga M.**, Balogh S., Csukás B.: Rekombináns fehérje szintézis számítógépi modelljének identifikálása genetikus/generikus szimulátorral. Műszaki Kémiai Napok'07, Veszprém, 2007, ISBN 978-963-9696-15-0. pp. 175-179.

Varga M.: A mitokondrium és a kloroplaszt modellezésének összehasonlító elemzése és tanulságai. *Acta Agraria Kaposváriensis*, 11 (2), 2007, pp. 249-259.

Varga M.: Modell bázisú optimálás a kapcsolódó folyamatok bizonytalan, kooperatív értékelésével. *Acta Oeconomica Kaposváriensis*, 7 (1-2), 2007, pp. 121-134.

Presentations

Varga M.: A mitokondrium és a kloroplaszt modellezésének összehasonlító elemzése és tanulságai. *VI. Alkalmazott Informatika Konferencia*, Kaposvár, 2007. május 25.

Varga M.: Részletesen modellezett termelési folyamatok optimálása a kapcsolódó folyamatok bizonytalan, kooperatív értékelésével. *I. Kaposvári Gazdaságtudományi Konferencia, Kaposvár, 2007. november 9.*

Varga M.: Bioreaktorok keveredésének szimulációval segített műszaki-gazdasági elemzése. *Alkalmazott Informatika Konferencia, Kaposvár, 2008. május 23.*

Varga M.: Economic optimization of sustainable complex processes under uncertain cost parameters. *2nd International Economic Conference, Kaposvár, 2009 április 2-3.*