SOFTWARE ARCHITECTURE FOR THE INVESTIGATION OF CONTROLLABLE MODELS WITH COMPLEX DATA SETS*

Dmitry Belyshev and Vladimir Gurman
Program Systems Institute RAS
Pereslavl, Russia

Abstract Investigation of a large control system needs rather sophisticated applications, which would satisfy such users' requirements as necessity of unprescribed modifying all elements of model and data set under consideration depending on the current results of calculations; opportunities for use of different methods to solve the same problem or the same part of problem most effectively; typically experimental character of computations defined by contentual task (comparisons, generating of versions to analyze these ore those dependencies, etc); adaptation to the user as an object expert, not as a programmer.

There are proposed basic principles of regional modeling and software architecture for creation and proceeding of such models. Socio-ecologo-economical modeling of region with optimization of its development strategy is considered. The main topics are related to software supporting of modeling, model proceeding and representation, usage of optimizing algorithms and creation of optimization procedures for regional development.

A sample software application and results of numerical experiments, based on the model of Pereslavl region are also described.

Keywords: socio-ecologo-economical modeling, optimal control, software architecture, multimethod procedures, regional development

1. INTRODUCTION

Investigation of a large control system needs rather sophisticated applications, which would satisfy such users' requirements as:

*Supported financially by RFBR, Project No 00-01-00731

The original version of this chapter was revised: The copyright line was incorrect. This has been corrected. The Erratum to this chapter is available at DOI: 10.1007/978-0-387-35407-1_22

R. F. Boisvert et al. (eds.), *The Architecture of Scientific Software*

- necessity of unprescribed modifying all elements of model and data set under consideration depending on the current results of calculations;
- opportunities for use of different methods to solve the same problem or the same part of problem most effectively;
- typically experimental character of computations defined by contentual task (comparisons, generating of versions to analyze these ore those dependencies, etc);
- adaptation to the user as an object expert, not as a programmer.

The socio-ecologo-economical model of a region with innovative submodel and algorithms of optimization of development strategy (Carraro, Deissenberg, Gurman et al., 1999) serves as an expressive example of such a control system.

The model with innovations for a region is constructed as a modification of the general model described in (Gurman, 1981)

$$c = (E - A)\,y - Bu - A^z z - B^z u^z - A^d d - B^d u^d \tag{1}$$

$$0 \le y \le \Gamma(k), \quad 0 \le z \le \Gamma^z(k^z) \tag{2}$$

$$\dot{r} = \frac{dr^*}{dt} + N(r - r^*) - Cy + C^z z + im^r - ex^r \tag{3}$$

$$\dot{k} = u - \delta k, \quad \dot{k}^z = w - \delta^z k^z. \tag{4}$$

Here y is current traditional output; z is output vector of the environmental sector; u and u^z are investments in traditional capital k and in capital k^z of the environmental sector; c the real consumption; r is a vector describing the state of the environmental sphere. E is the identity matrix and $A, B, C, D, A^z, B^z, C^z$, and D^z are coefficient matrices of appropriate dimensions; δ and δ^z are diagonal matrices of depreciation rates

The innovative part is described by

$$\dot{\theta} = -([d] + H_{\text{inv}} + H_{\text{diff}})(\theta - \bar{\theta}), \quad \theta(0) = 0,$$
$$\dot{k}^d = u^d - \delta^d \times k^d, \quad 0 \le d \le \Gamma^d(k^d),$$

where d is a control vector representing the activity of innovative sector which capital, capacity and depreciation rate are correspondingly

$k^d, \Gamma^d k^d$, and δ^d, $[d]$ is a diagonal matrix as "diagonalized" vector d, matrices H_{inv}, $[d]$, and H_{diff} correspond to three main sources of innovations:

1) the investment which typically imply replacing older vintage capital with newer one, with better characteristics;

2) specific expenditures for technology and management improvement that have no direct capacity effect;

3) innovative diffusion processes and similar endogenous innovative changes.

Typically, $\bar{\theta}$ may represent the best available innovative level in similar regions.

The positive impact of the basic capital, socium and environment quality can be accounted for via multiplying the right-hand side of (1) by some matrix K depending on related state variables.

The innovative costs are presented in the main balance equation (1) by $A^d d$ and $B^d u^d$.

The parameter vector a is defined as the vector of all stacked elements of the model's coefficient matrices, i.e. of the coefficient matrices and other parameters or their inverses so that "innovative improvement" means reducing the components of a.

The number of parameters can be very large: roughly speaking, it is proportional to the squared number of state variables in the model. We shall therefore in practice not work with the vector of parameters $a = (a_i)$, but with a lower-dimensional, aggregated vector θ. This vector is obtained with the help of the following procedure. We first partition the elements of the vector a in m subsets I_j, $j = 1, 2, \ldots, m$. To each subset I_j corresponds one element θ_j of θ. The current value of θ_j is constructed according to:

$$\theta_j = \left(\sum_{i \in I_j} \frac{|\Delta a_i|}{|a_i|} \right) \frac{1}{n_i} \tag{5}$$

where a_i is the current value of the i-th parameter in I_j; n_j is the number of parameters in I_j; $\Delta a_i = a_i - a_i(0)$; and $a_i(0)$ is the initial value of a_i. Thus, θ_j describes the mean percentage change for the subset of parameters I_j.

The values of θ_j are disaggregated when required by "redistributing" them for each t among the parameters belonging to the subset $\mathbf{I_j}$. Several rules can be used to perform this disaggregation, in particular:

— by changing all parameters in the subset by the same percentage;

— by weighting the changes according to the level of saturation of each given parameter, that is, its distance from its possible value boundary;

— according to empirical statistical distributions for the parameters;
— using an additional conditional optimization with respect to the parameters subject to θ_j.

For this purpose we use the following formula

$$a_{ij} = (1 + \theta_j \alpha_{ij}), \quad i \in \mathbf{I_j},$$

where α_{ij} are are distributive weight parameters which can be chosen differently according above different distributing rules.

As a result a complex nonlinear model is obtained even if the original model is linear.

An optimization problem for this model is stated which is to maximize the following intertemporal welfare function

$$J_T = \int_0^T [(1-\mu)V - \mu W] e^{-\rho t} dt \tag{6}$$

where $\rho \geq 0$ is the discount factor, $T < \infty$, and $\mu \in [0,1]$, $V = pc$, u^d is the vector of investments in k_d, $Q_d d$ is the current innovative cost, and W is a penalty function for the violation of "soft" sustainability constraints,

$$r \in \mathbf{\Psi^s(t)}.$$

Formally (6) is represented as

$$\dot{J} = \left[(1-\mu)V - \mu W(t,k,k^z,r) - u^d - Q^d\right] e^{-\rho t}, J(a) = 0, J(T) = J_T()$$

The problem of interest is to find a time profile for the control vectors y, u, z, w, d and u^d, that maximizes (6) subject to the model relations, the additional constraints on controls

$$\left(y,u,z,u^z,d,u^d\right) \in \Omega(t,k,k^z,k^d,r,\theta), \tag{7}$$

"hard" sustainability constraints

$$(k,r,k^z) \in \Psi^h(t),$$

and given initial and terminal boundary conditions for k, r, k^z, θ, and k^d.

The complexity and nonlinearity of the resulting integrated model does not allow one to use for it directly any special optimization method. However, if the original model does allow to do it in analytical way, may be approximately, then this opportunity can be used via the following multistep procedure.

Step 1. Disregarding the equation $\dot{\theta}$, the different model's coefficients are considered as arbitrary given functions of time.

Step 2. The thus defined optimal control problem is solved by the special method as mentioned above under a series of idealized assumptions.

Step 3. The control trajectory obtained under Step 2 is locally improved on account of control program $\{d(t)\}$.

Step 4. Steps 1–3 are repeated until no improvement is possible.

Step 5. The idealized assumptions are removed, and the solution is modified correspondingly by an iterative improvement method (see Belyshev and Shevchuk, 1999).

In the steps 1–2 we assume the following:

1) the controls u and z are unbounded; the equation for $\dot{k}^z$ in(4), is dropped, and the term $B_z w$ in (1) is taken zero;

2) the boundary conditions on k, k^z, and r are fix point ones.

To solve these steps, we base on the extension principle (Gurman 1985, 1997) which idea is to replace the original problem by an analogous one but free of some "difficult" constraints so that its solution either exactly coincides with or lies close to the solution of the original problem. Specifically we use "singular relaxation" method where the extensions considered are so called singular relaxation, i.e. extensions that, applied to a class of unbounded control systems, preserve their all integral characteristics. They allow to represent generalized optimal solutions as impulse modes in terms of regular ordinary differential equations.

2. DESCRIPTION OF THE SOFTWARE PACKAGE

The following approaches to the corresponding software architecture were proposed and implemented in part:

1 The level of elementary operations is increased in comparison with standard packages;

2 The object-oriented design of model representation is used. It is important to give a researcher chance to effectively work with such huge sets of data and algorithms. This approach creates an internal hierarchy in a model;

3 To store such complicated structures a special object database is needed, which is capable to keep different types of data;

4 The model dynamics (such as differential and difference equations and various restrictions) are represented as a source code of some

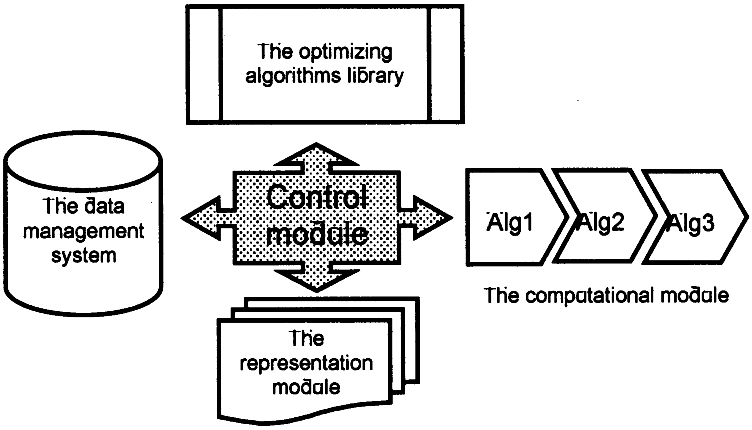

Figure 1 The scheme of software architecture

interpreted language. This allows the user to change all the model equations parameters in runtime. It is also useful as a method to keep this data in common database;

5 The model dynamics source code via is improved pre-compilation (supercompilation);

6 The elements of artificial intelligent (expert systems) are used in multimethod procedures and when assembling the main program out of modules;

7 Distributed and parallel computation are provided;

8 Common interfaces to work with different optimizing algorithms designed by other developers, and possibility to create an optimization sequences are developed.

One of the main our objectives is to create flexible software architecture to support different changes of models and algorithms and to give the user tools to verify each step of a computational experiment.

A sample software package, based on above principles, has been created and used to investigate above regional model.

Figure 1 represents the scheme of this software application. It consists of five main parts: (1) data management system, (2) optimizing algorithms library, (3) control module, (4) computation module, (5) representation module.

1. The data management system contains full description of active (procedures, realizing model dependencies and differential equations) and passive (static and dynamic scalars, vectors and matrices)

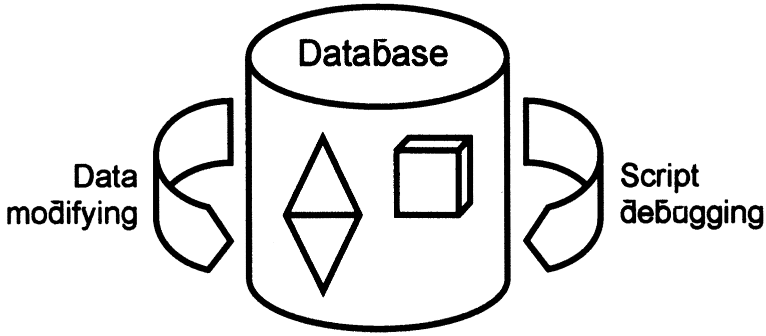

Figure 2 Data management system

parts of model. The model is considered as a whole object with internal hierarchy and specific methods, or "elemental operations", such as "calculate the state in time t, starting from initial state and control program", "calculate the value of functional at the time T_F", "determine if the current path is locally optimal or not" and "proceed to the optimal path in some neighborhood of the current state". Model procedures as it has been written before, are stored as source codes in interpreted language. Interpreter of Java Script is used in current program version, therefore the data management system has mechanisms for script writing and debugging.

2. The optimizing algorithms library contains collection of subprograms, realizing algorithms for model optimization. Each subprogram is a dynamic link library (DLL). In view of the fact that the model is rather intricate, it is useful to have a multi-step optimizing procedures and a set of optimizing algorithms. In this case it is necessary to have a common interface for algorithms integration with the system and with each other. Such interface was described, so that it lets one to construct sequences of optimization, connecting procedures with each other. Such approach also lets one to insert new algorithms into the library without any changes in the main program. Thus the algorithms library is open for insertion of new methods and algorithms.

3. The control module is made for the user-system cooperation between. It should give the user a friendly and intuitively understandable interface, which helps partly to remove the problem of misunderstanding (for the end user) the model description and the interaction formalism. Because of object-oriented design of the model representation, control tools are divided on groups, connected with special objects.

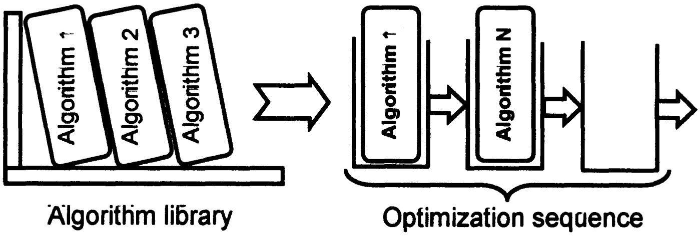

Figure 3 The optimizing algorithms library

It is very important that the control module have tools to backtrack all the dependencies and overpatching in the model under investigation. User can interrupt computational process and trace necessary parameters or he can print intermediate data and do any corrective actions to the model during the computation. These system properties give wide opportunities to the researcher in his work.

4. The computational module is the most important for final procedure. The argument for this module is prepared by the model which, as above example shows, may have very complicated data structure, a lot of variables of different nature divided on different groups.

The calculations work under the following scheme:

$$x(t+1) = f(x(t), u(t), t),$$

where x(t) is the state vector, u(t) is the control vector at the time t, and f is the model operator.

The user chooses an optimization algorithm and starts to set algorithm parameters for the current situation. As parameters of algorithm we consider, for example, settings for exponential penalty functions such as $F = \frac{1}{\beta}e^{\beta(x-x_{max})}$. It is important to set correct and effective coefficients for penalty expressions, because they are responsible for observing the model restrictions. To solve this problem effectively it is planned to develop procedures supported by artificial intelligence tools, such as expert systems. They will help the user to setup necessary parameters on the base of previous experience.

In perspective it will be added a precompilation subprogram, which improves the source code of the model dynamics before usage. The means to clear program code and to decrease the computation times are described in (Turchin, 1985).

5. The representation module is very important for research work. It helps to trace all changes and dependencies in the model and

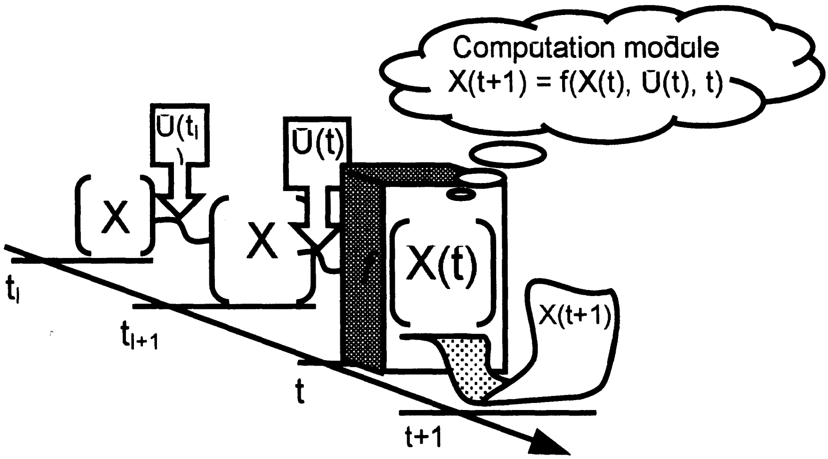

Figure 4 The computational module

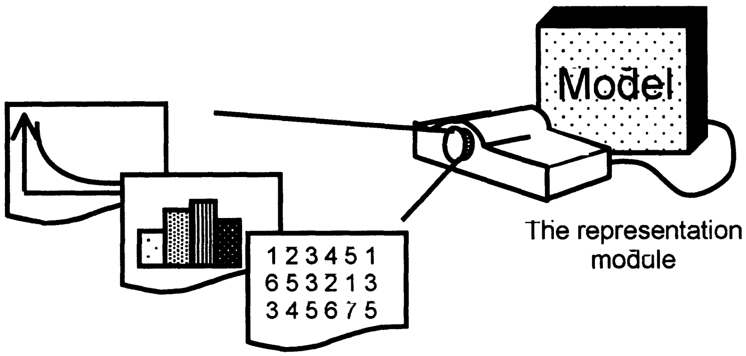

Figure 5 The representation module

to get the results. The software has to give the user a mechanism to get any information about model at any desired time, and to represent it in a convenient view. For more comfortable work and more qualitative representation it uses standard Windows libraries and prevalent software such as Microsoft Office. Optionally it can print data and plot graphs straight on Microsoft Excel sheets. Some results of this module are used in this paper as graphs and tables.

3. EXAMPLE: APPLICATION TO PERESLAVL REGION

Pereslavl region is a part of Yaroslavl region of Russia. Its center, Pereslavl-Zalessky town, is situated on the famous Golden Ring of Russia, halfway from Moscow to Yaroslavl city, on the lakeside Plestcheevo (aquatory of 51 sq. kilometers), which has the status of a natural and historical memorial. Its territory is of 3300 sq. kilometers, and present population is about 71 thousands.

The leading branches of economy are photochemical, textile, food industries, and agriculture. There is a great potential for tourism and recreation taking into account comparatively good ecological conditions and numerous beautiful landscapes, historical relics and monuments. The model described in the Introduction was specified for this region according to the following formulas:

$$\Gamma(k) = \gamma k,$$
$$\Gamma^z = \gamma^z k^z$$
$$\Gamma^d = \gamma^d k^d,$$
$$\Psi^s : r^s_{\min} \le r \le r^s_{\max}, \quad \Psi^h : r^h_{\min} \le r \le r^h_{\max},$$
$$W = \textstyle\sum_i w_i |r_i|, \text{ if } r \notin \Psi^s, \quad W = 0, \text{ if } r \in \Psi^s,$$
$$\boldsymbol{\Omega} : \mathbf{L}^{\text{dem}}_{\min} \le \mathbf{L}^{\text{dem}} \le \mathbf{L}^{\text{dem}}_{\max},$$
$$L^{\text{dem}} = l^y y + l^z z + l^d d,$$
$$q u_{\max} + q^z u^z_{\max} + q^d u^d_{\max} = u^{\text{sum}}_{\max}.$$

Here L^{dem} and l^y, l^z, l^d are the total labor demand, and specific demand row-vectors, $L^{\text{dem}}_{\min}$ corresponds to the admissible unemployment, and $L^{\text{dem}}_{\max}$ is the total labor resource available. The last relation means distribution of the total investment available in the region $i^{\text{sum}}_{\max}$ between different types of activity according the weight coefficient vectors q, q^z, and q^d. It is specified quantitatively in scenarios.

The corresponding date are listed in the Table 1.

The following basic scenarios of the region development were considered.

First scenario (I) is continuation of the trends of past basic period 1995-1998 up to 2014 without optimization, and without active control for the ecological and social and innovative components ($z = 0$ and $d = 0$).

Other scenarios were considered on the period 1999-2014 (15 years). They include approximately global optimization accompanied by iterative improvement.

Second scenario (II) is the same as (I) but with approximate optimization without active innovative control, to reveal the level of structural efficiency.

Third scenario (III) presumes that the innovative control is activated under different restrictions on investments (several versions). The fastest gaining of structural efficiency is taken for the innovative control program as initial approximation.

Table 1 Pereslavl region: basic data (the state variables' values refer to the end of 1995).

VAR	*1*	*2*	*3*	*4*	*VAR*	*1*	*2*	*3*
k	211	251	37.3		k^d	10	10	8
k^z	4	12	0.15	8.5	θ	0	0	0
r	5800	8.2	69.5	0.44	$\bar{\theta}$	-0.5	-0.8	-0.7
$r^h_{\max}$	-	10	200	1	p_{row}	1	1	1
$r^s_{\max}$	-	8.2	120	-	γ	0.4	0.35	0.5
$r^s_{\min}$	5000	-	60	0.4	γ^d	0.002	0.003	0.003
$r^h_{\min}$	4000	0	50	0	l^y	0.035	0.09	0.13
r^*	6000	1.2	100	0.5	l^d	10	10	10
γ^z	3.7	0.019	0.03	0.0026	δ_{diag}	0.06	0.06	0.07
l^z	0.03	0.3	2.5	40	δ^d_{diag}	0.06	0.06	0.06
im^r	0	0	0.6	0.08	H_{diag}	0.01	0.01	0.01
ex^r	0	0	0.1	0 7	A	0.08	0.001	10^{-5}
w	0.6	60	60	4200		0.5	0.4	0.35
δ^z_{diag}	0.07	0.09	0.06	0.11		0.001	0.006	0.06
C^z_{diag}	1	-1	1	1	B	0	0	0
A^z	0.2	27	3	200		0.45	0.15	0.4
	0.4	33	40	1000		0	0	0
	0	0.2	20	3000	B^d	0	0	0
B^z	0	0	0	0		0.3	0.3	0.4
	0.3	0.35	0.2	0.15		0	0	0
	0	0	0	0				
N	-0.003	-4	-0.05	0.01	Q^d	50	50	50
	-10^{-4}	-0.9	0.01	-0.005				
	0	-0.0015	-0.005	-0.2	$L^{dem}_{\max}$	43		
	10^{-6}	-0.005	-10^{-4}	-0.002				
C	0	0.011	0		ρ	0.05		
	-0.0053	-0.0059	-0.0034					
	0.0001	0.0005	0.0003		$L^{dem}_{\min}$	27		
	0.0002	0.0001	0.0003					

Fourth scenario (IY) accounts for the dependence of model parameters on r including positive dependence of innovative matrices on culture and education. Optimization is performed by the universal improvement algorithm.

Fifth scenario (Y) is to estimate the effect of price change due to changing the regional compatibility.

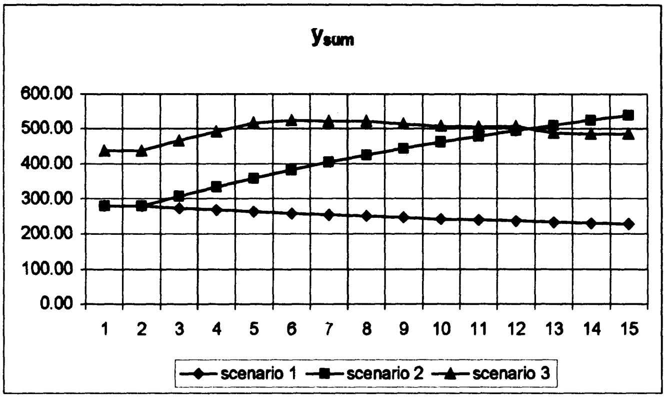

Figure 6 Computational results for first scenario

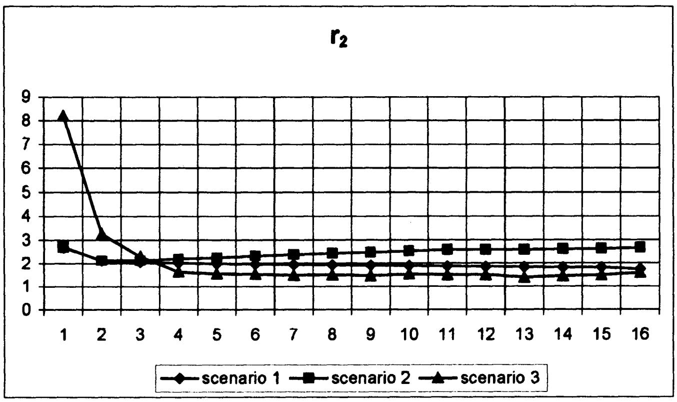

Figure 7 Computational result for the second scenario

Some of the results of model calculations for the first three scenarios are shown on Figures 6 – 8. The initial conditions for them were obtained via calculations according scenario (I) for 1995– 1998. In the whole they are confirmed by preliminary statistical data for 1998, which is insufficient to recalculate strictly into the required set of data.

As is seen from results for scenario 2, only second aggregated branch of economy is productive and works at the full capacity and with maximal investment in the optimal mode; two others branches are nonproductive; they should work at minimal admissible (from employment considerations) level and without investment. This means that at present time the regional system is structurally ineffective.

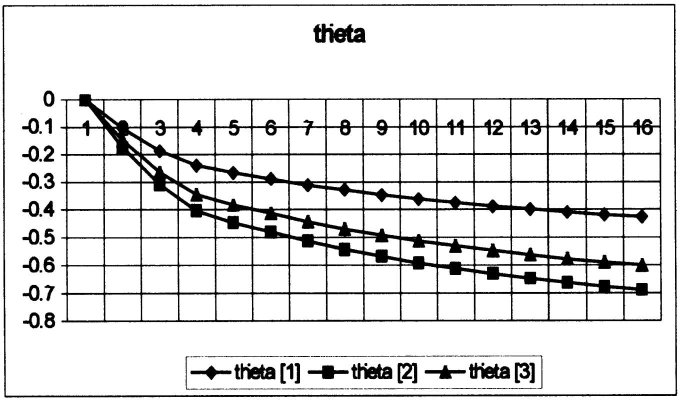

Figure 8 Computational result for the third scenario

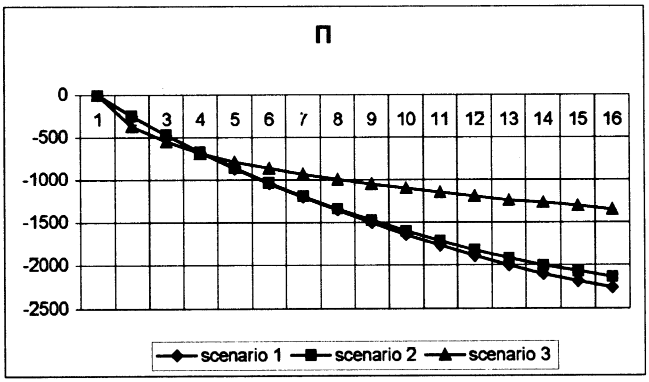

Figure 9 Comparison of the three scenarios

To overcome this drawback in acceptable time horizon active innovative policy is required. As is represented by scenario (III) it consists of radical redistribution the investment to come in favor of the innovative sector which functioning is aimed first of all to improve the structural efficiency parameters, that is elements of matrices A, C, A^z. As a result after 5 years all the three aggregated economy branches become productive, that is structural efficiency is attained. After that the investments are redistributed gradually in favor of economic sector and capacities supporting the improvement of most negative ecological and social characteristics.

4. CONCLUSION

The model presented and numerical experiment show the peculiarities of the research work with big complicated objects such as a region. Thus the standard approaches and methods of investigation of such objects are not effective. The problem is not only the computational complexity but also the complicated object structure, large quantity of interacting components and nonlinear dependencies between them. These features highly complicate the work of a researcher and the software proposed has to give him such object representation that simplifies its understanding and helps to work with it.

Another feature is the experimental character of computations. Thus the software architecture should have mechanisms to trace and change almost all parameters of the object under investigation at any time, moreover it has to be open for entering of new processing units and with installing.

The fulfillment of these software requirements allow one to achieve considerable results in investigation. Thus the software architecture proposed can find many applications in different areas.

References

[1] Belyshev D.V and Shevchuk E.V.(1999). *The algorithm and software package for optimal control search, in "Intelligent control"*, Proceedings of the International Conference, Pereslavl-Zalessky, December 6–9, Nauka • Fizmatlit, Moscow, pp 146–150 (in Russian)

[2] C.Carraro, Ch.Deissenberg, V.Gurman and E. Ryumina. *A general framework for regional sustainable development for NIS Countries: Model*, TACIS ACE Project No P95-4097R

[3] Gurman, V.I. (1985, 1997). *The extension principle in control problems*, Nauka, Moscow (in Russian)

[4] Gurman V.I.,red.(2000). *Modeling of a region as a socio-ecology-economic system*,Nauka, Moscow.

[5] Turchin V.F. *A program transformation by supercompilation* // Ganziner H., Jones M.D. (ed.) Programs as Data Objects. (Copenhagen, Denmark). Lecture Notes in Computer Science, Vol. 217, p. 257-281, Springer-Verlag 1985.

DISCUSSION

Speaker: Dimitry Belyshev

Margaret Wright : Who are the users of this system? What is their level of expertise?
Dimitry Belyshev : The system can be used for more than just regional modeling. We can predict the development of business and take the time period not a year, but a month or a week. In this case the system can be used for estimation of management strategy by businessmen.
Ivor Philips : How do you verify the correctness of the mathematical model since it is predicting events that won't be verifiable until 2010?
Dimitry Belyshev : The correctness of the model is a very important part of the investigations. Obviously, we don't know what will happen in 2010, but we have full information about our history of development. So to verify the model it is enough to begin with historical data and check the results in comparison of history. We can also "turn the time back" and use the model to "predict the past" and observe the discrepancy between our artificial history and real facts.
Morven Gentleman : You suggest that this model can be validated by using the historical record. However, your forward study compared three scenarios: Current, with a new industry, and with optimization. Since historically what happened is fixed, how could you compare those three scenarios?
Dimitry Belyshev : Yes, this method can't be used if we made some changes in the model. However, we can validate the basic characteristics and mechanisms of the model. It could be enough, because most of our optimizations relate to assignment of money between different branches of the economy and do not change its parameters. If the economics part will work properly in the basic scenario, it will be the same in others.
Bo Einarsson : Have you looked at the outcome of the prognoses if you did not include the social and ecological aspects?
Dimitry Belyshev : Yes. The regional profits became close to the real situation. They are not very big, but not negative.
Bo Einarsson : When you made the backwards prognoses, did you then consider the discontinuity in your country?
Dimitry Belyshev : No. We only try to model the economical processes, but we can't predict our politics. The historical investigations can be made only in a stable period of economy.