Concordance of Private and Public Interests: Dynamic Graph Representation, Identification and Simulation Modeling

A.V. Antonenko, O.I. Gorbaneva, G.A. Ougolnitsky

Department of Applied Mathematics and Computer Science, Southern Federal University, Rostov-on-Don, Russia

Annotation

A universal approach to the description of dynamics of complex systems, identification of their models, and finding the solutions by means of computer simulation is proposed (on the example of CPPI-models). The dynamic graph balance models permit to reflect in a convenient visual form the main matter-energetic processes of real-world systems dynamics. Computer simulation is proposed to use both for solving of complex game theoretic problems and for their identification. An idea of representative scenarios is developed in this frame. A special computer software should be developed for the implementation of the method.

Key words: Computer simulation, Concordance of private and public interests, Dynamic graph balance models, Identification.

1 Introduction

In the seminal paper [7] a static model of joint consideration of private and public interests was proposed. They proved that if each agent's payoff function is a convolution by minimum of the private and public parts then a Pareto optimal Nash equilibrium exists in the agents' game in normal form. An investigation of the models of concordance of private and public interests (CPPI-models) was continued by the authors [8,9]. The conditions of system compatibility in CPPI-models based on the notion of price of anarchy [1] were studied, and different mechanisms of control providing the system compatibility were analyzed. Dynamic versions of CPPI-models were also built [3,4].

This paper makes a contribution to dynamic graph representation, identification and simulation of CPPI-models as instruments of the applied systems analysis. First, there is a number of mathematical models which permit to describe the state of complex systems including explicit or implicit consideration of their dynamics. Some examples are Markov chains, finite automates, Petri nets, queuing systems. In this paper we develop a technique of dynamic graph balanced CPPI-models [15]. Second, the standard methods of econometrics [6] and theory of identification [14] are based on long time series of reliable data which are often absent in real applications. We propose to solve the problems of structural and numerical identification by means of building a special computer software. Third, computer simulation [13] is an appropriate method of solving complex dynamic problems. We specify this method for CPPI-models with different information structure and introduce the idea of representative scenarios.

The rest of the paper is organized as follows. In Section 2 dynamic graph balanced CPPI-models are described. Section 3 gives an idea of computer simulation with CPPI-models with different information structure based on a small number of representative scenarios. Section 4 is concerned with computer simulation support of the identification of CPPI-models. Section 5 concludes.

2 Dynamic Graph Balanced CPPI-Models

The description of the state of a complex system in the moment of time t taking into consideration its structure by means of dynamic multidigraphs [15] includes the following elements.

1. A set of vertices $Y(t)=(y_1(t),...,y_n(t))$ where n(t) is a number of vertices in the moment t. Decompose the set Y(t) onto two non-intersecting subsets: $\forall t\ Y(t) = Y_1(t) \cup Y_2(t), Y_1(t) \cap Y_2(t) = \varnothing$ (it is possible that $Y_2(t) = \varnothing$). Let's name the vertices from the subset $Y_1(t)$ compartments and denote them by squares and the vertices from the subset $Y_2(t)$ transformers and denote by circles.

2. A set of arcs $Z(t)=\{z_{ij}^k(t)\}$, $1 \leq i, j \leq n(t)$, $1 \leq k \leq N$, where $z_{ij}^k(t)$ is the arc from the vertex y_i to the vertex y_j (in particular the loop if i=j) on which a resource k can move in the moment t; N is a total number of resources within the system.

3. A set of state variables of the compartments $X(t) = \{x_i^k(t)\}$, $1 \leq i \leq n(t)$, $1 \leq k \leq N$, where $x_i^k(t)$ is a value of the resource k in the compartment $y_i \in Y_1$ in the moment t. Then $x_i(t)$ is a state vector of the compartment y_i in the moment t (the collection of all its resources).

4. A set of state variables of the arcs $F(t) = \{f_{ij}^k(t)\}$, $1 \leq i, j \leq n(t)$, $1 \leq k \leq N$, where $f_{ij}^k(t)$ is a weight of the arc $z_{ij}^k(t)$, i.e. a number of the resource k moved during the time [t,t+1] from the vertex y_i to the vertex y_j, $i \neq j$, or a quantity of increase (decrease) of the resource k in the compartment y_i during the same time, i=j ($\Delta t=1$). In each considered situation (problem) the set F(t) can be split into two non-intersecting subsets: $\forall t\ F(t)=F_1(t) \cup F_2(t)$, $F_1(t) \cap F_2(t) = \varnothing$ (in particular it is possible that $F_2(t) = \varnothing$). Variables from the set $F_1(t)$ are called regulated (they change in the strength of given rules) and variables from the subset $F_2(t)$ are called regulators (they can change arbitrarily in the admissible set).

5. A set of limitations on the compartments capacity $\underline{X} = \{\underline{x}_i^k\}$, $1 \leq i \leq n(t)$, $1 \leq k \leq N$, where $\underline{x}_i^k$ is a maximal number of the resource k which can be stored in the compartment y_i.

6. A set of limitations on the carrying capacity of the arcs $\underline{F} = \{\underline{f}_{ij}^k\}$, $1 \leq i, j \leq n(t)$, $1 \leq k \leq N$, where $\underline{f}_{ij}^k$ is a maximal number of the resource k which can be moved from the vertex y_i to the vertex y_j, $i \neq j$, or produced (destructed) in the compartment y_i, i=j, during the time unit. Thus, the extended state of a complex system is a set $S(t) = < Y(t), Z(t), X(t), F(t), \underline{X}, \underline{F}>$.

To avoid a consideration of digraphs with multiple arcs let's map to each vertex $y_i \in Y_1$ the only value $x_i(t)$ and to each arc $z_{ij} \in Z$ the only weight $a_{ij}(t)$. Then a dynamic structure of the system consists of the separate "scalar" structures each of which represent a certain aspect of matter and energy interactions within the system. The partition of a set of vertices of the dynamic digraph onto two parts permits to describe the principal matter-energetic processes in the real-world systems such as 1) movement (transfer, exchange) of resource between the compartments; 2) production/destruction of the resource in the compartments; 3) transformation of the resource. The respective models can be called dynamic graph balanced ones. Describe the processes by such models.

1. A movement of the resource k between two compartments y_i and y_j in the moment t can be performed if the arc $z_{ij}^k(t)$ exists (Fig.1).

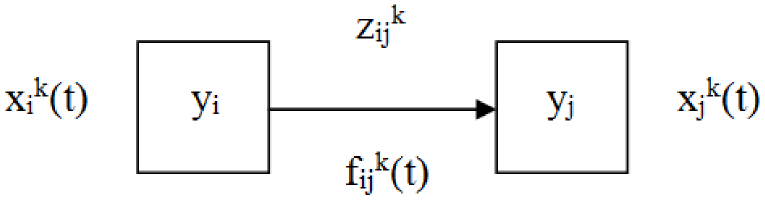

Fig.1 A movement of the resource k between compartments y_i and y_j

Assume that in the moment t the stocks of the resource k in the compartments y_i and y_j are equal to $x_i^k(t)$ and $x_j^k(t)$ respectively and the state variable of the arc $z_{ij}^k(t)$ is $f_{ij}^k(t)$. Then

$$x_i^k(t+1) = x_i^k(t) - f_{ij}^k(t), x_j^k(t+1) = x_j^k(t) + f_{ij}^k(t). \quad (1)$$

2. Production/destruction of the resource k in the compartment y_i in the moment t is possible if the loop $z_{ii}^k(t)$ exists (Fig. 2).

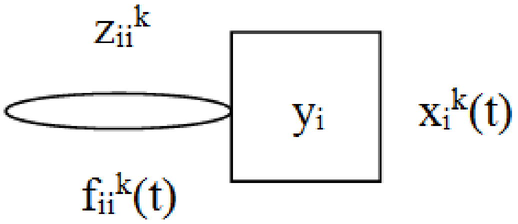

Fig.2 Production/destruction of the resource k in the compartment y_i

The case $f_{ii}^k(t) > 0$ corresponds to the production and the case $f_{ii}^k(t) < 0$ to the destruction of the resource k. The result is

$$x_i^k(t+1) = x_i^k(t) + f_{ii}^k(t). \quad (2)$$

3. A transformation of some resource into other one is possible if a vertex-transformer from the set Y_2 exists. It is the most complicated class of processes which contains a number of subclasses. The subclasses can be classified by different criterions such as

a) a simple transformation (resource k into resource l);

b) a composite transformation (one resource into several ones, many resources into one or many to many); or

A) a unit transformation (within one compartment);

B) a binary transformation (between two compartments);

C) a multiple transformation (between several compartments).

Consider the case bB as an example. Assume that initial stocks of the resource are $x_i^k(t)$, $x_i^l(t)$, $x_j^l(t)$. The transformation satisfies the equations

$$\begin{aligned} &x_i^k(t+1) = x_i^k(t) - f_{ip}^k(t), \\ &x_i^l(t+1) = x_i^l(t) - f_{ip}^l(t), x_j^l(t+1) = x_j^l(t) + f_{pj}^l(t), \end{aligned} \quad (3)$$

where y_i, y_j are compartments, y_p is a transformer.

Now consider as a more detailed example a known predator-prey model[12]

$$\frac{dx_1}{dt} = \varepsilon_1 x_1 - \gamma_1 x_1 x_2, \frac{dx_2}{dt} = -\varepsilon_2 x_2 + \gamma_2 x_1 x_2, \quad (4)$$

where $x_1(t)$, $x_2(t)$ are biomasses of the prey and predator respectively in the moment t; ε_1, ε_2 are coefficients of the natural increase of the populations; γ_1, γ_2 are coefficients of the predator-prey interaction. A representation of the model (4) by means of the dynamical digraph is shown in Fig. 3.

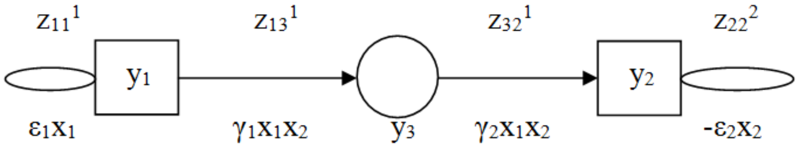

Fig.3 A representation of the predator-prey model by means of the dynamical hierarchical digraph

The loops z_{11}^1 and z_{22}^2 describe an increase of the prey biomass (resource 1) and a decrease of the predator biomass (resource 2) in the compartments y_1 and y_2 respectively, and the transformation y_3 describes a simple binary transformation of the prey biomass into the predator biomass. In the general case a natural dynamics of the system resources is represented by a balance equation for each compartment and each resource:

$$x_j^k(t+1) = x_j^k(t) + \sum_{y_i \in S_j^+(t)} f_{ij}^k(t) - \sum_{y_l \in S_j^-(t)} f_{jl}^k(t), \qquad (5)$$
$$1 \le i, j, l \le n(t), 1 \le k \le N,$$

The relation (5) added by initial data represents in fact a simulation model describing the system dynamics with consideration of its structure. The equation (5) considers both a passive regulation of the system (due to change of f_{jl}^k, $f_{mj}^k \in F_1$ in the strength of given relations) and an active one (due to choice of f_{jp}^k, $f_{qj}^k \in F_2$). The active regulation could additionally change the sets Y and Z. It is natural to name the changes of the sets X and F resource ones and the changes of the sets Y and Z structural ones. The totality of resource and structural changes determines the dynamics of the system S.

An extended state of the system S(t) is also changed by external impacts. The system environment can be represented by a vertex y_0 with the state vector $x_0(t) = (x_0^1(t), \dots, x_0^N(t))$. Respectively the set of arcs Z(t) is added by elements of the type $z_{0i}^k(t)$, $z_{i0}^k(t)$ and the set of state variables of the arcs F(t) by elements $f_{0i}^k(t)$, $f_{i0}^k(t)$, $1 \le i \le n(t)$, $1 \le k \le N$. An influence of the environment is considered in (5) without loss of generality with the condition that y_0 can belong to the sets S_j^+, S_j^-. Besides, an external impact can change the sets Y(t), Z(t). If there are several sources of impact then it is necessary to introduce several external vertices y_{01} ,..., y_{0M} , respective arcs and state variables.

Consider as an example the predator-prey model with man-made impact

$$\frac{dx_1}{dt} = \varepsilon_1 x_1 - \gamma_1 x_1 x_2 - \alpha\lambda x_1, \frac{dx_2}{dt} = -\varepsilon_2 x_2 + \gamma_2 x_1 x_2 - \beta\lambda x_2, \qquad (6)$$

where in comparison with the model (4) the characteristics of man-made exploitation of the community are added, namely an intensity of use λ and methods of use α, β. A representation of the model (6) by means of the dynamical hierarchical digraph is given in Fig. 4. In comparison with the Fig.3 to the compartments y_1 ("preys") and y_2 ("predators") the compartment y_0 reflecting the community environment (a source of exploitation) and arcs z_{10}^1, z_{20}^2 with state variables $\alpha\lambda x_1$, $\beta\lambda x_2$ are added [15].

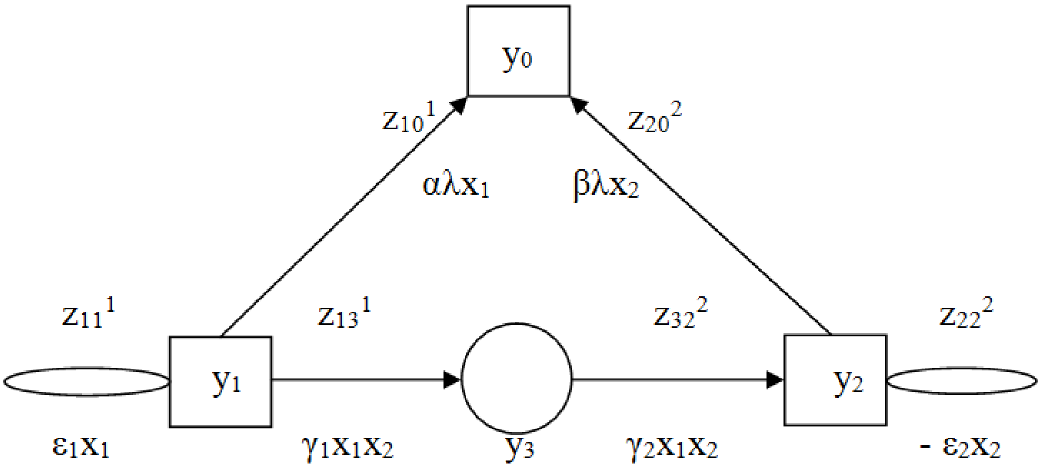

Fig.4 Modeling of exploitation in the predator-prey system by means of the dynamical digraph

Now consider CPPI-model written in a discrete form:

$$J = \sum_{i \in N} J_i \to \max; \tag{7}$$

$$s_i^t \geq 0; \quad \sum_{i \in N} s_i^t = 1; \tag{8}$$

$$J_i = \sum_{t=0}^{T} [p_i(r_i^t - u_i^t) + s_i^t c(x^t)] \to \max; \tag{9}$$

$$0 \leq u_i^t \leq r_i^t; \tag{10}$$

$$x^{t+1} = x^t + \sum_{i \in N} b_i u_i^t; \tag{11}$$

$$r_i^{t+1} = r_i^t + g_i(x^t, u_i^t); \quad i \in N; \quad t = 0,1,\ldots,T. \tag{12}$$

In this dynamic model each agent shares his resource r_i between a production of a public good (u_i) and a private activity $(r_i - u_i)$. Respectively, his current payoff is a sum of the private gain $p_i(r_i - u_i)$ and the share in the consumption of the public good $s_i c(x)$. His integral payoff is given by the formula (9), where c, p_i are continuous increasing concave functions, $c(0) = p_i(0) = 0$. The utilitarian social welfare function (7) is also introduced. If it is associated with a principal then a choice of the variables s_i s. t. (8) is considered as an economic control of the principal. The equations of dynamics are given by formulas (11)-(12), where x is a state vector, and the function of its dynamics is linear for simplicity.

The implementation of the model dynamics (11)-(12) can be given by the following algorithm:

```
given $x^0, z^0, y_i^0, r_i^0, i \in N$;
$for\ t := 0\ to\ T$
{$y_i^t := y_i^{t-1} + p_i(r_i^{t-1} - u_i^t)$;
$x^t := x^{t-1} + \sum_{i \in N} b_i u_i^t$;
$z^t := z^{t-1} + c(x^t)$;
$r_i^t := p_i(r_i^{t-1} - u_i^t) + s_i^t c(x^t) = y_i^t + s_i^t z^t$}.
```

This algorithm can be represented by a dynamic graph balanced model as follows (Fig. 5). Notice that the equation (12) is specified by means of this model. The relations for J and J_i are omitted for simplicity, they can be represented similarly.

In fact, in Fig. 5 a general relation $x_{jk}^{t+1} := x_{jk}^t + \psi(\varphi_{ijk}^t)$ is presented graphically as

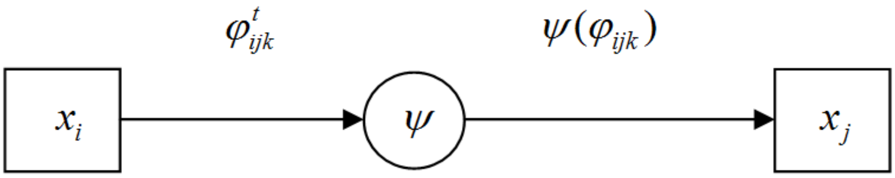

Fig.5 A balance relation

By default, it is supposed that $\psi(\varphi^t_{ijk}) = \psi\varphi^t_{ijk}$ (a linear transformation), and $\varphi^t_{ijk} = x^t_{ik}$ (a transference).

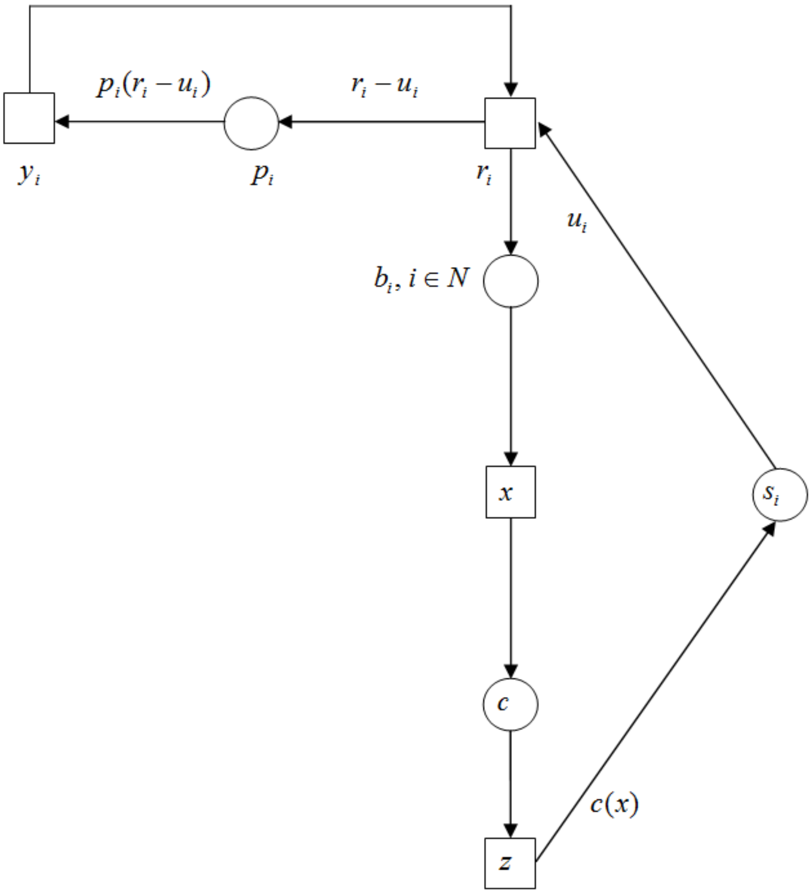

Fig.6 CPPI-model presented by the dynamic graph balanced model

3 Computer Simulation with CPPI-Models

Schematically, a process of simulation modeling with a CPPI-model can be represented as follows. Each scenario $j = 1,...,M$ includes a set of the principal's control variables $\{s^t_{i(j)}\}_{i=1}^{n}{}_{t=1}^{T}$ and a set of the agents' control variables $\{u^t_{i(j)}\}_{i=1}^{n}{}_{t=1}^{T}$. These two sequences generate the respective system trajectory $\{x^t_{(j)}\}_{t=1}^{T}$ and payoffs $\{J_{i(j)}\}_{i=1}^{n}, J_{(j)}$.

However, it is important to consider an information structure of the hierarchical game (7)-(12). To develop a classification of information structures in the hierarchical differential games with many followers three attributes characterizing the principal's strategy can be used:

1) absence/presence of a feedback of the principal's strategy on the state of a controlled dynamic system. This attribute has two basic values: open-loop strategies (OL) which depend only on the moment of time t, and closed-loop strategies (CL) which depend on the game position $(t, x(t))$ [2];

2) absence/presence of a feedback of the leader's strategy on the followers' strategies. In the first case we deal with a Stackelberg game, and games of the second type we propose to call Germeier games [10,11];

3) methods of hierarchical control. Here we differentiate compulsion, when the principal influences the followers' sets of feasible strategies, and impulsion, when the principal influences the followers' payoff functionals [15].

In turn, the followers can choose one of the three modes of behavior:

(a) isolation, when the followers act independently and come to a Nash equilibrium;

(b) cooperation, when they pool resources and combine efforts to maximize the summarized payoff functional;

(c) collaboration, when the followers voluntarily maximize the principal's payoff functional.

Notice that in the case of CPPI-models cooperation and collaboration coincide because $J = \sum_{i \in N} J_i$. To explain the proposed classification we use the following two tables.

Table 1 Basic information structures in the hierarchical games [11]

	Without a feedback on the followers' controls	With a feedback on the followers' controls
Without a feedback on the system state	Γ_{1t}	Γ_{2t}
With a feedback on the system state	Γ_{1x}	Γ_{2x}

Table 2 Maximal guaranteed payoffs of the principal for different information structures

Principal Followers	Inaction (s-const)	Impulsion	
		Γ_{1t}, Γ_{1x} (ST)	Γ_{2t}, Γ_{2x} (GER)
Isolation (NE)	J_{NE}^{0}	J_{NE}^{IMP-ST}	$J_{NE}^{IMP-GER}$
Cooperation (C)	J_{C}^{0}	J_{C}^{IMP-ST}	$J_{C}^{IMP-GER}$

In table 1 the types of leader's strategies using the denotations proposed in [11] are shown. The table 2 should be explained in more details. In hierarchical differential games the principle of optimality is a maximal guaranteed strategy of the principal with consideration of an optimal reaction of the followers. The respective maximal guaranteed payoffs of the principal for the enumerated information structures are collected in the Table 2. In the case of isolation it is supposed that the optimal reaction of the followers is their Nash equilibria set NE. In the case of cooperation the optimal reaction of the followers is the set C of points of maximum of their summary payoff functional. In this paper we consider only a case of impulsion, when the principal chooses a vector of strategies $s = (s_1, ..., s_n)$ in the modes Γ_{1t}, Γ_{1x} (Stackelberg games) or Γ_{2t} , Γ_{2x} (Germeier games). The strategies can be OL (Γ_{1t} , Γ_{2t}) or CL (Γ_{1x} , Γ_{2x}).In the degenerate case of inaction s is constant (no control).

Thus, in the case of inaction $J_{NE}^{0} = \inf_{u \in NE} J(u),\ J_{C}^{0} = \inf_{u \in C} J(u),\ J_{\max}^{0} = \inf_{u \in U} J(u)$.

In the case of impulsion for Stackelberg games we have $J_{NE}^{IMP-ST} = \sup_{s \in S} \inf_{u \in NE(s)} J(s,u)$, $J_{C}^{IMP-ST} = \sup_{s \in S} \inf_{u \in C(s)} J(s,u)$, and for Germeier games $J_{NE}^{IMP-GER} = \sup_{\tilde{s} \in \tilde{S}} \inf_{u \in NE(\tilde{s})} J(\pi(\tilde{s},u))$, $J_{C}^{IMP-GER} = \sup_{\tilde{s} \in \tilde{S}} \inf_{u \in C(\ddot{s})} J(\pi(\tilde{s},u))$, where $\tilde{S} = \{\tilde{s} : U \to S\}, \pi : \tilde{S} \times U \to S \times U$.

Now we can describe an approach to the implementation of the characterized information structures in the simulation mode. In the case Γ_{1t} strategies have the form $s(t), u(t)$. A scenario $j = 1,..., M$ represents a pair of discrete control trajectories $\{s_{i(j)}^{t}\}_{i=1}^{n}{}_{t=1}^{T}$, $\{u_{i(j)}^{t}\}_{i=1}^{n}{}_{t=1}^{T}$ for which a discrete phase trajectory $\{x_{(j)}^{t}\}_{t=1}^{T}$ and the payoffs $\{J_{i(j)}\}_{i=1}^{n}, J_{(j)}$ are calculated.

In the case of information structure Γ_{2t} strategies have the form $s(t,u(t)), u(t)$. Asarule, impulsion is implemented by a mechanism of reward and punishment of the type

$$s(t,u(t)) = \begin{cases} s^R(t), & u(t) \in U_R, \\ s^P(t), & otherwise, \end{cases} \tag{13}$$

where U_R is a set of the agents' strategies encouraged by the principal, $s^R(t), s^P(t)$ - strategies of reward and punishment respectively $(J_i(s_i^P, u_i) < J_i(s_i^R, u_i)$. In many situations it is possible to think that $s^R(t) \equiv s^R,\ s^P(t) \equiv s^P$. For each agent's control trajectory $\{u^t_{i(j)}\}_{i=1}^{n}{}_{t=1}^{T}$ belonging to the scenario $j = 1,..., M$, at each step $t = 1,..,T$ the condition (13) is checked, and the respective value $x^t_{(j)}$ and the respective summands of payoffs are calculated.

In the case of information structure Γ_{1x} strategies have the form $s(t,x(t)), u(t,x(t))$, and rules of the type

$$s^{(j)}(t,x(t)) = \begin{cases} s_1^{(j)}, & x(t) \in X_1^L, \\ s_2^{(j)}, & x(t) \in X_2^L, \\ \dots & \\ s_m^{(j)}, & x(t) \in X_m^L, \end{cases} \quad u^{(j)}(t,x(t)) = \begin{cases} u_1^{(j)}, & x(t) \in X_1^F, \\ u_2^{(j)}, & x(t) \in X_2^F, \\ \dots & \\ u_l^{(j)}, & x(t) \in X_l^F, \end{cases} \quad j = 1,..., M ,$$

(14)

where $X = X_1^L \cup ... \cup X_m^L,\ X_k^L \cap X_r^L = \varnothing,\quad X = X_1^F \cup ... \cup X_l^F,\ X_k^F \cap X_r^F = \varnothing$, should be applied step by step for all scenarios $\{s^t_{(j)}\}_{t=1}^T,\ \{u^t_{(j)}\}_{t=1}^T,\ j = 1,..., M$. The implementation of the game Γ_{2x} is more complicated and is omitted here.

It is important to note the following thing. In the majority of organizational and socio-economic systems a number of scenarios reflecting qualitatively different control strategies is quite small. In fact, for a qualitative differentiation of control strategies it is sufficient to study "strong", "moderate", and "weak" types of them. For example, if an effort is measured on a scale [0,1] then the values 0, 1/2, 1 representatively reflect the mentioned strategies. If it is not sufficient due to a reason then the values 1/4, 3/4 can be considered additionally, and so on.

Such scenarios can be called representative ones. Therefore, in the set of representative scenarios the complete enumeration becomes a practically implementable procedure.

4 Computer Simulation Support of the Identification of CPPI-Models

The problems of identification of mathematical models are solved by econometrics [6] and theory of identification [14]. But these theories have at least two essential shortages. First, their implementation requires to use long time series of reliable data of observations. But such time series are often hardly available and even absent (for example, in social processes such as corruption). Second, the standard methods solve only the problem of numerical identification, i.e. determination of the numerical values of model parameters. In the same time, in applied systems analysis a problem of structural identification is more important (which classes of functions should be used in the model).

We propose a unified computer-based approach to solving the problems of structural and numerical identification (CPPI-models are used as an example).

In mathematical modeling it is well known that to provide adequacy one must build hierarchical gradually complicated sequences of models. First, the simplest model is built which reflects only the most essential features of a modeled object. As a rule, the model contains only a few parameters and allows for an analytic investigation. After that investigation is made and some results are received, it becomes clear in which direction the model is bounded and which properties of the real system it does not describe. Then the model is perfected in the required direction, and the procedure repeats. In the limit, the respective sequence of gradually complicated models can describe the system's behavior with any required accuracy.

This methodology is used for both structural and numerical investigation of mathematical models. As for the structural identification, we always start from linear functions. Specific features of the models should always be considered. In CPPI-models the functions c and p_i are increasing, thus we use linear functions $y = ax + b\ (a > 0)$. After that, it is natural to explore power functions. In CPPI-models the functions c and p_i are concave, thus functions $y = ax^b\ (b < 1)$ are used. Again, it is natural to start from $b = 1/2$; if it is necessary then other values $b < 1$ can be analyzed. At last, other classes of continuous increasing concave functions can be taken if necessary.

In the numerical identification a feasible range of values is determined for each parameter (for example, $a_{\min} \le a \le a_{\max}$). Using the idea of representative scenarios, the values $a_{\min}, a_{\max}, (a_{\min} + a_{\max})/2$ are examined first. If it is not sufficient, a dichotomy of the segments $[a_{\min}, (a_{\min} + a_{\max})/2], [(a_{\min} + a_{\max})/2, a_{\max}]$ is made, and so on.

The idea is implemented by developing a computer software providing to a user the possibility of choice of different classes of functions and values of their parameters. The essence of the software consists in a modification of genetic algorithms for solving problems with several input parameters. The genetic algorithm decreases a number of required calculations on some orders in comparison with known numerical methods. The numerical methods give the value of a result with almost any given accuracy, but in the case of more than four input parameters too many calculations are required. In turn, genetic algorithms don't guarantee such an accuracy but practically don't depend on the number of model parameters. In the same time, the accuracy of the order 10^{-6} can be achieved which is more than sufficient for CPPI-models.

A possible development of the software includes usage of parallel calculations. Though genetic algorithms can decreases the number of calculations on some orders there are still hundreds of thousands and even millions of them, therefore paralleling seems a logical way. Each stream will contain its own genetic algorithm and input data set. The solutions of each stream are cumulated for further processing.

Another promising direction of the software development is to seek a solution not in the domain of parameters values but in the domain of types of the solved problem. A modification of the genetic algorithm of another type is required for this.

At last, the software can be developed by adding a possibility of the formation of arbitrary kinds of problems. The respective tool is parsing, i.e. a decomposition of a given term on more fine formulas which form a new term available for processing.

Thus, the described approach assumes an active usage of the methodology of applied systems analysis based on simulation modeling. If a precise analytical solution of an optimization problem is troubled then it can be investigated qualitatively on the base of scenario method using the idea of representative scenarios determined by means of systems analysis. Therefore, simulation modeling serves as a universal numerical method of solving complex mathematical problems.

5 Conclusion

In this paper a universal approach to the description of dynamics of complex systems, identification of their models, and finding the solutions by means of computer simulation is proposed (on the example of CPPI-models). The dynamic graph balance models permit to reflect in a convenient visual form the main matter-energetic processes of real-world systems dynamics. This technique is applied to CPPI-models. Computer simulation is proposed to use both for solving of complex game theoretic problems and for their identification. An idea of representative scenarios is developed in this frame. A special computer software should be developed for the implementation of the method.

References

[1] N. Nisan, T. Roughgarden, E. Tardos and V. Vazirani(2007), *Algorithmic Game Theory*, Cambridge University Press.

[2] Basar T. and Olsder G.J. (1999), *Dynamic Noncooperative Game Theory*, SIAM.

[3] Chistyakov A.E., Nikitina A.V., Ougolnitsky G.A. et al. (2015), "A differential game model of preventing fish kills in shallow waterbodies", *Game Theory and Applications*. Vol. 17.

[4] Dyachenko V.K., Ougolnitsky G.A. and Tarasenko L.V. (2014), "Computer Simulation of Social Partnership in the System of Continuing Professional Education", *Advances in Systems Science and Applications*, Vol.14, No.4, pp.27-37.

[5] Dockner E., Jorgensen S., Long N.V. and Sorger G. (2000). *Differential Games in Economics and Management Science*. Cambridge University Press.

[6] Dougherty C. (2011) *Introduction to Econometrics*. Oxford University Press.

[7] Germeier Yu. B., Vatel I.A. (1975). "Equilibrium situations in games with a hierarchical structure of the vector of criteria", *Lecture Notes in Computer Science*, 27, 460-465.

[8] Gorbaneva O.I. and Ougolnitsky G.A. (2013), "Purpose and Non-Purpose Resource Use Models in Two-Level Control Systems", *Advances in Systems Science and Applications*, Vol.14, No.4, pp.378-390.

[9] Gorbaneva O.I. and Ougolnitsky G.A. (2015) , "System Compatibility, Price of Anarchy and Control Mechanisms in the Models of Concordance of Private and Public Interests", *Advances in Systems Science and Applications*, Vol.15, No.1, pp. 45-59.

[10] Gorelov M.A. and Kononenko A.F. (2015). "Dynamic models of conflicts. III. Hierarchical games", *Automation and Remote Control*, Vol.76, No.2, pp. 264-277.

[11] Kononenko A.F. (1977), " Game-theory analysis of a two-level hierarchical control system", *USSR Computational Mathematics and Mathematical Physics*, Vol.14, No.5, pp. 72-81.

[12] Kot M. (2001), *Elements of Mathematical Ecology*. Cambridge University Press.

[13] Law A.M. and Kelton W.D. (2000) *Simulation Modeling and Analysis*. McGraw-Hill.

[14] Ljung L. (1999) *System Identification: Theory for the User*. Prentice Hall.

[15] Ougolnitsky G.A. (2011) *Sustainable Management.* Nova Science Publishers.

Corresponding author

Guennady A. Ougolnitsky can be contacted at: ougoln@gmail.com.