International Journal of Applied Engineering Research ISSN 0973-4562 Volume 12, Number 12 (2017) pp. 3455-3460

Optimization Algorithms of the Industrial Clusters' Innovative Development Programs

Maxim Dli, Artur Zaenchkovski, Dmitry Tukaev and Tatiana Kakatunova

National Research University, Moscow Power Engineering Institute, Smolensk branch, 214013, Smolensk, Russian Federation.

Abstract

The article considers the task of forming an effective program of innovative processes' management in the territorial industrial clusters as a problem of the knapsack with its own values of items' combinations for whose solving a wide class of algorithms can be used. Due to the NP-completeness of this problem and in order to reduce the time of calculations it is proposed to use approximation algorithms (such as greedy algorithm), allowing to get results close to the optimal solution. It is shown that another approach to the practical implementation of approximating algorithms can be considered an approach based on the use of artificial Hopfield network, which can be attributed to the class of optimizing networks (filters). In contrast to already known algorithms for solving the knapsack problem the algorithms which are considered additionally take into account the possible synergistic effects from the joint implementation of selected activities of the innovative processes' management program in the territorial industrial clusters.

The authors proposed the structure of an artificial neural Hopfield network and the recommendations on the choice of energy function consisting of three components: representing the target function in the form maximizing the effect of the programme activities' implementation, effectuating restrictions on the validity of the solution and maximizing the synergistic effect of selected events' combinations.

The results of computational experiments of the speed rating approximation algorithms and the degree of closeness of the result to the exact solution showed that, in practice, to solve this problem in the presence of high-speed computers and a relatively small set of events (less than 200), it is advisable to use algorithms «intelligent brute force». With tight time constraints to develop solutions for high-dimensional problems (over 500 events) and the lack of critical accuracy requirements to the result it is possible to apply a «greedy» algorithm. In the cases when in a relatively short time with high-dimensional problems it is necessary to obtain an exact solution, neural network algorithm based on Hopfield network is preferred.

Selected depending on the initial problem conditions, algorithms described in the article can be used as a component of algorithmic support of decision-making information systems for innovation management, being implemented in the industrial clusters of different types.

Keywords: Knapsack problem, artificial neural networks, greedy algorithm, algorithms «intelligent brute force», optimization of innovative programs

Currently in the global economy there are processes of innovative territorial industrial clusters (ITIC) formation, including the innovation-active industrial enterprises and organizations of innovation infrastructure. The most important role in the innovative development management of these structures plays a procedure of centralized distribution of resources to maximize the cumulative effect from implementation of innovative projects.

It is obvious that the solution of this problem is directly related to the functions of planning, coordination, monitoring and control of the innovation process that leads to the necessity of processing and analyzing a large number of information flows. To improve the implementation of these functions in the framework of decision-making support for innovation management in ITIC as the basic principles of its functioning it is advisable to use the principles of controlling, described in detail in [1, 2].

Assume, there is a basic set of activities (projects) $(Z)=\{z_i\}, i=\overline{1,N}$, that can be used in formation program for the innovation management in the ITIC. This set of measures is characterized by the vector of direct effects $(\boldsymbol{C})=\{c_i\}, i=\overline{1,N}$, their implementation and cost, formalized using the vector $(\boldsymbol{V})=\{v_i\}, i=\overline{1,N}$. As the effects of implementation can be considered the predictable revenue, the increased market share, the net present value from investments in innovation on a given time horizon, etc.

It is obvious that in practice there are always resource constraints $\boldsymbol{V}_{\mathbf{max}}$**,** defining allowable costs for the innovation management program in ITIC in general.

Form the vector $(\boldsymbol{X})=\{x_i\}, i=\overline{1,N}$, characterizing the set of elements (measures of the general population ***(Z)***), considered for its feasibility including in the innovation management programme in ITIC), given the fact that i element of this vector has the form:

$$x_i = \begin{cases} 1, \text{ if } i \text{ measure is included in the programme,} \\ 0, \text{ if } i \text{ measure is not included in the programme} \end{cases}.$$

Let it be the matrix ***[R]*** is a set of m synergistic combinations of the minimum elements of N dimension decomposition, i.e. the set of activities, joint implementation of which may give additional (synergistic) effect in the framework of the ITIC. We denote by $(\boldsymbol{S}) = \{s_j\},\ j = \overline{1, J}$ – the synergy effects vector from the implementation of two or more activities in innovation processes support in ITIC.

Consider ***(r)*** as the display ***(X)*** on ***[R]*** such that the elements

$$r_i = \begin{cases} 1, \text{ if } (\boldsymbol{X}) \cdot \left(\overset{m}{M}_j\right) = \left(\overset{m}{M}_j\right) \cdot (1) \\ 0, \text{ if } (\boldsymbol{X}) \cdot \left(\overset{m}{M}_j\right) < \left(\overset{m}{M}_j\right) \cdot (1) \end{cases} \qquad (1)$$

This display shows which of the sets ***(M)*** events included in the common set ***(Z)*** can be implemented with the predicted synergistic effect. The set ***(M)*** in this case is the column vector of the matrix ***[R]*** that specifies all possible combinations of events, and the vector ***(r)*** indicates which of the combinations considered in the vector ***(X)*** (formed the programme of events).

In this case, the task of optimizing the use of resources for development of program for the innovation management in ITIC can be represented as:

$$(\boldsymbol{X}) \to \max_{(x)} \{\boldsymbol{(C)} \cdot \boldsymbol{(X)} + \boldsymbol{(S)} \cdot \boldsymbol{(r)}\}, \qquad (2)$$

$$\text{with } \boldsymbol{(V)} \cdot \boldsymbol{(X)} \le \boldsymbol{V_{max}}.$$

This problem setting is identical to the formulation of the combinatorial optimization problem (knapsack problem) with their own values of items combinations for which there is a wide class of algorithms [3].

Figure 1 shows an example of the procedure for using the results of knapsack problem for the formation of the innovative development industrial cluster program based on the construction of this cluster's cognitive models and models for the region of its implementation. The algorithms for cognitive models constructing of regional innovative systems are considered in detail in [4,5].

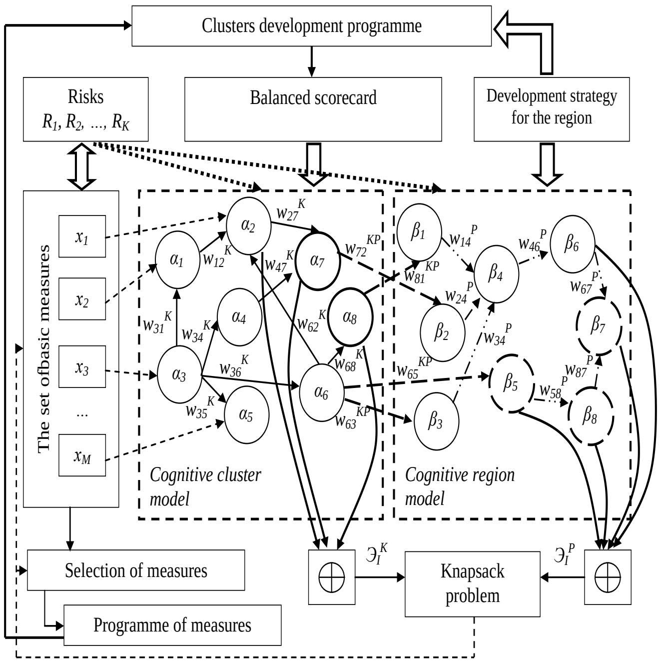

Figure 1: The procedure of knapsack results using for the formation of the program for innovative industrial cluster development

In the figure the following notation is used:

links between concepts of Cognitive cluster model,

links between concepts of Cognitive region model,

links between concepts of Cognitive cluster model and Cognitive region model,

w_{ij}^K - value of links between i and j concepts of Cognitive cluster model,

w_{lk}^{KP} - value of links between l and k concepts of Cognitive region model,

wik^P - value of links between i concepts of Cognitive cluster model and k concepts of Cognitive region model

Due to the availability of NP-completeness in knapsack problem to reduce the calculations time it is usually proposed to use approximation algorithms (such as greedy algorithm), allowing to get results close to the optimal solution. As an approach to the practical implementation of these algorithms it is possible to consider an approach based on the use of artificial Hopfield network, which can be attributed to the class of optimizing networks (filters) [6,7]. General view of the Hopfield network for this case is shown in Figure 1, where N is the number of inputs (the total number of possible events) corresponds to the number of outputs of the network, f_{lp} – characterizes the size of the connection between the l neuron with the input of R neuron. In the operation of this type networks, each neuron computes weighted sum of its inputs, generating input signal, which then with the use the nonlinear function is converted to the output vector ***(X)***.

International Journal of Applied Engineering Research ISSN 0973-4562 Volume 12, Number 12 (2017) pp. 3455-3460

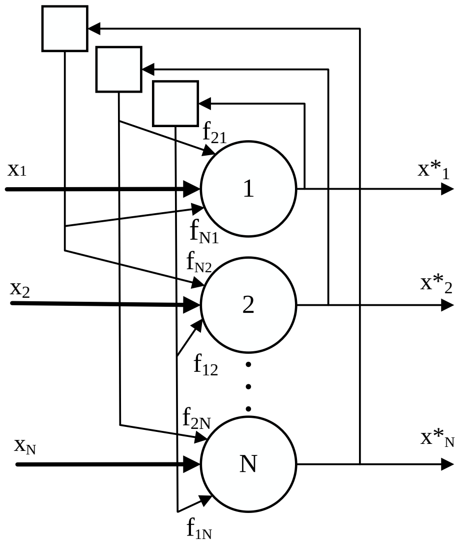

Figure 2: The Hopfield artificial neural network model

As information applying to input neurons, is considered a certain initial vector ***(X)*** describing the preliminary choice of a possible set of actions for innovation management in ITIC from the total population ***(Z)***. After a number of iterations at output of the network is formed the solution corresponding to (2).

It is known that the quality of Hopfield artificial neural network performance is greatly influenced by the choice of the energy function. It should be noted that this stage of building neural network model for solving the task is formalized insufficiently and can be based on inclusion of the energy function of three terms:

- representing the target function of the form (2);
- implementing restrictions on the admissibility decision, that is, a sum having minimum value for cases not exceeding the aggregate costs of the selected set of events beyond the total budget;
- maximizing the effect of synergistic combinations.

Given the above lack of a formalized procedure for energy functions selection of artificial neural network, the study considered various options.

For example, as the energy function of the neural network was considered an expression of the form:

$$E(x) = -A \cdot \left(\sum_{i=1}^{N} c_i \cdot x_i \right)^2 + B \cdot \left(\sum_{i=1}^{N} v_i \cdot x_i \right)^2 + D \cdot \sum_{i=1}^{N} \sum_{k=1}^{M} r_{ik} \cdot s_k \cdot x_i, \qquad (3)$$

where the coefficients A, B and D create a balance between optimal and acceptable solutions. For example, if you increase the rate A, then the solution will change in the direction of increasing the cumulative effect due to the implementation of additional measures.

When you increase the factor B to perform for limited appointments on total resource provision of the event total cost will decrease. The coefficient D determines the degree of synergistic combinations exposure of the activities undertaken in the energy function.

Despite the apparent conformity to the specified requirements to the energy function the dependence of the form (3) has no minimum corresponding to the optimal solution. This is due to the absence of restrictions from below for the first term of the expression (3) that with the operation of the network results in including in «the pack» an excessive number of events and the excess of the total budget.

One of the reasons for the negative result in selecting the type of energy functions is an unclear interpretation of the sought solution properties.

As an example, the energy function can be considered in the expression of the form:

$$E(x) = -A \cdot \sum_{i=1}^{N} c_i \cdot x_i + B \cdot \left(\sum_{i=1}^{N} v_i \cdot x_i - v \right)^2 + D \cdot \left(\sum_{i=1}^{N} \sum_{k=1}^{M} r_{ik} \cdot s_k \right) \cdot x_i, \qquad (4)$$

where the first term is responsible for the maximization of the total implemented actions effect, and the second term is for minimizing the total costs.

Analysis of the energy functions of the form (4) showed that it does not have a global minimum, and the solution is achieved through the local minimums in a situation where the «knapsack» is very close to full filling at the maximum at the total measures effect under the given total costs limits. In practice, however, due to apparent irregularities of the surface that describes the second item of the expression (3), with minor variations the set of values of ***(X)*** the magnitude of this term varies greatly.

To reduce the effect of various sizes source data on the final result of the neural network, it is advisable to normalize with transformations of the form:

$$c_i' = \frac{c_i}{\sum_{i=1}^{N} c_i}, \quad v_i' = \frac{v_i}{V_{max}}, \quad s_i' = \frac{s_i}{\sum_{i=1}^{N} c_i}.$$

In the result of a wide range of energy functions' analysis, as being effective was proposed and investigated the function of the form:

$$E(X) = A \cdot \left(\sum_{i=1}^{N} x_i \cdot c_i' - 1 \right)^2 + B \cdot \left(\sum_{i=1}^{N} x_i \cdot v_i' \right)^2 + D \cdot \left(\sum_{i=1}^{N} \sum_{k=1}^{M} x_i \cdot r_{ik} \cdot s_k' \right)^2. \qquad (5)$$

Using expression (5) we construct a connection matrix for a Hopfield network. To do this, imagine a function (5) in the quadratic form:

$$E(X)=A\left(\sum_{i=1}^{N}C_i\cdot X_i-1\right)^2+B\left(\sum_{i=1}^{N}V_i\cdot X_i\right)^2+D\left(\sum_{i=1}^{N}\sum_{k=1}^{M}X_i\cdot R_{ik}\cdot S_k\right)^2$$

$$E(X)=A\cdot\sum_{i=1}^{N}(C_i\cdot X_i)^2-2\cdot A\cdot\sum_{i=1}^{N}C_i\cdot X_i+2\cdot A\cdot\sum_{i=1}^{N}\sum_{\substack{j=1,\\ j\neq i}}^{N}C_i\cdot C_j\cdot X_i\cdot X_j+A+$$

$$+B\sum_{i=1}^{N}(V_i\cdot X_i)^2+2\cdot B\cdot\sum_{i=1}^{N}\sum_{\substack{j=1,\\ j\neq i}}^{N}V_i\cdot V_j\cdot X_i\cdot X_j+D\cdot\sum_{i=1}^{N}\left(X_i\left(\sum_{k=1}^{M}R_{ik}\cdot S_k\right)\right)^2+$$

$$+2\cdot B\cdot\sum_{i=1}^{N}\sum_{\substack{j=1,\\ j\neq i}}^{N}X_i\cdot X_j\sum_{k=1}^{M}R_{ik}\cdot S_k\sum_{k=1}^{M}R_{ik}\cdot S_k\ . \tag{6}$$

If we consider the energy function in the form as $E(x)=\frac{1}{2}(x,Qx)+(b,x)$,

so $Q_{ij}=2\cdot\left(A\cdot C_i\cdot C_j+B\cdot V_i\cdot V_j+D\cdot\sum_{k=1}^{M}R_{ik}S_k\cdot\sum_{k=1}^{M}R_{ik}S_k\right)$ and

$b_i=-2\cdot A\cdot C_i$.

As a result, in accordance with the method of Hopfield neural network construction [8], the weight matrix of connections should be set as:

$$f_{ij}=\begin{cases}-h\cdot 2\cdot\left(A\cdot C_i\cdot C_j+B\cdot V_i\cdot V_j+D\cdot\sum_{k}^{M}R_{ik}S_k\cdot\sum_{k}^{M}R_{jk}S_k\right), i\neq j\\ 1-h\cdot 2\cdot\left(A\cdot C_i\cdot C_j+B\cdot V_i\cdot V_j+D\cdot\sum_{k}^{M}R_{ik}S_k\cdot\sum_{k}^{M}R_{jk}S_k\right), i=j\end{cases}, \tag{7}$$

weight for a single continuous signal input in the form $h\cdot 2\cdot A\cdot C_i$ and the optimization step $h_T=\frac{1}{\sqrt{T}}$ (T – a (T – a counter increasing per unit after each recalculation of the network state). The stop of the cyclical process occurs when the state changes, the neural network reaches a value of 5% of the total neurons number.

The use of the Hopfield neural network with weights of the form (7) allows the first to find the solution to the knapsack problem. To improve the accuracy of the result it is appropriate to implement the adjustment of the weights based on applying the following algorithm.

On the first step are assigned the initial values to the coefficients *A*,*B* and *D*, and searched the solution. In the case of generation from the point of view non-compliance with restrictions on allowable resource costs in implementing the programme of infeasible solutions should reduce the ratio A/B. If the solution is valid from the point of constraints view, but suggests the inclusion of a relatively small number of events (which may indicate their low total efficiency), the specified proportion is increasing.

This algorithm also can be used to adjust the coefficient *D* in order to clarify the role of synergetic effect of activities joint implementation within the programme depending on the degree of importance of the program aims.

The main operation of the algorithm for constructing the model on the basis of the Hopfield network is a «walkthrough» of the network as a result of multiplying the input vector with dimension *N* with weight matrix *M*, which dimension is approximately $(N+M)^2$ operations. The total number of such «passes» is *N*, then the network goes into steady state of rest. During the adjustment of the energy functions coefficients it is possible to perform several such runs, the number of which doesn't depend significantly on the dimension of the problem. The speed performance of the building neural network models procedure can be assessed using standard evaluation functions by algorithms of the form $O(N(N+M)^2)$.

To assess the performance of the proposed algorithm was a computational experiment made. This experiment was based on the comparative assessing the results of using the above Hopfield neural network model, which is usually used to solve knapsack problem, and the results obtained by applying the known variants of «greedy» algorithms [9] and algorithms «intelligent brute force» [3].

The general idea of greedy algorithms is that at each step the most promising continuation is selected[3]. The implementation of the «greedy» algorithm for this problem involves the specific effect calculation of each *i* event $P_i=\frac{C_i}{V_i}$ and their subsequent sorting in descending order.

Attempting to turn events in the program («knapsack») is done sequentially from left to right using the sorted vector for specific effects. At the same time, a synergistic effect for the target function is summarized by the actual addition of the appropriate actions set in the program. The advantages of the greedy algorithm are simplicity of implementation and high speed performance.

Algorithm «intelligent brute force» (algorithm branch and bound) in comparison with brute force allows to avoid calculation of obviously unacceptable variants and to cut off the part of sub-optimal solutions at their early stages. According to this algorithm, solutions are constructed starting from the root of the tree for possible inclusion of events in programme, i.e. events placed in «knapsack» alternately. At each step (algorithm cycle) there is calculating of total efficiency of the measures included in the program, taking into account their costs[10].This procedure continues while resource constraints allow.

During the computational experiment, the result of those algorithms was evaluated by the degree of proximity to the exact solution. As input data were used test sets of costs, their effects, as well as synergistic effects from combinations of events. There was taken a series of 20 launches for different initial data with increasing dimensionality of the task and inputting network resource constraints. For this computational experiments were used the expansion pack Neural Network Toolbox for mathematical modeling program MatLab ver.6 on Intel Core i3.

Figures 3 and 4 show the results of using the proposed Algorithm Artificial Neural Network – ANN, greedy algorithm, brunch and bound algorithm B&B and polynomial smoothing curves.

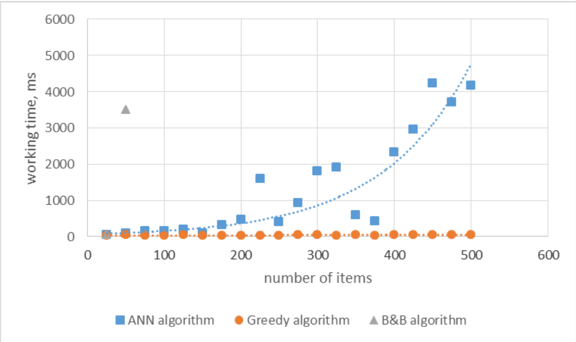

Figure 3: The dependence of considered algorithms duration on the dimension of the problem (total number of possible events)

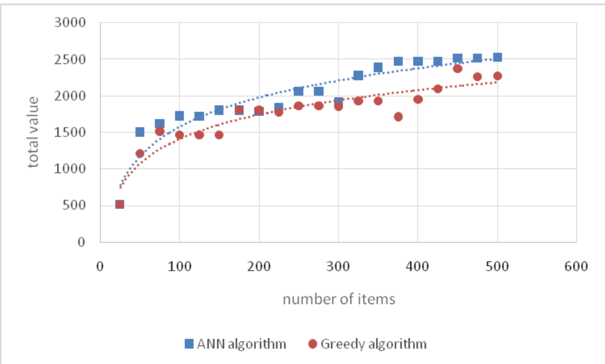

Figure 4: The dependence of cumulative effect of measures, which are included in the program using ANN and greedy algorithms, on the dimension of the problem (total number of possible events)

From figure 4 it is seen that greedy algorithm provides better performance than other considered algorithms. This is due to the fact that branch and bound algorithm (B&B) demonstrates the exponential growth of the time costs when the dimension of the problem increases. In this context the characteristics of speed performance of this method were performed only on data sets with a dimensionality of 50 and 75 (in order to obtain results in acceptable time). The smoothing curve, which was built using neural network algorithm for the obtained results and which describes the general trend of the operating time, also shows a sharp increase in time cost in the set of more than 500 events.

At the same time, from the solution accuracy point of view, greedy algorithm is lower than the neural network algorithm (Fig. 3), i.e. neural network algorithm consistently gives better quality in terms of accuracy using the same initial information. So, we can say that frequency of obtaining an exact solution for the neural network algorithm is 2.8 times higher than for the «greedy» algorithm.

Thus, we can conclude that, in practice, to solve the problem of the form (2) with high-speed computers and a relatively small set of events (less than 200), it is advisable to use algorithms «intelligent brute force». With tight time constraints to develop solutions for high-dimensional problems (over 500 events) and with the lack of critical accuracy requirements it is possible to apply a «greedy» algorithm. In cases where it is necessary in a relatively short time, to obtain an exact solution, and with high-dimensional problems it is necessary to use Hopfield networks.

The above studied algorithms used for solving the «knapsack problem» in relation to the procedure of forming of a rational set of organizational activities can be used as a component of the information system algorithmic support of innovation management decision-making support, in the industrial clusters of different types.

ACKNOWLEDGMENT

This work was supported by the State Task of the Ministry of Education and Science of the Russian Federation (the basic part, project no. 13.9619.2017/BCh).

REFERENCES

[1] Weber J., Sandt J. Erfolgdurch Kennzahlen: Neueempirische Erkenntnisse, Schriftenreihe Advanced Controlling, Bd. 21, Vallendar, 2001.

[2] Daum J. Innovation Management and the Role of Controlling. Controller Magazine, 2007, no. 6, pp. 20-25.

[3] Aho A., Hopcroft J., Ullman John. The construction and analysis of computational algorithms. - Moscow: Mir, 1979.

[4] Dli M.I., Kakatunova T.V. Three-level fuzzy cognitive model for the analysis of processes of innovative development of the region. Journal of applied computer science, 2013, no. 1 (43), pp.5-10.

[5] Dli M.I., Kakatunova T.V. The use of cognitive modeling for the analysis of complex systems. Transport case of Russia, 2013, no. 4, pp.193-195.

[6] Hopfield J.J. Neural networks and physical systems with emergent collective computational abilities, 1982, no. 79, pp. 2554-2558.

[7] Gimarov V.A., Dli M.I., Kruglov V.V. The problem of

recognizing nonstandard patterns. Journal of Computer and Systems Sciences International, 2004. no. 43(3), pp.414-418.

[8] Lazarev V.M., Sviridov A.P. Neural networks and neurocomputers. - Moscow: 2011.

[9] Cormen, T., Leiserson, C., Rivest, R., Stein, K. Chapter 16. Greedy algorithms Algorithms: construction and analysis. - Moscow: Williams, 2005.

[10] Martello S., Toth P. Knapsack problems: algorithms and computer implementations. John Wiley & Sons Ltd., 1990.