The Algorithm for Selection of Information Risk Management Mechanisms

Victor Zavgorodniy, Pavel Lukyanov, Stanislav Nazarov

The Financial University under the Russian Federation Government

Corresponding author. . Tel.: +7-499-277-21-13; fax: +7-499-157-7070

E-mail address: zvi@rambler.ru

Research task

We can formalize the task of mechanism selection as follows.

Assumed:

1) The set of risks $R = \{ r_1, r, \ldots, r_N \}$;

2) The set of damages in descending order

$$U = \{ u_{r_1}, u_{r_2}, \ldots, u_{r_N} \};$$

3) The set of information risk control mechanisms $\boldsymbol{M}=(m_1,m_2,\ldots,m_K)$;

4) Each *k-th* control mechanism is characterized by $\boldsymbol{R_k}$, $\boldsymbol{E_k}$ parameters sets as well as by the c_k parameter:

- The $\boldsymbol{R_k}=(r_1, r_2, \ldots, r_J)$ set consist of information risks to be controlled by the k mechanism;

- The $\boldsymbol{E_k} = (e_{k1}, e_{k2}, \ldots, e_{KN})$ set of indices estimates the efficiency of the *k*-th control mechanism. The e_{kn} value varies within the $0 \le e_{kn} < 1$ limits and shows the part of damage caused by the *n*-th information risk that will be eliminated when using the *k*-th control mechanism. The efficiency indices of all control mechanisms are contained in the matrix E.

The matrix E characterizes efficiency of all control mechanisms.

$$E = \begin{vmatrix} e_{11} & e_{12} & \dots & e_{1N} \\ e_{21} & e_{22} & \dots & e_{2N} \\ \dots & \dots & \dots & \dots \\ e_{K1} & e_{K2} & \dots & e_{KN} \end{vmatrix}$$

We used a multiplicative index:

$$\prod_{k=1}^{K}(1-e_{kn}) = (1-e_{1n})(1-e_{2n})\dots(1-e_{Kn})$$

This index characterizes the common part of damage caused by the *n* risk that will remain after all *K* control mechanisms are applied.

The c_k parameter denotes costs of acquisition or upgrade, development, creation and promotion of the *k*-th mechanism.

In practice, some mechanisms are incompatible. They cannot be used in the system due to their incompatibility. Compatibility of mechanisms is set by a compatibility matrix:

$$D = \begin{vmatrix} d_{11} & d_{12} & \dots & d_{1K} \\ d_{21} & d_{22} & \dots & d_{2k} \\ \dots & \dots & \dots & \dots \\ d_{K1} & d_{K2} & \dots & d_{KK} \end{vmatrix}$$

$$d_{ij} = \begin{cases} \dfrac{1,\ if\ mechanism\ number\ i\ and\ mechanism\ number\ j - are\ compatibility;}{0 - otherwise} \end{cases}$$

The set of mechanisms included into the information risks management system (IRMS) is defined as a binary configuration vector

$$X = (x_1, x_2, \ldots, x_K).$$

The vector components accept the following values:

$$x_k = \begin{cases} 1, if\ \ mechanism\ \ number\ k\ enter\ IRMS \\ \hline 0 - otherwise \end{cases}$$

The control mechanisms $x_i\ , x_j \in X$ are compatible if the following condition is satisfied:

$$x_i x_j \le d_{ij}, i = \overline{1, K},\ j = \overline{1, K}$$

.

The common damage, U^o, expected after introduction of control mechanisms into the IRMS will be referred to as remaining damage.

$$U^o(x_1, x_2, ..., x_K) = \sum_{n=1}^{N} u_{r_n} \prod_{k=1}^{K} (1 - e_{kn} x_k)$$

To define a binary vector $X^* = (x_1^*, x_2^*, ..., x_K^*)$, which provides a minimum of the summary costs of mechanisms use and residual damage from all significant risks:

$$\sum_{k=1}^{K} (c_k x_k^*) + \sum_{n=1}^{N} u_{r_n} \prod_{k=1}^{K} (1 - e_{kn} x_k^*)$$

Mathematical Definition of a Problem Related to Selection of Information Risk Protection Mechanisms

Let us define a binary vector $X^* = (x_1^*, x_2^*, ..., x_K^*)$ which provides the minimum common costs of using control mechanisms and residual damage caused by all significant risks:

$$\sum_{k=1}^{K}(c_k x_k^*) + \sum_{n=1}^{N} u_{r_n} \prod_{k=1}^{K}(1 - e_{kn} x_k^*)$$

given that .

$$x_i^* x_j^* \le d_{ij},\ i = \overline{1, K},\ j = \overline{1, K};$$

$$\sum_{k=1}^{K}(c_k x_k^*) \le C_{\max}$$

For the solution of this problem we suggest the method of a modified greedy algorithm.

According to this method, a mechanism ensuring the maximum effect is selected at each step.

The effect is determined by the difference between the cost reduction resulting from the use of a respective mechanism and a lost opportunity.

The lost opportunity is understood as impossibility to use at subsequent steps

mechanisms incompatible with the mechanism, included in the system.

Also taken into account are cost constraints related to the application of information risk management mechanisms.

Introduced designations

h –the number of an implemented algorithm step;

$X_h(x_{h1}, x_{h2}, \ldots, x_{hK})$ – the configuration vector status after the h-th algorithm step;

$W^1(h)$ – a set of mechanisms included into the system after h algorithm steps;

$S^1(h)$ – mechanisms not yet included into those used at the h-th step of the algorithm but compatible with mechanisms of the $W^1(h)$ set;

$\Omega^1(h)$ –a set of mechanisms incompatible with the of $W^1(h)$ set, i.e. excluded from further consideration;

$U_n^o(h)$ – a residual damage from n-th risk after h algorithm steps/

Let $m_{h+1} \in S^1(h)$ – be a mechanism selected at step $h+1$ from the $S^1(h)$set.

Assume that the k_{th} component corresponds to the selected mechanism m_{h+1} in a vector X_h .

Then the value by which the damage from the n-th risk will decrease in case the k-th mechanism m_{h+1} is selected at the step $h+1$ is equal:

$$\Delta U_n(h+1,k) = U_n^o(h)e_{kn} \quad .$$

The remaining damage caused by the nth risk will be equal to

$$U_n^o(h+1,k) = U_n^o(h)(1-e_{kn}).$$

The total reduction of damage caused by all type risks in case the *k*-th mechanism is selected at the step *h+1* is expressed with the following formula:

$$\Delta U(h+1,k) = \sum_{n=1}^{N} \Delta U_n(h+1,k) = \sum_{n=1}^{N} U_n^o(h) e_{kn}$$

Let us define the lost damage reduction opportunity at subsequent algorithm steps

as $\Delta U_{\tau}^{-}(h+1,k)$

The lost opportunity is determined by impossibility of subsequent use of the τ

mechanism incompatible with the k mechanism $k \ (\tau \in \Omega^{1}(h))$.

Then $$\Delta U_{\tau}^{-}(h+1,k) = \sum_{n=1}^{N} U_{n}^{o}(h)(1-e_{kn})e_{\tau n}\bar{d}_{k\tau}s_{h\tau}^{1}$$

$\bar{d}_{k\tau}$,

Where $s_{h\tau}^{1}$ is the inverse $d_{k\tau}$ value from the compatibility matrix *D;*

multiplier $s_{h\tau}^{1}$ $s_{h\tau}^{1}$ =1, if $\tau \in S^{1}(h)$ and $s_{h\tau}^{1}$ =0 – otherwise.

The presence of the $s_{h\tau}^{1}$ multiplier allows the τ mechanism to be taken into account at step *h+1*, which mechanism has become incompatible only at step *h+1* as a result of including the *k* mechanism.

The integrated lost damage reduction opportunity in case the *k* mechanism is selected at step *h+1* is equal to:

$$\Delta U^{-}(h+1,k)=\sum_{\tau=1}^{K}\sum_{n=1}^{N}U_n^{o}(h)(1-e_{kn})e_{\tau n}\overline{d}_{k\tau}s_{h\tau}^{1}$$

Let the value *Э(h+1,k)* be entered to estimate the effect of including the k control mechanism at step h+1*k*:

$$Э(h+1,k)=\Delta U(h+1,k)-(\Delta U^{-}(h+1,k)+c_k)$$

Then the mechanism selection algorithm is represented by a sequence of steps.

At each step h, the $Э_y(h+1,k)$ is calculated for $m \in S^1(h)$. In this case the k*th mechanism m^* is selected for which the $Э(h+1,k^*)$ = max and $\sum_{k=1}^{K}(c_k x_k^*) \le C_{max}$

If such a mechanism is unavailable, the algorithm implementation process is completed and the current value of the $X^*(x_1^*, x_2^*,..., x_K^*)$ vector is taken as the optimum value.

To increase the accuracy of the algorithm, we change the computation procedure of the value $\Delta U^-(h+1,k)$.

At each step, we define the dv quantity of mechanisms with the maximum values of $\Delta U(h+1,k) - c_k$ that may become incompatible after the k mechanism selection.

Experimentally it was established that the highest algorithm accuracy is reache

if the dv value falls within the interval $\frac{1}{4}K < dv < \frac{1}{3}K$

When the simulation was performed in the field of the full search algorithm applicability (to 30 mechanisms) the maximum relative error did not exceed 7%, and the mean relative error equaled 0,84%.

We compared the full search algorithm, the modified greedy algorithm and the genetic algorithm

The full search method works in real time if the number of mechanisms ranges within $1<k<20$. In case $k=27$, the time reaches 45 minutes (on the personal computer).

The modified greedy algorithm showed the highest performance rate practically independent of the task dimensionality and the level of mechanisms incompatibility.

At $k=100$ the algorithm runtime does not exceed 1 min.

The mean relative error does not exceed 5% over the whole operating range of basic data. The maximum relative error does not exceed 15%.

The genetic algorithm in terms of the implementation time is positioned between the modified greedy algorithm and the full search algorithm.

The algorithm operating time has a linear dependence on the task dimensionality and does not exceed tens of minutes in all operating ranges of basic data.

The accuracy of the algorithm depends on both the task dimensionality and a level of mechanisms incompatibility.

The best results related to accuracy and time of implementation are obtained in case of an insignificant level of incompatibility (less than 1%).

Given that there are 200 individuals in a population and 1000 cycles, the relative error of results does not exceed 3%.

At medium and high levels of mechanisms incompatibility, the method shows the worse accuracy results than the results of the modified greedy algorithm.

Thank you for attention.

Dr.Econ.Sci. Victor Zavgorodniy, professor of the Financial University under the Russian Federation Government

Tel.: +7-916-828-30-54

E-mail address: zvi@rambler.ru