12. Hanaev, V. V. Potrebiteli – reguljatory: vozmozhnosti i perspektivy primenenija [Text] / V. V. Hanaev // Nauchno – tehnicheskie vedomosti SPbGPU. – 2008. – Issue 1. – P. 59–63.
13. Pravila ustrojstva jelektroustanovok [Text]. – 6-e izd. – Moscow: Glavgosjenergonadzor Rossii, 1998. – 608 p.
14. Stevenson, W. J. Production/Operations Management: With Standard Cd-Rom Package (Irwin/Mcgraw-Hill Series Operations and Decision Sciences) [Text] / W. J. Stevenson. – Richard D Irwin, 1998. – 912 p.
15. Khotskina, V. B. Automatization of management processes by the first stage of ore enrichment with the usage of quickly acting regressive models [Text] / V. B. Khotskina // Naukovyj visnyk Nacional'nogo girnychogo universytetu. – 2014. – Issue 3. – P. 145–151.
16. Len'shin, V. N. Proizvodstvennye ispolnitel'nye sistemy (MES) – put' k jeffektivnomu predprijatiju [Electronic resource] / V. N. Len'shin, V. V. Kuminov // Sredstva i sistemy komp'juternoj avtomatizacii. – Available at: http://asutp.ru/?p=600359
17. Kharlamov, A. A. Attention mechanism usage to form framework structures on a semantic net. Vol. 11 [Text] / A. A. Kharlamov; A. V. Holden, V. I. Krynkov (Eds.) // Neurocomputers and Attention. Connectionism and neurocomputers. – Manchester-New York: Manchester University Press, 1991. – P. 747–756.
18. Shubladze, A. M. Sintez optimal'nyh linejnyh reguljatorov [Text] / A. M. Shubladze // Avtomatika i telemehanika. – 1984. – Issue 12. – P. 22–23.

В управлінні проектами ефективність роботи команди часто пов'язують з її балансуванням в термінах командних ролей. Для розуміння закономірностей переходів учасників команди між командними ролями запропоновано використовувати апарат ланцюгів Маркова. За допомогою моделювання виконано дослідження і показано, що для команди проекту з розробки програмного забезпечення природним є прагнення збалансувати командні ролі

Ключові слова: управління проектами, формування команди проекту, командні ролі Белбіна, ланцюги Маркова

В управлении проектами эффективность работы команды часто связывают с ее балансированием в терминах командных ролей. Для понимания закономерностей переходов участников команды между командными ролями предложено использовать аппарат цепей Маркова. С помощью моделирования выполнено исследование и показано, что для команды проекта по разработке программного обеспечения естественным является стремление сбалансировать командные роли

Ключевые слова: управление проектами, формирование команды проекта, командные роли Белбина, цепи Маркова

UDC 005.32
DOI: 10.15587/1729-4061.2017.91597

EXAMINING THE ATTRIBUTES OF TRANSITIONS BETWEEN TEAM ROLES IN THE SOFTWARE DEVELOPMENT PROJECTS

V. Liubchenko
Doctor of Technical Sciences, Associate Professor
Department of System Software*
E-mail: lvv@edu.opu.ua
I. Sulimova
Head of laboratory
Training and production laboratory
"Modern business technologies" of the Center of Partnership with Enterprises*
E-mail: julia.sulimova@gmail.com
*Odessa National Polytechnic University
Shevchenko ave., 1, Odessa, Ukraine, 65044

1. Introduction

Team development in software engineering requires the application of team-oriented methods for project management. Team roles are an important factor that affects the success of projects in the field of software engineering [1].

One of the practical tools for a project manager when forming a team is the analysis of the Belbin's team roles, who in his work tried to answer the question on why projects are successful or unsuccessful [2]. He demonstrated that effective work of a team is connected to its balance in terms of team roles of members of the team. Although the original work dealt with the overall project management, it was demonstrated [3] that it is relevant, in particular, for the area of software engineering.

In general, observing the behavior of representatives of particular team roles made it possible to identify strong and weak features of each of the roles, as well as general patterns of interactions between roles. All this is of significant interest for the field of software engineering. Taking into account the peculiarities of separate team roles, both at the stage of forming and in the process of work, allows overcoming the difficulties associated with the shortcomings linked to the team imbalance, maximizing individual efficiency and preventing possible sources of conflict within the team.

2. Literature review and problem statement

An ability to build effective teams is very important for many organizations in the software development [4]. As stated in [5], an important factor in ensuring the team of developers' best results for the projects is the formation of team from members with correct roles and appropriate characteristics. This issue is particularly relevant for small and medium-sized enterprises because they operate under limited budget and cannot select employees from a large pool of candidates when forming the teams for specific projects.

Many researchers paid attention to the fact that the formation of teams in the industry of software development is expedient to perform with regard to the theory of team roles by Belbin. In particular, article [6] argues that the distribution of team roles between participants has the importance for determining the structure of the team equal to the distribution of knowledge and skills. Paper [7] presents results of empirical studies that confirmed the relationship between successful fulfillment of particular technical tasks and the peculiarities in the behavior of team members.

Based on the numerical observations of group work and the means for measuring success of the teams, Belbin identified eight team roles that describe the needs and capabilities of both the teams and their individual members. Article [8] demonstrated that the full set of team roles provides for the largest probability of success of the team, although certain situations and problems mostly fit to individual roles. Let us consider briefly the attributes of each of the roles.

Implementer (IMP) turns concepts into working procedures, has rather narrow and deep professional competences, high level of discipline, persistence and the ability to plan all activities. Completer-Finisher (CF) manages completing all launched plans, strives to and actually finds unmet needs of the team and takes on the tasks that others trying to avoid. Shaper (SH) is the versatile loner who may potentially fulfill the entire project independently, with strict rules of motivation, which ensures the implementation and completion of the tasks. Plant (PL) is the source of creative and original ideas for the team who deals with complicated challenges and problems. Monitor-Evaluator (ME) is the analyst, with a strategic vision who steers the team away from conceptual errors and reveals the flaws in the plan before they adversely affect the execution of the project. Resource Investigator (RI) provides a team with external resources as well as the Plant – with internal ideas and resources. Coordinator (CO) is the positive motivator for the team who knows well the competences of all team members, who organizes them to achieve a common goal. Team Worker (TW) is a member of the team that smoothes contradictions in the team and helps participants to collaborate successfully.

Roles can be grouped by different characteristics [8]. For example, by the direction of major efforts, we can distinguish between the groups of managing (Resource Investigator, Coordinator and Team Worker), executive (Shaper, Implementer, and Completer-Finisher) and mental (Plant and Monitor-Evaluator) roles.

One of the most important results, formulated by Belbin, is the assertion that the groups balanced in terms of team roles demonstrate better performance and more success than the non-balanced groups. That is, for effective work of the team it is necessary for every its participant to possess own strong traits that do not repeat strong features of other members of this team. This is confirmed by the numerical studies. In particular, paper [9] presents statistical data to confirm that the balance of the team positively affects quality of the processes within the team. Article [10] demonstrated that the more complete the set of team's roles is, the higher the team's capability to solve the tasks it faces, especially at the initial stages of the project.

One should consider a perfectly balanced team to be the team in which all team roles are presented at the same level, while an ideally imbalanced team is the team in which all the participants represent one role. There are, however, studies that criticize assertion about the balance. For example, article [11] noted that the software development project teams require two roles – Shaper and Plant. Usually, the role of Shaper matches the leader of the team, and the role of Plant suits the rest of the team members. It should be noted that the authors of this study do not prove that only these two roles are sufficient for teams. That is why, it is more correct to say that the team of software developers should be balanced and the presence of participants with the team roles of Shaper and Plant is obligatory for it.

To define the team roles, a Self Perception Inventory questionnaire is usually used, which was proposed by Belbin for measuring the behavioral characteristics demonstrated by team members. This questionnaire assesses the degree of individual susceptibility to each of the roles. Article [12] establishes that the questionnaire on self-perception yields satisfactory results to be used in the formation of project teams.

Contribution of each participant to the teamwork is described by primary and secondary roles. A primary role is the behavior demonstrated by participant under normal circumstances. He may take on a secondary role, for example, under condition of special need of the team in the role that is not primary for any team member. The reason for the transition can also be due to a conflict of interests between two participants with the same primary roles.

Therefore, to establish effective balance in the existing team, it is necessary not only to encourage the team members to recognize and utilize their strong traits but to assist some team participants in taking on secondary roles. This allows preventing the duplication of roles, meeting the needs of the team or solving certain problems. Transitions between roles may also be needed in order to help participants expand their skills and the range of team roles.

One should pay attention to the fact that most studies into the balance of team roles with the attributes of the team were conducted using the observations over artificially or naturally created teams and statistical processing of results of the research. Although the application of mathematical methods may also yield useful results. For example, article [13] proposed a mathematical model for the balanced team. Applying this model simplifies consideration of interrelations between the roles when forming a balanced team.

As for the attributes of transitions between primary and secondary roles for the purpose of balancing the project team, then all the explanations and conclusions were produced verbally and are not confirmed by either theoretical or empirical research. The reason for this, obviously, is the complexity of designing the experiments under real conditions. One possible means to overcome the difficulty is the application of computational experiments.

3. The aim and tasks of the study

The aim of the conducted study is to verify, based on computational experiments, a nature of balancing the team roles when performing a software development project.

To achieve the set aim, the following tasks were formulated:

– to build a model of transitions between team roles and to define parameters of the model;

– to conduct computational experiments to examine the attributes of transitions between team roles when executing a project and to analyze obtained results.

4. A method for constructing the model of transitions between team roles

In the system of project management, people are the most changeable and unpredictable components. A reaction of the team members towards control actions is always uncertain to some extent. Therefore, we can say that the process of transitions between team roles is a random process. Human behavior, in addition to inherent properties (for example, primary and secondary team roles), is defined by internal and external factors that influence every day and by the combinations, which changed on a daily basis. That is why we can state that the reaction of a team member towards control actions depends only on the condition under which he happens to be now, and does not depend on what happened before. That is, a Markov random process is the adequate model for the process of transitions between team roles. Since the transition between team roles occurs as a reaction to control actions, then one should choose as the basis for a model that is being developed a discrete Markov chain [14].

In the proposed model, the Markov chain states correspond to individual team roles: S_1 – implementer, S_2 – completer-finisher, S_3 – shaper, S_4 – plant, S_5 – monitor-evaluator, S_6 – resource investigator, S_7 – coordinator, S_8 – team worker.

A modeling of transitions between the roles is carried out by introducing ties between states S_1 – S_8, which make it possible to determine quantitative characteristics for the states and transitions between them. Each tie corresponds to a transition between roles whose opportunities and probabilities are defined based on results of the self-perception questionnaire and experts' survey. All transitions from state S_i form a complete group of events, that is, the probability of occurrence of these events is equal to unity:

$$\sum_{j=1}^{8} p_{ij} = 1, \forall i \in \{1, 2, \ldots, 8\}, \tag{1}$$

where p_{ij} is the probability of transition from state S_i to state S_j.

Since the primary and secondary roles are determined by personal characteristics of the team members, then the probability of transitions between states do not change over time, that is, a Markov chain is uniform [15].

Probability $P_i(k)$ of state S_i corresponds to the probability of corresponding team roles in step k. By step k we understand a certain control action, which made a team member change the role. The sum of probabilities of all states $P_i(k)$ at each step k is equal to unity:

$$\sum_{i=1}^{8} P_i(k) = 1, \forall k \in \{1, 2, \ldots, \infty\}, \tag{2}$$

where $P_i(k)$ is the probability of ith state in step k.

For a uniform Markov chain, determining the probabilities of states in (k+1)th step is performed in a general case by formula of total probability:

$$\begin{Vmatrix} P_1(k+1) & P_2(k+1) & \ldots & P_8(k+1) \end{Vmatrix} = \\ = \begin{Vmatrix} P_1(k) \\ P_2(k) \\ \vdots \\ P_8(k) \end{Vmatrix}^T \times \begin{Vmatrix} p_{11} & p_{12} & \cdots & p_{18} \\ p_{21} & p_{22} & \cdots & p_{28} \\ \vdots & \vdots & \ddots & \vdots \\ p_{81} & p_{82} & \cdots & p_{88} \end{Vmatrix}. \tag{3}$$

After constructing a structural model for the transitions between roles in the form of a Markov chain, it is necessary to perform parametric identification of the model.

First, using the self-perception questionnaire, we define initial probabilities for each of primary roles $P_i(0)$, as well as the possible transition frequency for specific pairs of roles.

Next we run a group expert evaluation of probability values p_{ii} for that a team participant will not change his role after control action. Accordingly, the probability of changing a team role is $(1-p_{ii})$. In order to obtain the probabilities of transitions to specific roles, this magnitude should be divided in proportion to the transition frequency defined previously.

A parameterization of the model allows us to reflect by using a Markov chain a conventional behavior of software development team members.

5. Results of modeling the transitions between team roles

To identify the possibilities and probabilities of transitions between team roles, by using self-perception questionnaires, a survey of 119 persons was conducted, different in age (from 22 to 36), gender, work experience (students-interns, junior, middle and senior staff) and job responsibilities (developers or teams leaders). Using the self-perception questionnaire, all participants independently defined their primary and secondary roles. As a result, we received 119 vectors of team roles $<r_1, r_2, ..., r_8>$, where r_i takes the value of 1 if S_i corresponds to the primary role, it takes the value of 2 if S_i corresponds to the secondary role, and it takes a value of 0 for the rest of the cases.

Based on the obtained vectors of team roles, we defined initial probabilities of each of primary roles as

$$P_i(0) = \frac{N_{r_i=1}}{N}, \tag{4}$$

where $N_{r_i=1}$ is the number of vectors in which *i*th role is the primary one, N is the total number of analyzed vectors. The results obtained are shown in Fig. 1.

In addition, the vectors of team roles were used to define the structure of a Markov chain. The existence in one vector of combination r_i=1 and r_j=2 reveals the possibility of transition in one step from state S_i to state S_j. Accordingly, taking into account all vectors, we can also calculate a transition frequency between states. Determining the probability of transitions is not possible until there remains uncertain probability of maintaining the same state, that is, the probability that the role will not change following a control action.

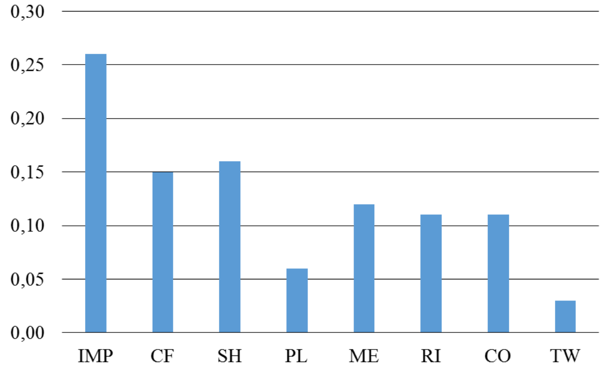

Fig. 1. Values of initial probabilities of team roles of implementer (IMP), completer-finisher (CF), shaper (SH), plant (PL), monitor-evaluator (ME), resource investigator (RI), coordinator (CO) and team worker (TW)

To visualize the received model, we shall represent all states of the system and the transitions between them as a weighted directed graph. Vertices of the graph correspond to the states of the system, arrows – to possible transition of the system from one state to another in a single step, scales – to appropriate transition probabilities (Fig. 2).

A group assessment of determining the probabilities of transitions between roles included as experts successful team leaders and project managers from the companies engaged in product or outsourced software development. In these companies, the number of employees does not exceed 100 persons and so the question of effective team building is relevant for them.

As a result of parameterization of the model, we received the following matrix of transition probabilities:

$$P=\left\|\begin{matrix}0,85 & 0,10 & 0,05 & 0 & 0 & 0 & 0 & 0\\ 0,20 & 0,70 & 0,10 & 0 & 0 & 0 & 0 & 0\\ 0,05 & 0,05 & 0,80 & 0 & 0,05 & 0 & 0 & 0\\ 0 & 0 & 0 & 0,60 & 0,20 & 0,20 & 0 & 0\\ 0 & 0 & 0,05 & 0,15 & 0,65 & 0,15 & 0 & 0\\ 0 & 0 & 0 & 0,10 & 0,10 & 0,75 & 0,05 & 0\\ 0 & 0 & 0 & 0 & 0 & 0,05 & 0,85 & 0,10\\ 0 & 0 & 0 & 0 & 0 & 0 & 0,10 & 0,90\end{matrix}\right\|.$$

Expression (3) provides for the possibility of performing practical calculations for the model. Simulation results are shown in Fig. 3.

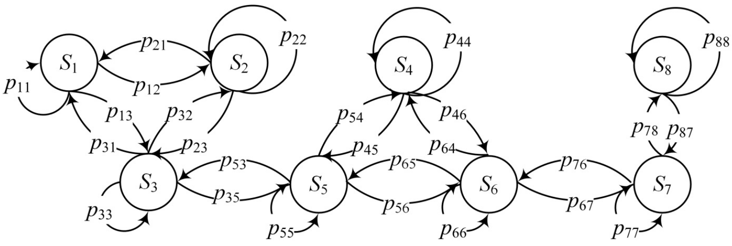

Fig. 2. Marked graph of a Markov chain for the simulation of transitions between team roles in software development projects

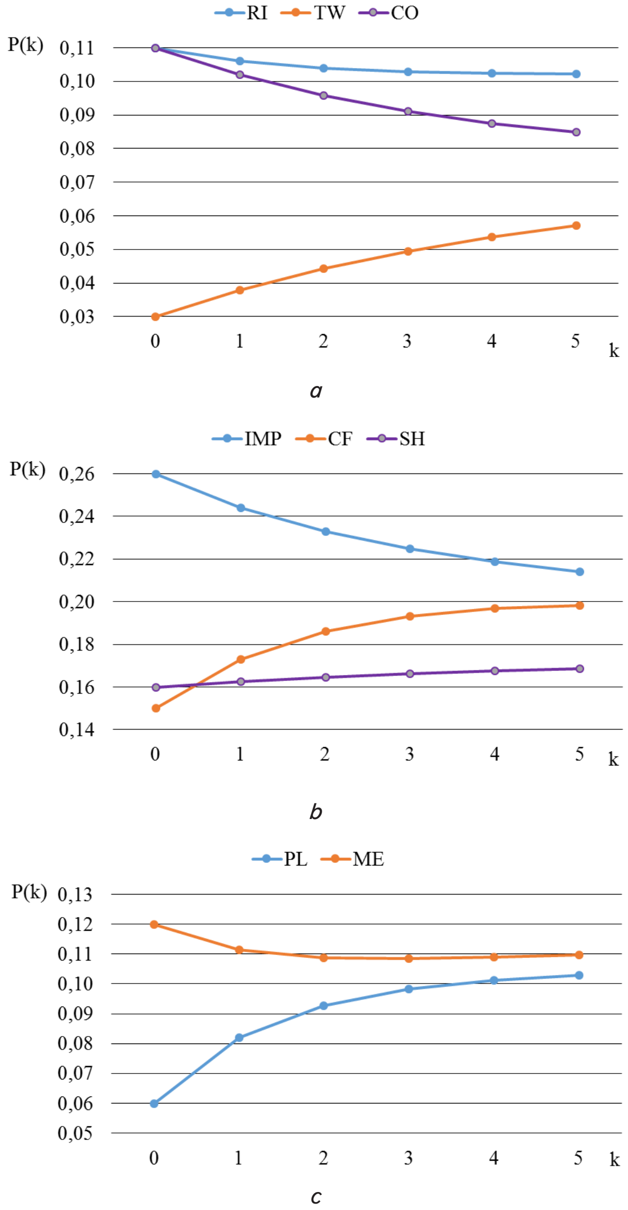

Fig. 3. Change in probabilities P(k) of team roles according to steps k:
a – a set of management roles; *b* – a set of executive roles; *c* – a set of mental roles

As was noted above, the necessary roles for software development teams are those of Shaper and Plant. Let us consider using mathematical modeling what happens if a team lacks representatives of these roles. For this purpose, we shall run two experiments in which, in the beginning of modeling (k=0), we take the according probability, $P_3(0)$ for the role of Shaper and $P_4(0)$ for the role of Plant, as equal to zero. Note that probability $P_i(k)=0$ indicates that in kth step the system will not enter state S_i. Modeling results are shown in Fig. 4.

Fig. 4 shows that, under such conditions, there is a probability of transition of some persons to their secondary roles, in order to compensate for the lack of key skills and characteristics in the project team.

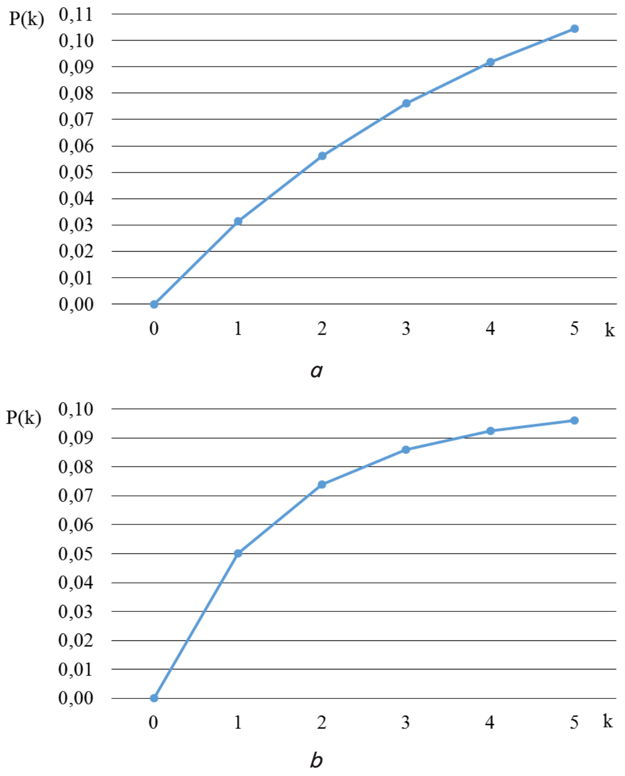

Fig. 4. Change in probabilities P(k) of roles: *a* – Shaper; *b* – Plant

6. Discussion of results of modeling the transitions between team roles

In the course of running the model experiments, we verified two working hypotheses:

H_1 – it is natural for a software development project team to strive for the balance in team roles;

H_2 – when representatives for the necessary roles are missing, their substitution is carried out within the team.

As noted above, it is a desirable pattern that a project team includes representatives of all the roles, that is, there is a balance of team roles. In terms of state probabilities of the Markov chain, a perfectly balanced team is matched by a situation where the probabilities of all states are the same. However, right from the start, the probabilities of states differ significantly (Fig. 2). If hypothesis H_1 is valid, then the difference between the probabilities of states in the Markov chain, that is, of corresponding team roles, will decrease over the course of the project.

Fig. 3 shows that in the process of project execution, probability values are drawn together, that is, hypothesis H_1 is true and team members will swap team roles to balance the influence of different roles in the team.

In addition, to verify hypotheses H_1, we shall analyze a change in the value of probability dispersion in the first steps of the project implementation (Fig. 5).

As can be seen from Fig. 5, the dispersion in the first five steps is reduced almost by two times, which also allows us to render hypothesis H_1 true.

Fig. 4 shows the validity of hypothesis H_2 is true, where one can see that, already in the first step, the probabilities of corresponding states are non-zero. Additionally, we shall consider how the probability for each of the necessary roles changes under different initial conditions (Fig. 6).

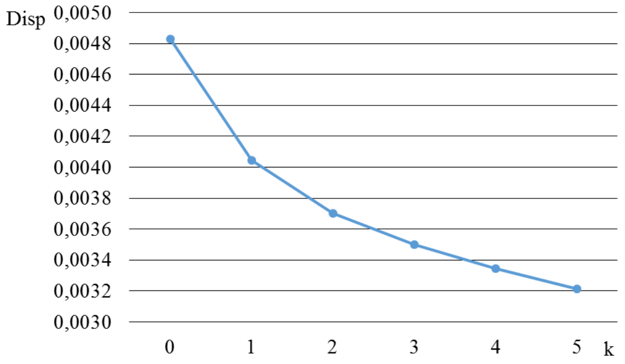

Fig. 5. Change in probability dispersion of team roles: k – steps, Disp is the probability dispersion

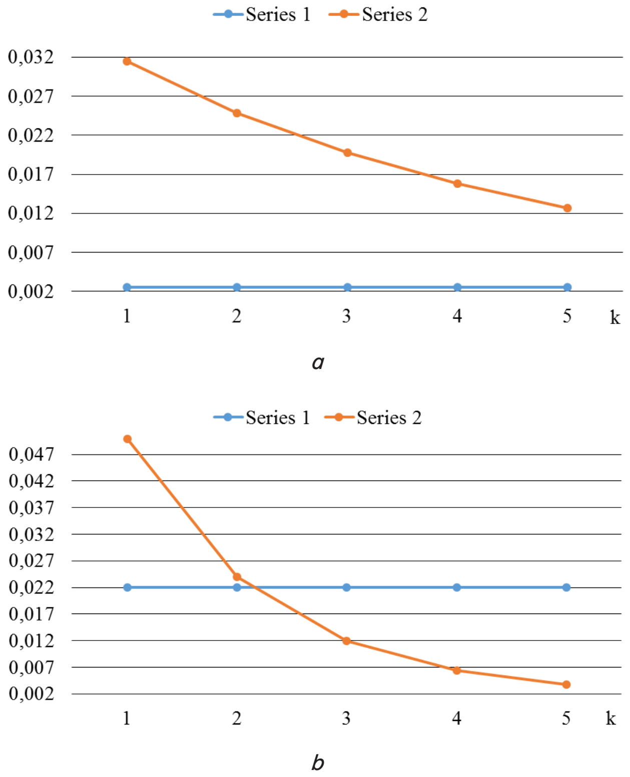

Fig. 6. Increment in the value of probabilities in the presence (series 1) and absence (series 2) of representative of the role in a team from the start of the project: *a* – for the role of Shaper; *b* – for the role of Plant

As can be seen from the performance of series 1 and 2 in Fig. 6, the nature of increment in the probabilities is significantly different. If, from the start, the team has representatives of appropriate roles, a change in the value of probability is stable. In the absence of representatives of appropriate roles, their substitution occurs as early as possible, which is why the magnitude of increment in the value of probabilities is the largest on the first step (k=1), and then gradually decreases.

Thus, both hypotheses should be considered true but some of the results would require further research into determining their causes.

7. Conclusions

1. Based on a discrete Markov chain, we constructed a model for the transitions between team roles. A parameterization of the model is performed using data obtained as the result of a survey using a self-perception questionnaire and group expert assessment. The parameterization makes it possible to adjust a Markov chain to configure the simulation of transitions between team roles in the specified projects.

2. Using the devised model, we performed modeling of transitions between team roles in the course of execution of the project. Results of the modeling were applied to verify two hypotheses concerning a change in team roles during execution of the project. It is demonstrated that under different initial conditions, it is natural for a software development project team to balance the team roles. It is expedient to continue the studies in order to examine the changes in team roles in the teams built of professionals of different age and various experience.

References

1. Ebert, C. Surviving global software development [Text] / C. Ebert, P. De Neve // IEEE Software. – 2001. – Vol. 18, Issue 2. – P. 62–69. doi: 10.1109/52.914748
2. Belbin, R. M. Management Teams. Why They Succeed or Fail [Text] / R. M. Belbin. – Oxford: Butterworth-Heinemann, 2010. – 208 p.
3. Stevens, K. T. The Effects of Roles and Personality Characteristics on Software Development Team Effectiveness [Text] / K. T. Stevens. – Virginia: Blacksburg, 1998. – 136 p.
4. Ralph, P. The dimensions of software engineering success [Text] / P. Ralph, P. Kelly // Proceedings of the 36th International Conference on Software Engineering – ICSE 2014. – 2014. doi: 10.1145/2568225.2568261
5. Syed-Abdullah, S.-L. Team achievements equality using fuzzy rule-based technique [Text] / S.-L. Syed-Abdullah, M. Omar, M. F. I. M. Idris // World Applied Sciences Journal. – 2011. – Vol. 15, Issue 3. – P. 359–363.
6. Beranek, G. Functional group roles in software engineering teams [Text] / G. Beranek, W. Zuser, T. Grechenig // ACM SIGSOFT Software Engineering Notes. – 2005. – Vol. 30, Issue 4. – P. 1. doi: 10.1145/1082983.1083108
7. Da Silva, F. Q. B. An Experimental Research on the Relationships between Preferences for Technical Activities and Behavioural Profile in Software Development [Text] / F. Q. B. da Silva, A. C. F. Cesar // 2009 XXIII Brazilian Symposium on Software Engineering. – 2009. doi: 10.1109/sbes.2009.16
8. Belbin, R. M. Method, reliability and validity, statistics and research: A comprehensive review of Belbin team roles [Text] / R. M. Belbin. – Belbin UK, 2014. – P. 1–26. – Available at: http://www.belbin.com/media/1158/belbin-uk-2014-a-comprehensive-review.pdf
9. DeChurch, L. A. The Cognitive Underpinnings of Effective Teamwork: A Meta-Analysis [Text] / L. A. DeChurch, J. R. Mesmer-Magnus // Journal of Applied Psychology. – 2010. – Vol. 95, Issue 1. – P. 32–53. doi: 10.1037/a0017328
10. Meslec, N. Are balanced groups better? Belbin roles in collaborative learning groups [Text] / N. Meslec, P. L. Curseu // Learning and Individual Differences. – 2015. – Vol. 39. – P. 81–88. doi: 10.1016/j.lindif.2015.03.020
11. Omar, M. Applying Fuzzy Technique in Software Team Formation Based on Belbin Team Role [Text] / M. Omar, B. Hasan, M. Ahmad, A. Yasin, F. Baharom, H. Mohd, N. Muhd Darus // Journal of Telecommunication, Electronic and Computer Engineering. – 2016. – Vol. 8, Issue 8. – P. 109–113.
12. Aritzeta, A. Belbin's Team Role Model: Development, Validity and Applications for Team Building [Text] / A. Aritzeta, S. Swailes, B. Senior // Journal of Management Studies. – 2007. – Vol. 44, Issue 1. – P. 96–118. doi: 10.1111/j.1467-6486.2007.00666.x
13. Van de Water, T. A balanced team generating model [Text] / T. Van de Water, H. van de Water, C. Bukman // European Journal of Operational Research. – 2007. – Vol. 180, Issue 2. – P. 885–906. doi: 10.1016/j.ejor.2006.04.017
14. Kemeni, Dzh. Konechnye tsepi Markova [Text] / Dzh. Kemeni, Dzh. Snell. – Moscow: Nauka, 1970. – 272 p.
15. Turchin, V. N. Markovskie tsepi: Osnovnye ponjatija, primery, zadachi [Text] / V. N. Turchin, E. V. Turchin. – Dnepropetrovsk: Lizunov Press, 2016. – 192 p.