A Theoretical Validation for Empirical Organizational Patterns

Satoshi Hattori and Koichiro Ochimizu

September 24, 2003
IS-RR-2003-010

School of Information Science
Japan Advanced Institute of Science and Technology
1-1 Asahidai, Tatsunokuchi,
Ishikawa, 923-1292 JAPAN
{satoshi,ochimizu}@jaist.ac.jp

A Theoretical Validation for Empirical Organizational Patterns

Satoshi Hattori and Koichiro Ochimizu
School of Information Science
Japan Advanced Institute of Science and Technology
{satoshi,ochimizu}@jaist.ac.jp

Abstract

Coplien tells us that efficient software development processes have common features. He says that such process patterns or organizational patterns are empirical facts. In this paper, a mathematical theory validates some classes of his organizational patterns. Here we use a stochastic petri-net model to analyze the efficiency of communication in software development processes.

Keywords
Organizational Pattern, Stochastic Petri-Net, Efficiency of Communication

1. Introduction

"Software pattern" is the generic name for principles or structures which are often found in software development processes. Now, software patterns are well organized and then we are able to develop software easily and efficiently by using patterns. There are patterns not only for artifacts, such as UML diagrams or source codes, but also patterns for processes themselves or development organizations. Such "process patterns" or "organizational patterns" are originated by James Coplien[1]. Now, there are many research works about process patterns and organizational patterns.

In the meantime, our research group develops works on the theme, "support distributed cooperative software development work"[2]. Our interests contain for example sharing instabilities, incremental reinforcement of consistencies and certainties, and improving the efficiency, in distributed cooperative works. A recent result is developing an automatic extraction method of deliberation threads from discussions in mailing-lists. Another recent work is to construct an extension framework for the CVS (concurrent version system). The most important future work on the theme is to develop the "Active Coordinator" for distributed cooperative works. The Active Coordinator is to obtain the total efficiency of works in an organization by controlling the work of each role in the organization.

In this paper, we discuss about organizational patterns from a theoretical perspective, for the design of the Active Coordinator. Coplien says that organizational patterns are empirical facts. However, a mathematical theory can validate some classes of organizational patterns. Here we use a stochastic petri-net model to analyze the efficiency of communication in software development processes. After showing the analysis of organizational patterns, we discuss about future works for the design of Active Coordinator.

We consider two organizational patterns, "Buffalo Mountain" and "Work Flows Inward"[1]. Target organizations of these patterns and our target, distributed cooperative development organizations, have a relationship from the following viewpoint. An organization consists of some hierarchical "roles" and a significant problem for such an organization is to make communication among the roles as much efficient as possible.

Coplien points out the following patterns which are found in organizations with efficient communication:

- (Buffalo Mountain) Communication among higher roles in the hierarchy are frequent; communication among lower roles are not frequent.
- (Work Flows Inward) Flows of messages from lower roles to higher roles (inward flows) are large; flows from higher roles to lower roles (outward flows) are small.

In this paper, communication among the roles is modeled by the following petri-net[3]. A place represents a role; a transition represents sending a message (token) from a role to another role. A role A may send a message to another role B, and also B may send a message to A, hence there are the two transitions between A and B. For such a petri-net, let a place be a node and let the two transitions between each two places be an undirected edge. Then the petri-net becomes a complete graph.

Now, let us give a firing rate to each transition of a petri-net. Such a petri-net is called a stochastic petri-net[4]. In this paper, a firing rate means the number of the messages sending by a role to another role in a unit time. Let us assume a role and a layer in a hierarchy correspond one-to-one and each role has the number of the layer that the role belongs. For an organization with n roles, the top role has the number 0 and the bottom role has the number $n - 1$. We let the Coplien's patterns be the following assumption on firing rates.

- (Buffalo Mountain) The firing rate λ_{ij} from the i-th layer role to the j-th layer role is inversely proportional to the value $i + j$. This assumption is just the same as Coplien says.
- (Work Flows Inward) Consider the i-th layer role and the j-th layer role with $i < j$, that is, the i-th layer is higher than the j-th layer. Decrease a certain value from λ_{ij} (higher to lower), and add the same value to λ_{ji} (lower to higher).

For a stochastic petri-net, we can compute the steady state probability (the staying probability) for each marking. Because we deal with only petri-nets which are similar to a complete graph and we assume the number of the tokens is only one, a marking shows the place which currently has the token. Hence, the staying probability for a marking denotes the probability that the corresponding place has the token. The performance of a petri-net, that is, the efficiency of communication is computed as follows. First, for each place, compute the sum of the firing rates of the transitions outgoing from the place. Next, for the petri-net, compute the sum of the following values for the places: the product of the staying probability for the place and the sum of the firing rates for the place obtained above. The obtained value is the total number of the sending messages in an organization described in the petri-net.

The obtained result in this paper is as follows.

- We combine the two patterns. If we decide "a certain value" suitably in Work Flows Inward pattern, then Buffalo Mountain pattern becomes effective. In such a case, the efficiency introduced above becomes large.

We conclude that Coplien tells the truth not only empirically but also mathematically.

The paper is organized as follows. First, in section 2, we will briefly explain about the two organizational patterns. Next, in section 3, we will introduce a stochastic petri-net model to describe communication in a hierarchical organization. Using this model, we will analyze the organizational patterns in section 4. In section 5, a related work will be shown. Finally, in section 6, we will explain about future works for the application of the proposed model to improve the efficiency of communication in an actual organization.

2. Patterns found in organizations with efficient communication

We briefly explain about the Coplien's "Buffalo Mountain" pattern and "Work Flows Inward" pattern [1].

2.1. Buffalo Mountain

Consider a software development organization consisting of hierarchical roles. We want to make communication in the organization as much efficient as possible. For this purpose, we should consider the frequency of communication between each two roles. Suppose that the organization consists of hierarchical n roles. Each role has the number of the hierarchical layer that the role belongs. The number of the top role (the highest role in the hierarchy) is 0; the number of a role directly under the top role is 1. The number of a bottom role is $n - 1$. Now, let us introduce the (x, y)-graph to show the frequency of communication: if the frequency of sending messages from the i-th layer role to the j-th layer role is high, then plot the grid point $(x, y) = (i, j)$ in the graph.

An organization with efficient communication have the following patterns. Let i and j be the layer numbers of arbitrary two roles in an organization with n layers.

1. (Sub-pattern 1) The frequency of communication is high, only if $i + j < n$.
2. (Sub-pattern 2) The frequency of communication is not high, if i and j are neighboring and both are near the number n.
3. (Sub-pattern 3) The frequency of communication is inversely proportional to the value $i + j$.

If communication in an organization follows the above patterns, then the shape of the plotted grids in the graph is like "Buffalo Mountain" in Colorado.

We don't know the actual shape of Buffalo Mountain. But Coplien kindly tells the analysis of the shape of the graph by considering the sub-patterns, as follows.

1. (Sub-pattern 1) Most of the grids are near the x-axis and the y-axis.
2. (Sub-pattern 2) There are few grids near the line $y = x$.
3. (Sub-pattern 3) There are also many grids near the origin $(0, 0)$.

Coplien also shows the way to approach the patterns.

- By moving responsibilities from a role to another role, reset the frequency of sending messages.

- Replace a role with another role that belongs to a different layer of hierarchy.
- Combine some roles into one new role.

2.2. Work Flows Inward

Consider again an organization consisting of hierarchical roles. Another problem is the direction of flows of messages in the organization. If flows of messages from higher roles to lower roles (outward flows) are large, then there are many redundant messages in the consequence. Hence, we should make flows from lower roles to higher roles large. In this case, the shape of the plotted grids in the graph has the following feature.

- The grids are near the x-axis rather than the y-axis.

Note that in the original description of Coplien[1], the word "message" in the above explanation should be replaced into the word "work". However, we think that the above explanation does not violate the Coplien's original idea.

3. Modeling of communication in an organization by stochastic petri-net

For a stochastic petri-net (SPN) N, each transition t has a firing rate λ. A firing rate for a transition is the inverse of the average of firing delays for the transition; firing delays for a transition follow an exponential distribution. In a SPN, the notion of "the steady state probability for a marking" is defined. This is the probability that a marking is the current marking in the process described in a SPN. (So, we call this probability by "staying probability" in other words.) From the reachability graph of a SPN N, we can obtain the state transition diagram of the Markov chain for N. By this diagram, we have the equations on the firing rates and the staying probabilities for N. Solving the equations for given firing rates, we have the staying probability for each marking of N. The above theory is called the "steady state analysis for SPN". See for example [4] for detail.

Now, recall the model of organization with hierarchical roles introduced in the previous section. First of all, in this paper, we assume the following.

- A role and a layer in a hierarchy correspond one-to-one.

We use the following SPN N_n to model communication in an organization with hierarchical n layers (i.e. the organization has n roles). The SPN N_n consists of the set P_n of places and the set T_n of transitions, where

$$P_n = \{p_1, \cdots, p_n\},$$
$$T_n = \{t_{ij}, t_{ji} | 1 \leq i \leq n, 1 \leq j \leq n, i \neq j\}.$$

(Note that the indices used in the previous section are from 0 to $n-1$, but we use the indices from 1 to n here after.) If we regard the two transitions t_{ij} and t_{ji} as one undirected edge for each pair of i, j, then the SPN N_n can be regarded as the complete graph K_n with the node set P_n. Figure 1 shows the SPN N_4, which models communication in an organization with hierarchical 4 roles.

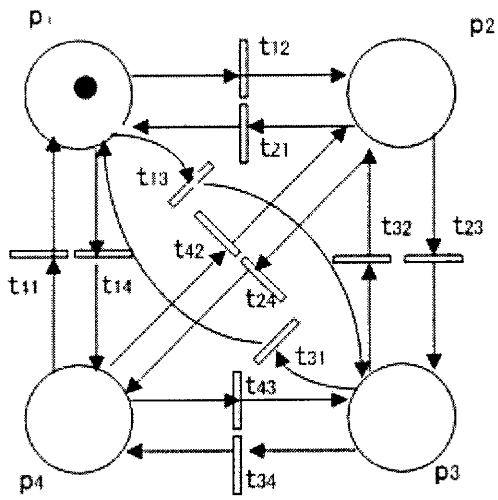

Figure 1. SPN N_4

Let us denote a marking (distribution of the tokens) in N_n by $[m_1, \cdots, m_n]$. In this paper, we assume the following.

- (The assumption for the initial marking)
  The initial marking for SPN N_n is $M_1 = [1, 0, \cdots, 0]$.

By this assumption, we have the set of markings for N_n as follows: $\{M_1 = [1,0,0,\cdots,0], M_2 = [0,1,0,\cdots,0], \cdots, M_n = [0,0,0,\cdots,1]$. In other words, we have the following proposition.

- (The proposition for the markings)
  Under the assumption for the initial marking, for each n, every marking in N_n shows the place which has the only one token.

Let λ_{ij} be the firing rate for a transition $t_{ij} (1 \leq i \leq n, 1 \leq j \leq n, i \neq j)$, and let π_k be the staying probability for a marking $M_k (1 \leq k \leq n)$. Consider the SPN N_4 in Figure 1. Using the proposition for the markings, we obtain the reachability graph for N_4 by replacing each place $p_k (1 \leq k \leq 4)$ in N_4 into the corresponding marking M_k. Let us regard the obtained reachability graph as the state transition diagram of the Markov chain. Then, we have the following five equations on the firing rates and the staying probabilities. Each of the first four equations shows the following for each marking: for a unit time, the number of the incoming firings and the number of the outgoing firings are equal. The last equation shows the sum of the staying probabilities of all markings equals to 1. Solving the equations, we have the value of the staying probability for each marking.

$$
\begin{aligned}
&(\lambda_{12} + \lambda_{13} + \lambda_{14})\pi_1 = \lambda_{21}\pi_2 + \lambda_{31}\pi_3 + \lambda_{41}\pi_4, \\
&(\lambda_{21} + \lambda_{23} + \lambda_{24})\pi_2 = \lambda_{12}\pi_1 + \lambda_{32}\pi_3 + \lambda_{42}\pi_4, \\
&(\lambda_{31} + \lambda_{32} + \lambda_{34})\pi_3 = \lambda_{13}\pi_1 + \lambda_{23}\pi_2 + \lambda_{43}\pi_4, \\
&(\lambda_{41} + \lambda_{42} + \lambda_{43})\pi_4 = \lambda_{14}\pi_1 + \lambda_{24}\pi_2 + \lambda_{34}\pi_3, \\
&\pi_1 + \pi_2 + \pi_3 + \pi_4 = 1.
\end{aligned}
$$

For the SPN N_n with arbitrary n, we can have the similar equations. Here we define a benchmark to estimate the efficiency of communication in an organization.

- The total number of the firings F_n:

$$F_n = \sum_{1 \le k \le n} \pi_k \sum_{1 \le i < k, k < i \le n} \lambda_{ki}.$$

  F_n shows the number of the all firings in the process described in N_n. F_4 for N_4 is the sum of the left-hand sides of the first four equations in the above.

- The average of firing rate A_n:

$$A_n = \sum_{1 \le k \le n} \frac{1}{n} \sum_{1 \le i < k, k < i \le n} \lambda_{ki}.$$

  For each place, we can compute the sum of the firing rates of the transitions outgoing from the place. A_n is the average of the above sum of all places.

- Firing efficiency E_n:

$$E_n = \frac{F_n}{A_n}.$$

  This is the benchmark of the efficiency of the firings, the efficiency of communication in an organization, in other words. Note that "the performance of the petri-net" introduced in the Introduction of this paper is F_n. But we actually use E_n here after to estimate the performance of the petri-net N_n.

The following is another explanation for F_n and A_n. A_n is the syntactical average of the sum of the firing rates; it is independent of the execution of the SPN N_n. F_n is the semantic average of the sum of the firing rates; it is determined by executing N_n.

For the SPN N_n with any n, we have the following fact.

- Let $\lambda_{ij} = \lambda_{ji}$ for all pairs of indices i and j. By solving the equations on the firing rates and the staying probabilities, we have $\pi_1 = \pi_2 = \cdots = \pi_n = \frac{1}{n}$. Therefore, $F_n = A_n$ and $E_n = 1$ hold true.

This fact tells us that the following situation is ideal ($E_n = 1$): if a role r_1 send a message to another role r_2, r_2 should send a reply message to r_1 ($\lambda_{ij} = \lambda_{ji}$).

Suppose that we use only Buffalo Mountain pattern. Then the above assumption $\lambda_{ij} = \lambda_{ji}$ holds true for all pairs of indices i and j. Hence, we have "only the ideal" situation, $E_n = 1$. In the Introduction of this paper, we said that "if we decide a certain value suitably in Work Flows Inward pattern, then Buffalo Mountain pattern becomes effective". "Effective" means that $E_n > 1$ holds true for some case. We show this in the next section.

4. Analysis of organizational patterns using stochastic petri-net

In this section, we discuss about an analysis of the SPN N_5 with firing rates following the two organizational patterns. Here we use the same notations as the previous section.

About Buffalo Mountain pattern, Coplien enumerated the three sub-patterns. But we deal with only the last sub-pattern, "inversely proportional" pattern, because we think this pattern is the most essential among the three sub-patterns. Here we introduce four patterns including Buffalo Mountain pattern.

1. Buffalo Mountain pattern, that is, the firing rate λ_{ij} from the i-th layer role to the j-th layer role is inversely proportional to the value $i + j$.

2. The pattern that the firing rate λ_{ij} is inversely proportional to the value $i \times j$. This is a pattern that higher roles communicate each other more frequently and lower roles do more infrequently than Buffalo Mountain pattern.

3. The pattern that all λ_{ij}'s are equal for every indices i and j.

4. The pattern that λ_{ij} is directly proportional to $i + j$.

The below matrix Λ shows an example of the firing rates for the pattern 1 (Buffalo Mountain pattern). The value of the i-th row and j-th column denotes the firing rate λ_{ij}. We let $\lambda_{ij} = 0$ for the case $i = j$ which is not defined in the SPN N_5.

$$\Lambda = \begin{pmatrix} 0 & \frac{1}{1+2} & \frac{1}{1+3} & \frac{1}{1+4} & \frac{1}{1+5} \\ \frac{1}{2+1} & 0 & \frac{1}{2+3} & \frac{1}{2+4} & \frac{1}{2+5} \\ \frac{1}{3+1} & \frac{1}{3+2} & 0 & \frac{1}{3+4} & \frac{1}{3+5} \\ \frac{1}{4+1} & \frac{1}{4+2} & \frac{1}{4+3} & 0 & \frac{1}{4+5} \\ \frac{1}{5+1} & \frac{1}{5+2} & \frac{1}{5+3} & \frac{1}{5+4} & 0 \end{pmatrix} = \begin{pmatrix} 0 & \frac{1}{3} & \frac{1}{4} & \frac{1}{5} & \frac{1}{6} \\ \frac{1}{3} & 0 & \frac{1}{5} & \frac{1}{6} & \frac{1}{7} \\ \frac{1}{4} & \frac{1}{5} & 0 & \frac{1}{7} & \frac{1}{8} \\ \frac{1}{5} & \frac{1}{6} & \frac{1}{7} & 0 & \frac{1}{9} \\ \frac{1}{6} & \frac{1}{7} & \frac{1}{8} & \frac{1}{9} & 0 \end{pmatrix}$$

Next, we adopt Work Flows Inward pattern for the firing rates. Note that the following is one candidate for adoption of Work Flows Inward pattern. We think that there are many other candidates for adoption.

- Consider the i-th layer role and the j-th layer role with $i < j$, that is, the i-th layer is higher than the j-th layer. First compute the sum of λ_{ij} and λ_{ji}. Then decrease the l (parameter) times of the sum from λ_{ij} (higher to lower), and add the l times of the sum to λ_{ji} (lower to higher).

Applying the above adoption, we have the next matrix of the firing rates Λ' for the previous Λ for the pattern 1.

$$\Lambda' = \begin{pmatrix} 0 & \frac{1}{3} - \frac{2}{3}l & \frac{1}{4} - \frac{2}{4}l & \frac{1}{5} - \frac{2}{5}l & \frac{1}{6} - \frac{2}{6}l \\ \frac{1}{3} + \frac{2}{3}l & 0 & \frac{1}{5} - \frac{2}{5}l & \frac{1}{6} - \frac{2}{6}l & \frac{1}{7} - \frac{2}{7}l \\ \frac{1}{4} + \frac{2}{4}l & \frac{1}{5} + \frac{2}{5}l & 0 & \frac{1}{7} - \frac{2}{7}l & \frac{1}{8} - \frac{2}{8}l \\ \frac{1}{5} + \frac{2}{5}l & \frac{1}{6} + \frac{2}{6}l & \frac{1}{7} + \frac{2}{7}l & 0 & \frac{1}{9} - \frac{2}{9}l \\ \frac{1}{6} + \frac{2}{6}l & \frac{1}{7} + \frac{2}{7}l & \frac{1}{8} + \frac{2}{8}l & \frac{1}{9} + \frac{2}{9}l & 0 \end{pmatrix}$$

If the value of l is positive, then messages flow inward; if the value of l is negative, then messages flow outward. By the way, in the section 2, the (x, y)-graph is introduced to show the frequency of communication. The (x, y)-graph there and the matrix here correspond as follows: rightward on x-axis in the graph and downward on a column in the matrix correspond, and upward on y-axis in the graph and rightward on a row in the matrix correspond.

For each of the four patterns, we have the the equations on the firing rates and the staying probabilities for the SPN N_5, similar to the equations for N_4. Solving the equations, we have the firing efficiency E_5 with the parameter l. Figure 2,3,4 and 5 show the graphs of $E_5(l)$ for the patterns 1,2,3 and 4 respectively.

For the patterns 1 and 2, $E_5(l) > 1$ holds for some value of l. For the pattern 1, the maximal value of $E_5(l)$ is 1.048 when $l = 0.138$. For the pattern 2, the maximal value of $E_5(l)$ is 1.211 when $l = 0.282$. Therefore, the pattern 2 is better than the pattern 1 (Buffalo Mountain pattern) from the viewpoint of the maximal efficiency.

Note that also for the pattern 4, $E_5(l) > 1$ holds for some value of l. However, in this case, the value of l is negative. Hence, the messages should flow outward. The pattern 4 is the case that the lowest role is actually the highest role.

Let us discuss about another estimation for the patterns 1 and 2. Here we consider the distribution of the staying probabilities. By "the proposition for the markings" in the preceding section, every marking in the SPN N_5 shows the place which has the only one token. Hence the staying probability for a marking is the probability that the corresponding place has the token. In other words, the staying probability for a marking shows the coverage that the corresponding place (role) does communication in the whole organization. Therefore, the distribution of the staying probabilities shows the load balance among the roles.

Let π_1 and π_5 be the staying probabilities of the markings corresponding to the highest role and the lowest role in the SPN N_5, respectively. For the value of the parameter l that makes the efficiency $E_5(l)$ maximal, the value of π_1 and π_5 are as follows: $\pi_1 = 0.317$ and $\pi_5 = 0.129$ for the pattern 1 and $\pi_1 = 0.498$ and $\pi_5 = 0.088$ for the pattern 2. The pattern 2 has the larger maximal efficiency than the pattern 1, but the load of the highest role is also greater than that of the pattern 1. The maximal value of the efficiency and the load of higher roles seem to be trade-off.

By the above argument, we can conclude that if we give the suitable value of "Inwardness" for Work Flows Inward pattern, then "inversely proportional pattern" becomes effective. If we think the load of the highest role is too high for the pattern 2, we can conclude that the pattern 1(Buffalo Mountain) is better than the pattern 2.

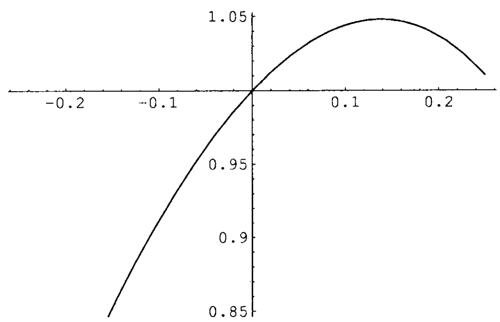

Figure 2. The graph of the firing efficiency for pattern 1

5. Related Works

The paper [5] has already discussed about an application of stochastic petri-net to support of software development. We use the theory of steady state analysis of SPN, but in [5]

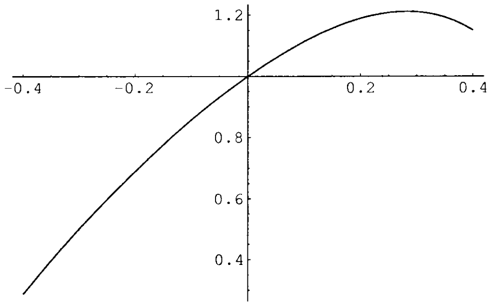

Figure 3. The graph of the firing efficiency for pattern 2

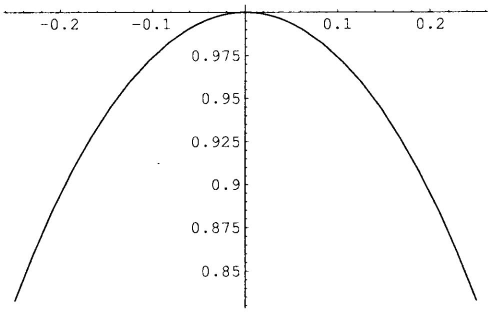

Figure 4. The graph of the firing efficiency for pattern 3

Generalized SPN (GSPN) is used for simulation of software development process.

The below is a brief explanation of simulation proposed in [5]. A GSPN is given for each "activity" of a software development process. The simulator gives the firing rates and the value of the "workload" to the GSPN for each activity. A firing of a transition means an advance of the development process. The token has some attributes such that the "size of the product" and the "consumed workload". They increase when a transition fires. When the consumed workload becomes the given value of the workload, the activity finishes. At the time, one can know the current size of the product and the time to spend for the activity. The latter is obtained because the simulation uses the GSPN.

6. Future works: to improve the efficiency of communication in an actual organization

In this section, we discuss about future works for improvement of the efficiency of communication in an actual organization, by applying the discussion in the previous section.

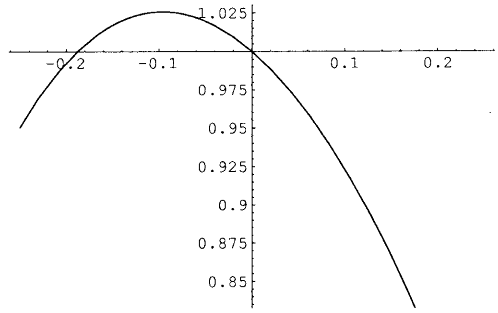

Figure 5. The graph of the firing efficiency for pattern 4

6.1. About the modeling of an organization by petri-net

In this paper, we assume that there is only one role in a hierarchical layer and we use the SPN N_n with very simple form. However, in an actual organization, there can be more than two roles in a hierarchical layer. Moreover, for sending a message, synchronizations of receiving some messages may be required. The SPN N_n cannot be used for such a case.

We also have "the assumption for the initial marking", i.e. we assumed that a petri-net has only one token. This assumption means that the number of the talking subjects in an organization is always only one and the number of the persons who have the right to send a message is also only one. However, actually, subjects arise and disappear dynamically and people discuss in parallel. There may be a case that when a role asks to another role, a third role gives an answer.

The following problem arises when we model communication for such an actual cases. For an arbitrary petri-net, "the proposition for the markings" does not hold true. Even for the case that we use the SPN N_n the proposition does not hold if we do not have the assumption for the initial marking. Our first work to do is to conquer such cases.

6.2. About the way to approach the patterns

Next, we discuss about problems when we use the SPN N_n for the modeling of communication. We can know the current firing rates by checking the mail server of an organization. By solving the equations on the firing rates and the staying probabilities, we have the efficiency of communication E_n and the distribution of the staying probabilities, as shown in the previous section. To make the efficiency E_n more larger, we should do as follows.

1. First, by changing the values of some firing rates, make

the distribution of the firing rates as an "inversely proportional" pattern, at least a "monotonically decreasing" pattern. However, there are many candidates for the following.

- Ways to make the distribution of the firing rates monotonically decreasing.
- Constraints on the distribution. Because we should not change the current distribution drastically, we need some constraints.

The average of firing rate A_n is not changed when we apply the way in the previous section to change the inwardness (parameter l). We think that preserving the value of A_n is a candidate for a constraint on the distribution. Under this constraint, we introduce a candidate for the way to make the distribution monotonically decreasing. Suppose that there is a row $(10, 9, 8, 11, \cdots)$ in the matrix of the current firing rates such as Λ in the preceding section. The distribution in the row is not monotonically decreasing. So, decrease the value 3 from the firing rate 11 and add 1 for each of the succeeding rates, that is, $10, 9$ and 8. We have the new distribution $(11, 10, 9, 8, \cdots)$ and this is monotonically decreasing. Applying this procedure for each row and for each column of the matrix, we have the new matrix of the firing rates. The above is only one candidate for making the distribution monotonically decreasing. To develop a good method is a future work.

2. Next, introduce the parameter l of the inwardness of the flow. Then, compute the value of l to make the efficiency E_n maximal.

It is more difficult to improve the distribution of the staying probabilities, in other words, the load balance among the roles. Suppose that one decides an ideal distribution of the staying probabilities after computing the current distribution. The problem here is to obtain the new matrix of the firing rates which satisfies the equations on the firing rates and the given staying probabilities. Also in this case, we should introduce some constraints on the distribution of the firing rates, because we should not change the current distribution drastically. To give a solution for the above problem is also a future work.

The above discussion is to approach the patterns "mathematically". In the section 2, we introduced the way to approach the patterns that Coplien proposes. For example, consider the way that "by moving responsibilities from a role to another role, reset the frequencies of sending messages". The new problem is to obtain the way to suitably move responsibilities to get the desirable firing rates. We must solve this problem when we implement the Active Coordinator explained in the Introduction of this paper.

6.3. About the mathematical processing

By the way, to solve the equations appearing in this paper, we use Mathematica[6] (Ver3.0) on a personal work station (Sun Blade 150, Ultra SPARC IIi 550MHz, Memory 512MB, Solaris 9). To solve equations with parameters requires great cost. In spite of this problem, we do not use any device to solve equations efficiently. In the consequence, the number of the hierarchical layers is limited to 5. In [1], Coplien presents an example of a hierarchical organization with 16 layers for Buffalo Mountain pattern. Devices for computation is necessary to solve equations for such a large organization in practical time. Another way to conquer the situation is to implement a specific tool for us.

References

[1] Coplien, J.: "A Generative Development-Process Pattern Language", Pattern Languages of Program Design, pp.183–237, Addison Wesley(1995). http://www.bell-labs.com/user/cope/Patterns/Process/

[2] Ochimizu, K., Murakoshi, H., Fujieda, K. and Fujita, M.: "Sharing Instability of a Distributed Cooperative Work", Proceedings of the International Symposium on Principles of Software Evolution(ISPSE 2000), pp.33–42(2000).

[3] Murata, T.:"Petri Nets: Properties, Analysis and Applications", Proceedings of the IEEE, Vol.77, No.4, pp.541–580(1989).

[4] Molloy, M.: "Performance Analysis Using Stochastic Petri Nets", IEEE Transactions on Computers, Vol.C-31, No.9, pp.913–917(1982).

[5] Kusumoto, S., Mizuno, O., Kikuno, T., Hirayama, Y., Takagi, Y. and Sakamoto, K.: "A New Software Project Simulator Based on Generalized Stochastic Petri-net", Proceedings of the 19th International Conference on Software Engineering(ICSE'97), pp.293–302, ACM Press(1997).

[6] Wolfram, S.: "The Mathematica Book, Fourth Edition", Wolfram Media/Cambridge University Press(1999).