

ALTERNATIVE ROUTS OF GAMES WITH RIGID SCHEDULE

E.V. Larkin¹, A.N. Privalov²

¹ Tula State University, Tula, Russian Federation

² Tula State Lev Tolstoy Pedagogical University, Tula, Russian Federation

E-mail: privalov.61@mail.ru

Corporative-competitive system, which is inside of corporations, can be determined as a “game”, step-by-step performing a certain type of activity. The system operates in real physical time, and the result of operation is the distance, which is divided into stages. The stages are passed by the team participants due to rigid schedule, which may be occasionally selected from the set of possible schedules. The abstraction “M-parallel semi-Markov process” is used for description of a system under consideration. In semi-Markov process degenerate distribution is used for description of time intervals between relay points. For analysis of relay-race evolution, recurrent method which takes into account rigidity of schedule and stochastic character of route selection is used. In accordance with the concept of distributed forfeit and proposed recurrent procedure, the method of calculation of summing forfeit, which one of competing teams receives from other teams, is proposed.

Keywords: relay-race; semi-Markov process; degenerate distribution; route; evolution; distributed forfeit; recurrent procedure.

Introduction

Relay-races, as the basic conception of corporative-concurrent system description, may be applied to modeling of such fields of human activity, as industry, economics, politics, defense, sport, etc [1–4]. Due to conception announced teams, participating in relay-race, should to pass the distance, which is divided onto stages by relay-points, and team participants should to pass the stage in real physical time. Common case of random time relay-race simulation was considered in [2, 5], where for description of teams behavior such abstraction, as semi-Markov process was used. Semi-Markov process is quite universal mathematical apparatus, and when instead of random time emerges rigid schedule, it can be used too. Rigidity of stage passing time permit substantially simplify model of the system and calculation of forfeit, but also leads to substantial restrictions of results obtained.

On practice teams, participated the relay race, may vary their schedules, and for an external observer such variations are the stochastic ones. This permits to consider different combinations of schedules and to improve results obtained. Approaches to modeling of relay-races with rigid schedules and alternative routes are currently known insufficiently, that explains necessity and relevance of the investigations in this domain.

1. Relay-race as M-parallel semi-Markov process

The graph, which shows the alternative routes rigid schedule relay-race structure, is shown on the fig. 1.

Following assumptions, when modeling this kind of races, are made bellow [5]:

- in relay-race participate M -teams, every of which pass its distance in real physical time;
- distance of every team is divided onto stages, every of which is overcame by one participant of a team, and first participants of all teams start their stages at once;
- every participant may choose a route, for passing the stage, for an external observer the route selection is a random event;
- passing the stage route by participant lasts rigid time, which is individual for the team, the stage and the route;
- after completion of a current stage on the selected route next participant of the team selects route and starts the passing next stage without a lag;
- forfeit, which is imposed on the teams is defined as the distributed payment, value of which depends on the time and difference of stages, which currently teams pass.

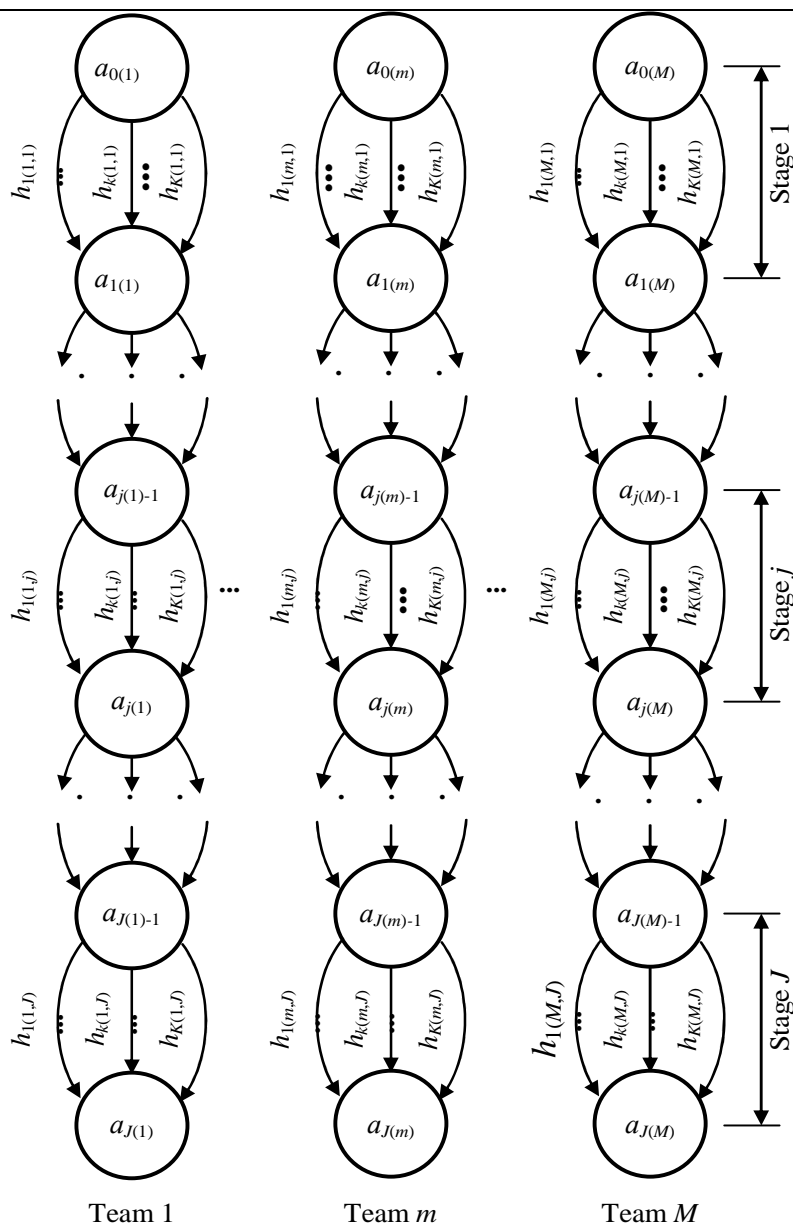


Fig. 1. Structure of alternative routes rigid schedule relay-race

The model of relay-race with M teams may be performed as M -parallel semi-Markov process [6–8]:

$$\mu = \{\mu_1, \dots, \mu_m, \dots, \mu_M(t)\}, \quad (1)$$

$$\mu_m = \{A_m, h_m(t)\},$$

where t is the physical time; μ_m is the ordinary semi-Markov process, A_m is the set of states; $h_m(t)$ is the semi-Markov matrix, which describes an activity of the m -th team;

$$A_m = \{a_{0(m)}, a_{1(m)}, \dots, a_{j(m)}, \dots, a_{J(m)}\}; \quad (2)$$

$$\mathbf{h}_m(t) = [h_{j(m),l(m)}(t)]; \quad (3)$$

$$\mathbf{h}_m(t) = [h_{j(m),l(m)}(t)], \quad 1 \leq j(m), l(m) \leq J; \quad (4)$$

$$h_{j(m),l(m)}(t) = \begin{cases} \sum_{k(m,j)=1}^{K(m,j)} h_{k(m,j)}(t), & \text{when } l(m) = j(m) + 1, 0 \leq j(m) \leq J(m) - 1; \\ 0 & \text{in all other cases.} \end{cases} \quad (5)$$

Due to rigid schedule and quasi-stochastic principle of route selection [9]

$$h_{k(m,j)}(t) = p_{k(m,j)} f_{k(m,j)}(t) = p_{k(m,j)} \delta[t - T_{k(m,j)}], \quad (6)$$

where $p_{k(m,j)}$ is the probability of route selection; $f_{k(m,j)}(t)$ is the pure time density of residence the process at the state $a_{j(m)}$ with further switch to the state $a_{j(m)+1}$ on the k -th route; $\delta(t - T_{k(m,j)})$ is the Dirac δ -function; $T_{k(m,j)}$ is the rigid time of passing the j -th stage by participant of team m with $k(m, j)$ -th route.

Comparison of classic competition [2] and competition with rigid schedule is shown on the fig. 2;

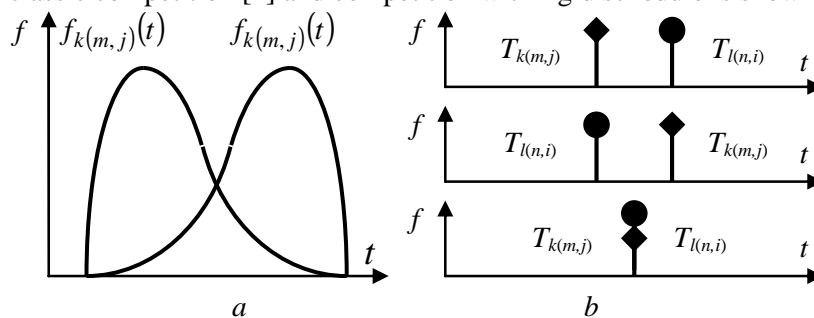


Fig. 2. Classic competition (a) and competition with rigid schedules (b)

When classic competition, times of semi-Markov process residence in states $a_{j(m)}$ and $a_{i(n)}$ are equal to $f_{k(m,j)}(t)$ and $f_{l(n,i)}(t)$, correspondingly. When rigid schedule times are equal to $\delta[t - T_{k(m,j)}]$ and $\delta[t - T_{l(n,i)}]$. Formulae, which describe weighted time density and probabilities of winning the j -th stage of race by m -th team, if all participating teams start their stages simultaneously, are as follows:

$$\psi_{k(m,j)}^w(t) = f_{k(m,j)}(t) \prod_{\substack{n=1 \\ n \neq m}}^M [1 - F_{l(n,i)}(t)], \quad (7)$$

$$\pi_{k(m,j)}^w = \int_0^{\infty} \psi_{k(m,j)}^w(t) dt; \quad (8)$$

$$\psi_{k(m,j)}^w(t) = \begin{cases} \delta(t - T_{k(m,j)}), & \text{when } T_{k(m,j)} = \min\{T_{l(1,i)}, \dots, T_{k(m,j)}, \dots, T_{l(M,i)}\}; \\ \text{nonsense,} & \text{otherwise,} \end{cases} \quad (9)$$

$$\pi_{k(m,j)}^w = \begin{cases} 1, & \text{when } T_{k(m,j)} = \min\{T_{l(1,i)}, \dots, T_{k(m,j)}, \dots, T_{l(M,i)}\}, \\ 0, & \text{otherwise,} \end{cases} \quad (10)$$

where $F_{\dots}(t) = \int_0^t f_{\dots}(\theta) d\theta$ is the distribution function; θ is the auxiliary argument.

Pure time density of winning the j -th stage of race by m -th team is as follows:

$$\phi_{k(m,j)}^w(t) = \frac{\psi_{k(m,j)}^w(t)}{\pi_{k(m,j)}^w}. \quad (11)$$

When paired competition, formulae, which describe the time density of waiting by m -th, winner, team until n -th, loser, team, finishes the stage, are as follows:

$$\varphi_{k(m,j) \rightarrow l(n,i)}(t) = \frac{\eta(t) \int_0^{\infty} f_{k(m,j)}(\theta) \cdot f_{l(n,i)}(t+\theta) d\theta}{\int_0^{\infty} F_{k(m,j)}(t) dF_{l(n,i)}(t)}; \quad (12)$$

$$\varphi_{k(m,j) \rightarrow l(n,i)}(t) = \begin{cases} \delta[t - T_{l(n,i)} + T_{k(m,j)}], & \text{when } T_{k(m,j)} \leq T_{l(n,i)}; \\ \text{nonsense,} & \text{otherwise,} \end{cases} \quad (13)$$

where $\eta(t)$ is the Heaviside function;

$$\eta(t) = \begin{cases} 0, & \text{when } t \leq 0; \\ 1, & \text{otherwise.} \end{cases}$$

It is necessary to admit, that in the case under consideration, unlike the case considered at [2], the draw effect emerges. It is caused with the infinitesimal probability of two or more teams stage passing times coincidence when time intervals are a random ones, and quite real rigid schedule case, when some time intervals are quite the same (the case is shown on the fig. 2, *b*, bottom line).

Due to the fact, that for all teams, participated in a race, number of the stage j at every relay point may to increment only, for external viewer sequence of switches during relays in the system as a whole has the nature of evolution [10–13], which develops from functional state, being defined with vector [14] $(a_{0(1)}, \dots, a_{0(m)}, \dots, a_{0(M)})$ till functional state $(a_{J(1)}, \dots, a_{J(m)}, \dots, a_{J(M)})$. Trajectory of evolution depends of routes, which every team select for passing of proper stages and schedules which develop from routes selected. Owing to random character of routes selection trajectory of evolution is the random one. Common number of routes, on which m -th team as a whole may overcome the distance is equal to

$$\tilde{K}(m) = \prod_{j=1}^J K(m, j), \quad (14)$$

where $K(m, j)$ is the common number of routs of m -th team j -th stage.

Common number of different variants of rigid relay-races is as follows:

$$\tilde{K} = \prod_{m=1}^M \tilde{K}(m). \quad (15)$$

2. Recursive procedure of relay-race evolution analysis

Let us select from all possible routes on which m -th team may overcome the distance the $\tilde{k}(m)$ -th variant, in which routes $k(m, 1), \dots, k(m, j), \dots, k(m, J)$ are selected (note, that in common case $k(m, 1) \neq \dots \neq k(m, j) \neq \dots \neq k(m, J)$, since these are different functions of m and j). So $\tilde{1}(m)$ -th and $\tilde{K}(m)$ -th variants correspond to combinations $1(m, 1), \dots, 1(m, j), \dots, 1(m, J)$ and $K(m, 1), \dots, K(m, j), \dots, K(m, J)$ respectively. Such a selection should be executed on routes of all other teams. For selected routes the recursive procedure of relay-race evolution may be worked out. For this purpose auxiliary time θ should be introduced. Recursive procedure of evolution analysis is as follows.

Initial functional state of the semi-Markov process (1) as a whole is $(a_{0(1)}, \dots, a_{0(m)}, \dots, a_{0(M)})$, where elements of vector are numbers of relay-race starting points. All teams start race simultaneously, and into auxiliary time θ next substitutions should be made:

$$\theta_0(1) \Leftarrow T_{k(1,1)}, \dots, \theta_0(m) \Leftarrow T_{k(m,1)}, \dots, \theta_0(M) \Leftarrow T_{k(M,1)}, \quad (16)$$

where arrow \Leftarrow indicates the direction of substitution; index means the quantity of previous switches.

Rigid time intervals $\theta_0(1), \dots, \theta_0(m), \dots, \theta_0(M)$ compete between them, and result of competition is winning the stage by those teams, which have minimal value of θ . This values fulfill the subset Θ_0^* :

$$\Theta_0^* = \{\theta_0^*(u), \dots, \theta_0^*(v), \dots, \theta_0^*(w)\}; \quad (17)$$

$$\{\theta_0^*(u), \dots, \theta_0^*(v), \dots, \theta_0^*(w)\} = \min_{\theta} \{\theta_0(1), \dots, \theta_0(m), \dots, \theta_0(M)\},$$

where u, v, w are auxiliary indices for nomination the number of team.

It is obviously, that

$$\theta_0^* \Leftarrow \theta_0^*(u) = \dots = \theta_0^*(v) = \dots = \theta_0^*(w), \quad (18)$$

i. e. teams with numbers

$$N_0^* = \{\arg[\theta_0^*(u)], \dots, \arg[\theta_0^*(v)], \dots, \arg[\theta_0^*(w)]\} \quad (19)$$

win the competition among other teams, but draw the competition among themselves. Quantity of switches $\bar{s}(1)$ on the first step of recursion is equal to cardinal number of subset Θ_0^* :

$$\bar{s}(1) = |\Theta_0^*|. \quad (20)$$

Due to the switches, next substitutions should be maid to prepare next step of recursion:

index of $\theta_0(m)$, $1 \leq m \leq M$, should be replaced with index of $s(1) \Leftarrow 0 + \bar{s}(1)$;

time intervals, which will compete further should be replaced as follows

$$\theta_{s(1)}(m) \Leftarrow \begin{cases} \theta_0(m) - \theta_0^*, & \text{when } m \notin N_0^*; \\ T_{k(m,2)}, & \text{when } m \in N_0^*; \end{cases} \quad (21)$$

indices $\iota(m)$ of elements of vector $(a_{0(1)}, \dots, a_{0(m)}, \dots, a_{0(M)})$ should be replaced as follows:

$$\iota(m) \Leftarrow \begin{cases} 0(m), & \text{when } m \notin N_0^*; \\ 1(m), & \text{when } m \in N_0^*. \end{cases} \quad (22)$$

In such a way on the second step or recursion rigid time intervals $\theta_{s(1)}(1), \dots, \theta_{s(1)}(m), \dots, \theta_{s(1)}(M)$ will compete between them.

Let us assume that on the r -th step of the recursion vector of functional state of semi-Markov process is $(a_{j(1)-1}, \dots, a_{j(m)-1}, \dots, a_{j(M)-1})$, and time intervals $\theta_{s(r-1)}(1), \dots, \theta_{s(r-1)}(m), \dots, \theta_{s(r-1)}(M)$ compete between them. Result of competition is winning the stage by those teams, which have minimal value of θ . This values fulfill the subset $\Theta_{s(r-1)}^*$:

$$\Theta_{s(r-1)}^* = \{\theta_{s(r-1)}^*(u), \dots, \theta_{s(r-1)}^*(v), \dots, \theta_{s(r-1)}^*(w)\}; \quad (23)$$

$$\{\theta_{s(r-1)}^*(u), \dots, \theta_{s(r-1)}^*(v), \dots, \theta_{s(r-1)}^*(w)\} = \min_{\theta} \{\theta_{s(r-1)}(1), \dots, \theta_{s(r-1)}(m), \dots, \theta_{s(r-1)}(M)\}.$$

It is obviously, that

$$\theta_{s(r-1)}^* \Leftarrow \theta_{s(r-1)}^*(u) = \dots = \theta_{s(r-1)}^*(v) = \dots = \theta_{s(r-1)}^*(w), \quad (24)$$

i.e. teams with numbers

$$N_{s(r-1)}^* = \{\arg[\theta_{s(r-1)}^*(u)], \dots, \arg[\theta_{s(r-1)}^*(v)], \dots, \arg[\theta_{s(r-1)}^*(w)]\} \quad (25)$$

win the competition. Quantity of switches on the r -th step of recursion is equal to cardinal number of subset $\Theta_{s(r-1)}^*$:

$$\bar{s}(r) = |\Theta_{s(r-1)}^*|. \quad (26)$$

Due to the switches, following substitutions should be maid to prepare next step of recursion:

index of $\theta_{s(r-1)}(m)$ $1 \leq m \leq M$, should be replaced with index $\bar{s}(r) \leftarrow s(r-1) + \bar{s}(r)$;

time intervals, which will compete further should be replaced as follows –

$$\theta_{s(r)}(m) \leftarrow \begin{cases} \theta_{s(r-1)}(m) - \theta_{s(r-1)}^*, & \text{when } m \notin N_{s(r-1)}^*; \\ T_{k(m,j+1)}, & \text{when } m \in N_{s(r-1)}^*; \end{cases} \quad (27)$$

components of vector $(a_{j(1)-1}, \dots, a_{j(m)-1}, \dots, a_{j(M)-1})$ should be replaced as follows –

$$a_{j(m)-1} \leftarrow \begin{cases} a_{j(m)-1}, & \text{when } m \notin N_{s(r-1)}^*; \\ a_{j(m)}, & \text{when } m \in N_{s(r-1)}^*. \end{cases} \quad (28)$$

In such a way on the $(r+1)$ -th step or recursion rigid time intervals $\theta_{s(r)}(1), \dots, \theta_{s(r)}(m), \dots, \theta_{s(r)}(M)$ compete between them.

Let assume that on the last but one step of recursion after $MJ-3$ switches vector became $(a_{J(1)}, \dots, a_{J(u)-1}, \dots, a_{J(v)-1}, \dots, a_{J(m)}, \dots, a_{J(w)-1}, \dots, a_{J(M)})$, and rigid time intervals $\theta_{MJ-3}(k), \theta_{MJ-3}(l), \theta_{MJ-3}(n)$ compete between them only. Let us also assume, that minimal values of θ , which fulfill the subset Θ_{JM-3}^* , are

$$\Theta_{JM-3}^* = \{\theta_{JM-3}^*(u), \theta_{JM-3}^*(w)\}; \quad (29)$$

$$\{\theta_{JM-3}^*(u), \dots, \theta_{JM-3}^*(w)\} \subseteq \min_{\theta} \{\theta_{MJ-3}(u), \theta_{MJ-3}(v), \theta_{MJ-3}(w)\}.$$

$$\theta_{MJ-3}^* \leftarrow \theta_{MJ-3}^*(u) = \theta_{MJ-3}^*(w), \quad (30)$$

i. e. teams with numbers u and w win the competition.

Due to (29) and (30) on the last but one step two switches occur, namely u -th and w -th teams reach the final point of race and after that functional state became $(a_{J(1)}, \dots, a_{J(u)}, \dots, a_{J(v)-1}, \dots, a_{J(m)}, \dots, a_{J(w)}, \dots, a_{J(M)})$, i.e. w -th team on the last step should finish the distance at the time

$$\theta_{MJ-3+2}(l) = \theta_{MJ-3}(l) - \theta_{MJ-3}^*, \quad (31)$$

and relay race in the state $(a_{J(1)}, \dots, a_{J(m)}, \dots, a_{J(M)})$ will over.

3. Schedule effectiveness evaluation recursive procedure

Quite natural for evaluation of rigid schedule effectiveness is the model, in which all possible pairs of teams, f.e. m -th and n -th, and forfeit $\sigma_{i(m),j(n)}(t)$, distributed at the time, are considered. Distributed forfeit, in turn, is defined as payment, which the n -th team pays to the m -th team, namely

$$\sigma_{i(m),j(n)}(t) \leftarrow \begin{cases} > 0, & \text{when } i(m) > j(n); \\ = 0, & \text{when } i(m) = j(n); \\ < 0, & \text{when } i(m) < j(n), \end{cases} \quad (32)$$

where $0 \leq i(m), j(n) \leq J$.

For evaluation of common forfeit, which the m -th team receives from the n -th team, one can to use the recursive procedure, described above. At initial two-elements functional state $(a_{0(m)}, a_{0(n)})$ substitutions

$$\theta_0(m) \leftarrow T_{k(m,1)}, \quad \theta_0(n) \leftarrow T_{k(n,1)} \quad (33)$$

should be done.

In this simplified two-parallel semi-Markov process only time intervals $\theta_0(m)$, $\theta_0(n)$ compete between them. Possible result of competition may be the next:

a) wins the team m , if $\theta_0(m) < \theta_0(n)$, then $\Theta_0^* = \{\theta_0^*(m)\} = \min\{\theta_0(m), \theta_0(n)\}$, $\theta_0^* \Leftarrow \theta_0^*(m)$, $s(1) = 1$;

b) wins the team n , if $\theta_0(m) > \theta_0(n)$, then $\Theta_0^* = \{\theta_0^*(n)\} = \min\{\theta_0(m), \theta_0(n)\}$, $\theta_0^* \Leftarrow \theta_0^*(n)$, $s(1) = 1$;

c) competition is draw, if $\theta_0(m) = \theta_0(n)$, then $\Theta_0^* = \{\theta_0^*(m), \theta_0^*(n)\} = \min\{\theta_0(m), \theta_0(n)\}$, $\theta_0^* \Leftarrow \theta_0^*(m) = \theta_0^*(n)$, $s(1) = 2$.

Value of forfeit is equal as follows:

$$\sigma_{s(0)}[k(m,1), k(n,1)] = \int_0^{\theta_0^*} \sigma_{0(m),0(n)}(t) dt, \quad (34)$$

where insertion indices $k(m,1), k(n,1)$ substitution (33).

Substitutions for preparing next step of recursion are as follows:

indices of $\theta_0(m)$, $\theta_0(n)$, should be replaced with

$$s(1) \Leftarrow 0 + \bar{s}(1); \quad (35)$$

$$s(1) = \begin{cases} 1 & \text{in the case a), b);} \\ 2 & \text{in the case c);} \end{cases} \quad (36)$$

time intervals, which will compete further and components of vector $[0(m), 0(n)]$ should be replaced as follows

$$\begin{aligned} \theta_{s(1)}(m) &\Leftarrow \begin{cases} T_{k(m,2)} & \text{in the cases a), c);} \\ \theta_0(m) - \theta_0^* & \text{in the cases b);} \end{cases} \\ \theta_{s(1)}(n) &\Leftarrow \begin{cases} T_{k(n,2)} & \text{in the cases b), c);} \\ \theta_0(n) - \theta_0^* & \text{in the cases a);} \end{cases} \end{aligned} \quad (37)$$

components of vector $(a_{0(m)}, a_{0(n)})$ should be replaced as follows –

$$(a_{0(m)}, a_{0(n)}) \Leftarrow \begin{cases} (a_{1(m)}, a_{0(n)}) & \text{in the case a);} \\ (a_{0(m)}, a_{1(n)}) & \text{in the case b);} \\ (a_{1(m)}, a_{1(n)}) & \text{in the case c).} \end{cases} \quad (38)$$

In such a way on the second step or recursion rigid time intervals $\theta_{s(1)}(m), \theta_{s(1)}(n)$ will compete between them.

Let us assume that on the r -th step of the recursion vector of functional state of semi-Markov process is $(a_{i(m)}, a_{j(n)})$, and time intervals $\theta_{s(r-1)}(m), \theta_{s(r-1)}(n)$ compete between them. Possible result of competition may be the next:

a) wins the team m , if $\theta_{s(r-1)}(m) < \theta_{s(r-1)}(n)$, then $\Theta_{s(r-1)}^* = \{\theta_{s(r-1)}^*(m)\} = \min\{\theta_{s(r-1)}(m), \theta_{s(r-1)}(n)\}$, $\theta_{s(r-1)}^* \Leftarrow \theta_{s(r-1)}^*(m)$, $s(r) \Leftarrow s(r-1) + 1$;

b) wins the team n , if $\theta_{s(r-1)}(m) > \theta_{s(r-1)}(n)$, then
 $\Theta_{s(r-1)}^* = \{\theta_{s(r-1)}^*(n)\} = \min\{\theta_{s(r-1)}(n), \theta_{s(r-1)}^*(n)\}$, $\theta_{s(r-1)}^* \leftarrow \theta_{s(r-1)}^*(n)$, $s(r) \leftarrow s(r-1) + 1$;
 c) competition is draw, if $\theta_{s(r-1)}(m) = \theta_{s(r-1)}(n)$, then
 $\Theta_{s(r-1)}^* = \{\theta_{s(r-1)}^*(m), \theta_{s(r-1)}^*(n)\} = \min\{\theta_{s(r-1)}(m), \theta_{s(r-1)}(n)\}$, $\theta_{s(r-1)}^* \leftarrow \theta_{s(r-1)}^*(m) = \theta_{s(r-1)}^*(n)$,
 $s(r) \leftarrow s(r-1) + 2$.

Value of forfeit is equal as follows:

$$\sigma_{s(r-1)}[k(m, i), k(n, j)] = \int_0^{\theta_{s(r-1)}^*} \sigma_{i(m), j(n)}(t) dt. \quad (39)$$

Substitutions for preparing next step of recursion are as follows:

indices of $\theta_{s(r-1)}(m)$, $\theta_{s(r-1)}(n)$, should be replaced with index of $\bar{s}(r) \leftarrow s(r-1) + \bar{s}(r)$, where $\bar{s}(r)$ is defined as (36);

time intervals, which will compete further and components of vector $[i(m), j(m)]$ should be replaced as follows

$$\theta_{s(r)}(m) \leftarrow \begin{cases} T_{k(m, i+1)} & \text{in the cases a), c);} \\ \theta_{s(r-1)}(m) - \theta_{s(r-1)}^* & \text{in the case b);} \end{cases}$$

$$\theta_{s(r)}(n) \leftarrow \begin{cases} T_{k(n, j+1)} & \text{in the cases b), c);} \\ \theta_{s(r-1)}(n) - \theta_{s(r-1)}^* & \text{in the case a);} \end{cases} \quad (40)$$

components of vector $[i(m), j(n)]$ should be replaced as follows

$$(a_{i(m)}, a_{j(n)}) \leftarrow \begin{cases} (a_{i(m)+1}, a_{j(n)}) & \text{in the case a);} \\ (a_{i(m)}, a_{j(n)+1}) & \text{in the case b);} \\ (a_{i(m)+1}, a_{j(n)+1}) & \text{in the case c).} \end{cases} \quad (41)$$

So on the $(r+1)$ -th step or recursion rigid time intervals $\theta_{s(r)}(m)$, $\theta_{s(r)}(n)$ will compete between them.

Let us assume, that on the last step of recursion only m -th team stays in race, and time, it spend from a previous switch till finishing J -th stage, obtained on previous stage of recursion, is $\theta_{s(R-1)}^*$. Value of forfeit on the last stage is is equal as follows:

$$\sigma_{s(R-1)}[k(m, J-1), k(n, J)] = \int_0^{\theta_{s(R-1)}^*} \sigma_{J(m)-1, J(n)}(t) dt. \quad (42)$$

For evaluation of common forfeit one should to evaluate probability of realization of the $\tilde{k}(m)$ -th evolution trajectory of team m and $\tilde{k}(n)$ -th evolution trajectory of team n . They are as follows:

$$\tilde{p}_{\tilde{k}(m)} = \prod_{i=1}^J \tilde{p}_{k(m, i)}, \quad \tilde{1}(m) \leq \tilde{k}(m) \leq \tilde{K}(m);$$

$$\tilde{p}_{\tilde{k}(n)} = \prod_{j=1}^J \tilde{p}_{k(n, j)}, \quad \tilde{1}(n) \leq \tilde{k}(n) \leq \tilde{K}(n). \quad (43)$$

Weighted sum of forfeit, which the m -th team receives from m -th team when realized $\tilde{k}(m)$ -th and $\tilde{k}(n)$ -th evolution trajectories is as follows:

$$\sigma_{\tilde{k}(m), \tilde{k}(n)}(m, n) = \tilde{p}_{\tilde{k}(m)} \tilde{p}_{\tilde{k}(n)} \sum_{r=0}^{R-1} \sigma_{s(r-1)} [k(m, i), k(n, j)]. \quad (44)$$

Common sum of forfeit, which the m -th team receives from the n -th team is as follows:

Common sum of forfeit, which the m -th team receives from the n -th team, when is as follows

$$\sigma(m, n) = \sum_{\tilde{k}(m)=\tilde{1}(m)}^{\tilde{K}(n)} \sum_{\tilde{k}(n)=\tilde{1}(n)}^{\tilde{K}(m)} \sigma_{\tilde{k}(m), \tilde{k}(n)}(m, n). \quad (45)$$

Common sum of forfeit, which the m -th team receives from all other teams, is as follows

$$\sigma(m) = \sum_{\substack{n=1, \\ n \neq m}}^M \sigma(m, n). \quad (46)$$

It is necessary to admit, that common sum of forfeit, $\sigma(m)$, the m -th team receives from all other teams, rigidly depends of schedules of all participants of races, including the m -th team and probabilities of routes selection. So to change a sum of forfeit $\sigma(m)$ one may both change m -th team schedule, and probabilities of selection possible routes on the every stage of the distance. This is essential obstacle, from point of view of putting and solving the forfeit optimization task.

Conclusion

Results obtained gives to system engineers and economists rather simple method of evaluation the effectiveness of those or that team management decisions on the basis of opponent teams behavior observation. Method of forfeit calculation proposed, gives a solid forfeit figure, based on the real knowledge of the strategies of opponents and possible changes of opponents plans. This, in turn, permits to use the game theory [15, 16] to work out a managing strategy, for increasing sum of forfeit which team under management receives from other teams, participated in relay race.

Further investigation in this area should be directed to finding common recommendation for scheduling m -th team strategy with optimization sum of forfeit [17].

The research was carried out within the state assignment of the Ministry of Education and Science of Russian Federation (No 2.3121.2017/PCH).

References

1. Bellman R.E. *Dynamic Programming*. N.Y.: Dover Publications, Inc, 2003, 384 p.
2. Ivutin A.N., Larkin E.V. Simulation of Concurrent Games. *Bulletin of the South Ural State University. Series: Mathematical Modelling, Programming and Computer Software*, 2015, Vol. 8, no. 2, pp. 43–54. DOI: 10.14529/mmp150204
3. Larkin E.V., Bogomolov A.V., Privalov A.N. A method for estimating the time intervals between transactions in speech-compression algorithms. *Automatic documentation and mathematical linguistics*, 2017, Vol. 51, Issue 5, pp. 214–219. DOI: 10.3103/S000510551705003X
4. Ivutin A.N., Larkin E.V., Kotov V.V. Established Routine of Swarm Monitoring Systems Functioning. *Advances in Swarm and Computational Intelligence. ICSI 2015. Lecture Notes in Computer Science, vol. 9141*, Springer, Cham, 2015, pp. 415–422. DOI: 10.1007/978-3-319-20472-7_45
5. Ivutin A.N., Larkin E.V., Lutskov Yu.I. Simulation of Concurrent Games in Distributed Systems. *Proc. 5th International Workshop on Computer Science and Engineering (WCSE 2015)*, Moscow, Russia, April 15–17, 2015, pp. 60–65.
6. Korolyuk V., Swishchuk A. Semi-Markov random evolutions. *Semi-Markov Random Evolutions. Mathematics and Its Applications. Vol 308*, Springer, Dordrecht, 1995, pp. 59–91. DOI: 10.1007/978-94-011-1010-5_4
7. Larkin E.V., Lutskov Yu.I., Ivutin A.N., Novikov A.S. Simulation of concurrent process with Petri-Markov nets. *Life Science Journal*, 2014, Vol. 11, no. 11, pp. 506–511. DOI: 10.7537/marslsj111114.86

8. Larkin E., Bogomolov A., Privalov A. Data buffering in information-measuring system. *Proc. 2nd International Ural Conference on Measurements (UralCon)*, Chelyabinsk, 2017, pp. 118–123. DOI: 10.1109/URALCON.2017.8120697
9. Shiryayev A.N. *Probability*. Springer, New York, NY, 1996, 623 p. DOI: 10.1007/978-1-4757-2539-1
10. Cleaveland R., Smolka S. Strategic directions in concurrency research. *ACM Computing Surveys (CSUR)*, 1996, Vol. 28, Issue 4, pp. 607–625. DOI: 10.1145/242223.242252
11. Heymann M. Concurrency and Discrete Event Control. *IEEE Control Systems Magazine*, 1990, Vol. 10, Issue 4, pp. 103–112. DOI: 10.1109/37.56284
12. Valk R. Concurrency in Communicating Object Petri Nets. *Concurrent Object-Oriented Programming and Petri Nets. Lecture Notes in Computer Science, vol. 2001*, Springer, Berlin, Heidelberg, 2001, pp. 164–195. DOI: 10.1007/3-540-45397-0_5
13. Dijkstra E.W. Cooperating sequential processes. *Programming Languages*, Academic Press, London, 1968, pp. 43–112.
14. Larkin E.V., Ivutin A.N. "Concurrency" in M-L-Parallel Semi-Markov Process. *MATEC Web of Conferences*, 2017, Vol. 108, Article no. 05003. DOI: 10.1051/mateconf/201710805003
15. Myerson R.B. *Game theory*. Cambridge, London: Harvard University Press, 1997, 568 p.
16. Chatterjee K., Jurdziński M., Henzinger T. Simple stochastic parity games. *Computer Science Logic. CSL 2003. Lecture Notes in Computer Science*, 2003, Vol. 2803, pp. 100–113. DOI: 10.1007/978-3-540-45220-1_11
17. Squillante M.S. Stochastic Analysis and optimization of multiserver systems. *Run-time Models for Self-managing Systems and Applications. Autonomic Systems*. Springer, Basel, 2010, pp. 1–25. DOI: 10.1007/978-3-0346-0433-8_1

Received May 30, 2018

*Bulletin of the South Ural State University
Series "Mathematics. Mechanics. Physics"
2018, vol. 10, no. 3, pp. 30–40*

УДК 519.2

DOI: 10.14529/mmph180304

АЛЬТЕРНАТИВНЫЕ МАРШРУТЫ ИГР С ЖЁСТКИМ РАСПИСАНИЕМ

Е.В. Ларкин¹, А.Н. Привалов²

¹ Тульский государственный университет, г. Тула, Российская Федерация

² Тульский государственный педагогический университет им. Л.Н. Толстого, г. Тула, Российская Федерация

E-mail: privalov.61@mail.ru

Корпоративно-конкурирующая система, которая присутствует внутри корпораций, может быть определена как «игра», шаг за шагом реализующая определённую деятельность. Система функционирует в реальном физическом времени, а результат операции – это «дистанция», которая делится на «этапы». Этапы проходят участники команд в соответствии с «жестким графиком», который может быть выбран из набора возможных графиков. Абстракция «М-параллельный полумарковский процесс» может быть использована для описания рассматриваемой системы. В полумарковском процессе вырожденное распределение используется для описания временных интервалов между промежуточными точками эстафеты. Для анализа развития эстафеты используется рекуррентный метод, учитывающий жесткость графика и стохастический характер выбора маршрута. В соответствии с предложенной концепцией распределенной «неустойки» и предлагаемой рекуррентной процедуры предлагается метод расчета суммирующей «неустойки», которую получает одна из конкурирующих команд от других команд.

Ключевые слова: эстафета; полумарковский процесс; вырожденное распределение; маршрут; эволюция; распределенная неустойка; рекуррентная процедура.

Литература

1. Bellman, R.E. Dynamic Programming / R.E. Bellman. – N.Y: Dover Publications, Inc. – 2003. – 384 p.
2. Ivutin, A.N. Simulation of Concurrent Games / A.N. Ivutin, E.V. Larkin // Вестник ЮУрГУ. Серия «Математическое моделирование и программирование». – 2015. – Т. 8, № 2. – С. 43–54.
3. Larkin, E.V. A method for estimating the time intervals between transactions in speech-compression algorithms / E.V. Larkin, A.V. Bogomolov, A.N. Privalov // Automatic documentation and mathematical linguistics. – 2017. – Vol. 51, Issue 5. – P. 214–219.
4. Ivutin, A.N. Established Routine of Swarm Monitoring Systems Functioning / A.N. Ivutin, E.V. Larkin, V.V. Kotov // Advances in Swarm and Computational Intelligence. ICSI 2015. Lecture Notes in Computer Science, Vol. 9141: сб. науч. тр. – Springer, Cham, 2015. – P. 415–422.
5. Ivutin, A.N. Simulation of Concurrent Games in Distributed Systems / A.N. Ivutin, E.V. Larkin, Yu.I. Lutskov // The 5th International Workshop on Computer Science and Engineering (WCSE 2015). – Moscow, Russia, April 15–17, 2015. – P. 60–65.
6. Korolyuk, V. Semi-Markov random evolutions / V. Korolyuk, A. Swishchuk. – Semi-Markov Random Evolutions. Mathematics and Its Applications. Vol 308: сб. науч. тр. – Springer, Dordrecht, 1995. – P. 59–91.
7. Simulation of concurrent process with Petri-Markov nets / E.V. Larkin, Yu.I. Lutskov, A.N. Ivutin, A.S. Novikov // Life Science Journal. – 2014. – Vol. 11, no. 11. – P. 506–511.
8. Larkin, E. Data buffering in information-measuring system / E. Larkin, A. Bogomolov, A. Privalov // 2-nd International Ural conference on measurements (UralCon). – Chelyabinsk, 2017. – P. 118–123.
9. Shiryayev, A.N. Probability / A.N. Shiryayev. – Springer, New York, NY, 1996. – 623 p.
10. Cleaveland, R. Strategic directions in concurrency research / R. Cleaveland, S. Smolka // ACM Computing Surveys (CSUR). – 1996. – Vol. 28. – Issue 4. – P. 607–625.
11. Heymann, M. Concurrency and Discrete Event Control / M. Heymann // IEEE Control Systems Magazine. – 1990. – Vol. 10. – Issue 4. – P. 103–112.
12. Valk, R. Concurrency in Communicating Object Petri Nets / R. Valk. – Concurrent Object-Oriented Programming and Petri Nets. Lecture Notes in Computer Science, vol. 2001: сб. науч. тр. – Springer, Berlin, Heidelberg, 2001. – P. 164–195.
13. Dijkstra, E.W. Cooperating sequential processes / E.W. Dijkstra // Programming Languages: сб. науч. тр. – Academic Press, London, 1968. – P. 43–112.
14. Larkin, E.V. "Concurrency" in M-L-Parallel Semi-Markov Process / E.V. Larkin, A.N. Ivutin // MATEC Web Conf. – 2017. – Vol. 108. – Article no. 05003.
15. Myerson, R.B. Game theory: analysis of conflict / R.B. Myerson. – Cambridge, London: Harvard University Press, 1997. – 568 p.
16. Chatterjee, K. Simple stochastic parity games / K. Chatterjee, M. Jurdziński, T. Henzinger // Computer Science Logic. CSL 2003. Lecture Notes in Computer Science, vol. 2803: сб. науч. тр. – Springer, Berlin, Heidelberg, 2003. – P. 100–113.
17. Squillante, M.S. Stochastic Analysis and optimization of multiserver systems / M.S. Squillante // Run-time Models for Self-managing Systems and Applications. Autonomic Systems. – Springer, Basel, 2010. – P. 1–25.

Поступила в редакцию 30 мая 2018 г.