УДК 519.872

О ПРИМЕНЕНИИ НМ-СЕТИ ПРИ РЕШЕНИИ НЕКОТОРЫХ ЗАДАЧ БАНКОВСКОЙ СФЕРЫ

Стефанович О.И.
Гродненский государственный университет им. Я. Купалы, г. Гродно
Научный руководитель: Матальцкий М.А., д.ф.-м.н., профессор

Рассмотрим НМ (Howard - Matalytski)-сеть [1], состоящую из *n* систем обслуживания (СМО) $S_1, S_2, ..., S_n$, в которых обслуживаются заявки двух типов. СМО S_i является системой с двумя очередями и m_{ic} линиями обслуживания для заявок типа *c*, $c = 1,2$, в которых отдельно обслуживаются заявки 1-го и 2-го типов, т.е. функционирование СМО S_i можно рассматривать как параллельное функционирование двух отдельных подсистем S_{i1} и S_{i2}, $i = \overline{1,n}$. Под состоянием сети будем понимать вектор

$$k(t) = (k,t) = (k_{11}, k_{12}; k_{21}, k_{22}; ...; k_{n1}, k_{n2}, t),$$

где k_{ic} – число заявок типа *c* в подсистеме S_{ic} в момент времени t, $i = \overline{1,n}$, $c = 1,2$.

Пусть λ_{ic} – интенсивность поступления заявок типа *c* извне в подсистему S_{ic}, $\mu_{ic}(k_{ic})$ – интенсивность обслуживания заявок типа c в каждой из m_{ic} линий подсистемы S_{ic}, $p_{i0}^{(c)}$ – вероятность ухода заявок типа *c* из сети после обслуживания в системе S_i, $p_{ij}^{(c)}$ – вероятность перехода заявки типа *c* из системы S_i в систему S_j $\sum_{j=0}^{n} p_{ij}^{(c)} = 1$, $i = \overline{1,n}$, $c = 1,2$, $i \neq j$. Система S_0 при этом соответствует внешней среде. Заявка при переходе из одной СМО в другую приносит последней СМО некоторый случайный доход и соответственно доход первой СМО уменьшается на эту величину. Обозначим через $V_i(t)$ доход системы S_i в момент времени t, $V_i(t_0) = v_{i0}$.

Описанная НМ – сеть может служить моделью перевозки денежной наличности между условным центральным банком (ЦБ) и периферийными банками (ПБ) города. Количество перевозок денежной наличности, а также количество перевозимой наличности являются случайными величинами. За каждую перевозку ЦБ получает от ПБ определенную сумму, которая определяется в процентах от суммы перевозимой наличности. Система $S_i, i = \overline{1,n}$ соответствует *i*-му банку, подсистема S_{i1} – это отдел банка, который занимается обработкой денег 1-го типа (например, бел. рублей), подсистема S_{i2} – отдел, который занимается обработкой денег 2-го типа (например, валюты).

Опишем также, что означают другие параметры сети:

v_{i0} – резерв банка S_i, $i = \overline{1,n}$;

r_{0i}^{c} – количество денег типа *c*, поступивших извне в банк S_i за одну перевозку, $c = 1,2$;

$R_{i0}^{(1)}$, $R_{i0}^{(2)}$ – соответственно количество денег 1-го и 2-го типов, перевозимых из банка S_i в другие банки, не входящие в нашу банковскую сеть, за одну перевозку $i = \overline{1,n}$;

$r_{ji}^{(1)}$, $r_{ji}^{(2)}$ – соответственно количество денег 1-го и 2-го типов, перевозимых из банка S_j в банк S_i за одну перевозку;

$R_i^{(1)}$, $R_i^{(2)}$ – плата ПБ S_i ЦБ за перевозку денег 1-го и 2-го типов соответственно;

$r_i^{(1)}$, $r_i^{(2)}$ – величины увеличения дохода банка S_i за счет процентов от суммы перевозимой наличности, $i=\overline{1,n}$.

Как только банк установил сумму резервов, для обслуживания клиентов банка, эти деньги выпадают из оборота. Поэтому банк заинтересован в том, чтобы суммы резервов были минимальными. Но если в начале дня установить малые резервы, а затем увеличивать их при необходимости, то за каждое увеличение резерва надо платить. Поэтому важной задачей является нахождение оптимальных резервов банков.

Было получено выражение для ожидаемого дохода системы S_i:

$$v_i(t)=M\{V_i(t)\}=v_{i0}+\sum_{c=1}^{2}v_{ic}(t)=$$

$$\begin{aligned}&=v_{i0}+\sum_{c=1}^{2}\left[(c_i^{(c)}+\lambda_{ic}a_{0i}^{(c)})t+\sum_{j=1}^{n}\mu_{jc}a_{ji}^{(c)}p_{ji}^{(c)}\int_0^t\min(N_{jc}(s),m_{jc})ds-\right.\\&-\mu_{ic}\int_0^t\min(N_{ic}(s),m_{ic})ds\sum_{j=0}^{n}(b_{ij}^{(c)}+b_i^{(c)})p_{ij}^{(c)},\ i=\overline{1,n},\end{aligned} \qquad (1)$$

где $N_{ic}(s)$ – среднее число заявок типа c в подсистеме S_{ic} системы S_i. Величины $N_{ic}(s)$, могут быть найдены с помощью метода, рекуррентного по моментам времени. $M\{r_{0i}^{(c)}\}=a_{0i}^{(c)}$, $M\{R_{i0}^{(c)}\}=b_{i0}^{(c)}$, $M\{r_{ji}^{(c)}\}=a_{ji}^{(c)}$, $M\{R_{ij}^{(c)}\}=b_{ij}^{(c)}$, $M\{R_i^{(c)}\}=b_i^{(c)}$, $M\{r_i^{(c)}\}=c_i^{(c)}, i=\overline{1,n},\ c=1,2$, здесь M – знак математического ожидания.

Рассмотрим частный случай, когда $m_{ic}=1$ и пусть подсистемы S_{ic} функционируют в условиях высокой нагрузки, т.е. $\forall t \quad k_{ic}(t)>0$, $i=\overline{1,n}$, $c=1,2$. В этом случае $\min(N_{ic}(s),1)=1$ и, как следует из (1), ожидаемый доход системы S_i равен:

$$v_i(t)=v_{i0}+g_i t\,,$$

где
$$g_i=\sum_{c=1}^{2}(c_i^{(c)}+\lambda_{ic}a_{0i}^{(c)}-\mu_{ic}b_i^{(c)}+\sum_{j=1}^{n}\mu_{jc}a_{ji}^{(c)}p_{ji}^{(c)}-\mu_{ic}\sum_{j=0}^{n}b_{ij}^{(c)}p_{ij}^{(c)}),$$

v_{i0} – ожидаемый доход системы S_i в начальный момент времени (резерв банка S_i), $i=\overline{1,n}$.

Отсюда следует, что если $g_i\ge 0$, то для того, чтобы

$$v_i(t)\ge 0, \qquad (2)$$

нам достаточно взять $v_{i0}=v_{i0}^{*}=0$; если же $g_i<0$, то для того, чтобы выполнялось неравенство (2) на отрезке времени $[0,T]$, мы должны положить $v_{i0}=v_{i0}^{*}=|g_i|T$.

Ещё одной важной практической задачей является следующая. Разобьём интервал [0; T] точками t_k: $0<t_1<t_2<\cdots<t_{m-1}<t_m=T$. Предположим, что если мы меняем значение v_{i0} в момент времени t_k, то мы платим плату $g_i(t_k)$, т.е. доход системы S_i уменьшается на эту величину. Предположим также, что на интервалах времени $[0;t_1),[t_1;\ t_2),\ldots,[t_{k-1};\ t_k),\ldots,[t_m;T)$ меняются параметры нашей сети $\lambda_{ic},\ \mu_{ic},p_{ij}^{(c)},$ и мы получаем последовательность $g_i^{(1)},g_i^{(2)},\ldots,g_i^{(m)}$. Задача тогда принимает вид ($t_0=0;\ t_m=T$):

$$\sum_{k=1}^{n}\left[\left|g_i^{(k)}\right|\left(t_k-t_{k-1}\right)+g_i(t_k)\right]\to\min_{n,t_1,t_2,\ldots,t_{m-1}},\ i=\overline{1,n}, \qquad (3)$$

где $g_i^{(k)}$ – величина дохода системы S_i при изменении параметров сети $\lambda_{ic}, \mu_{ic}, p_{ij}^{(c)}$ на интервале времени $[t_{k-1}, t_k)$, $i = \overline{1, n}$.

Если $t_1, t_2, ..., t_{m-1}$ – целые числа, то задача (3) является задачей целочисленной оптимизации. Для решения данной задачи применялся метод полного перебора, разработана программа для её решения. Она находит сумму произведений величины резерва на время отвлечения этого резерва, а затем эта сумма минимизируется. Результаты, полученные при помощи программы, могут быть использованы для анализа и принятия управленческих решений.

Список цитированных источников

1. Маталыцкий, М.А. Системы и сети массового обслуживания: анализ и применение: монограмма / М.А. Маталыцкий, О.М. Тихоненко, Е.В. Колузаева. – Гродно: ГрГУ, 2011. – 817 с.

УДК 681.3.06

ПРИЛОЖЕНИЯ ПРОИЗВОДЯЩЕЙ ФУНКЦИИ

Сухоцкий Р.П.
Брестский государственный технический университет, г. Брест
Научный руководитель: Лебедь С.Ф., к.ф.-м.н., доцент

В работе рассмотрено понятие производящей функции и ее приложения.

Определение. Производящей функцией последовательности $\{a_n\}$ называется сумма степенного ряда $f_a(t) = \sum_{n=0}^{\infty} a_n t^n$.

Приложения производящей функции.

- **Комбинаторика.**

Постановка задачи о расстановке черных и белых шаров: Сколькими различными способами можно расположить в линию чёрные и белые шары, общее количество которых равно *n*?

Решение: В этой задаче есть один параметр – число шаров *n*. Решением считается формула, позволяющая получить ответ для любого заданного *n* (в данном случае *n≥0*). Этот ответ будем обозначать символом a_n.

Обозначим белый шар символом – ○, а чёрный – ●. Нулевое количество шаров будем обозначать – o. Получим решение для небольшого значения параметра. Например: n = 2 : ○○, ○●,●○ ,●● $\Rightarrow a_2$= 4; n = 1 : ○, ●$\Rightarrow a_1$ = 2. Единственный способ не располагать в линию ничего – это **ничего не делать**, причём ничего не делать **можно одним способом**.

В случае *n = 3* можно взять самый левый шар белым и **закончить комбинацию** ○… **четырьмя способами**, а можно взять его чёрным, **закончив комбинацию ●… также четырьмя способами**. Значит, $a_3 = 2a_2$. Рассуждая аналогично, приходим к выводу, что $a_n = 2a_{n-1}$ (для *n≥1*), это означает, что $a_n = 2^n$.

Решим данную задачу с помощью производящей функции. Для этого «просуммируем» все возможные комбинации (включая пустую) следующим образом:

А = ø+○ +●+○○ +●○ +○ ●+●●+○○○ +○○ ●+○ ●○ +○ ●●+…