Санкт-Петербургский государственный университет
Факультет прикладной математики — процессов управления
Кафедра информационных систем

Гусев Сергей Михайлович

Выпускная квалификационная работа бакалавра

Нестационарная система обслуживания с конечным источником заявок с относительными приоритетами

Направление 010400
Прикладная математика, фундаментальная информатика и программирование

Заведующий кафедрой,
доктор физ.-мат. наук,
профессор Олемской И. В.

Научный руководитель,
кандидат физ.-мат. наук,
доцент Ерёмин А. С.

Рецензент,
доктор физ.-мат. наук,
профессор Буре В. М.

Санкт-Петербург
2016

Содержание

Введение

Нестационарные системы обслуживания (НСО) — это системы массового обслуживания [1] с конечным количеством заявок и, следовательно, состояний, каждое из которых определяется числом поступивших и числом обслуженных заявок [2]. Бо́льшая часть работ по теории массового обслуживания посвящена стационарному режиму систем с неограниченными заявками и его характеристикам, таким как пропускная способность, среднее количество заявок в очереди и т. д. Но он существует только у тех систем, у которых задания обслуживаются в среднем быстрее, чем поступают. Потому для многих задач практический интерес представляет исследование нестационарных систем. Поскольку они завершают обслуживание за конечное время, то вместо стационарного режима рассматривают переходный процесс и его характеристики, такие как вероятность того, что длина очереди превысит заданное число, вероятность нахождения в свободном состоянии, математическое ожидание времени, за которое будет обработано некое количество заявок, вероятность завершения обслуживания за фиксированное время и т. д.

Вероятности нахождения системы в том или ином состоянии есть функции от времени, которые являются решением обратного уравнения Чепмена — Колмогорова [3]:

$$\frac{dP(t)}{dt} = AP(t), \tag{1}$$

с начальным условием:

$$P(0) = (1, 0, \dots, 0)^T, \tag{2}$$

где $P(t)$ — вектор-функция, i-й элемент которой равен вероятности нахождения системы в i-м состоянии в момент t, а A — матрица интенсивностей, где A_{ij} — интенсивность перехода из j-го состояния в i-е при $i \neq j$, A_{ii} — суммарная интенсивность переходов из i-го состояния.

Простая нестационарная система обслуживания

Рассмотрим одноканальную нестационарную систему обслуживания, на вход которой поступает N заявок с интенсивностями поступления и обработки, зависящими от номера заявки. Дисциплина обслуживания — FIFO (First in first out), что значит, что заявки обслуживаются в порядке их поступления. Интервалы времени между поступлением заявок, случайны и подчиняются экспоненциальному закону распределения. Интенсивности поступления обозначим вектором λ, обработки — μ, p-е компоненты векторов относятся к p-й заявке. Интенсивность потока равна среднему количеству событий, происходящих в единицу времени. Данной системе можно сопоставить граф, изображенный на рис. 1, вершины которого соответствуют состояниям системы, а стрелки — возможным переходам. У вершин указано число поступивших и обработанных заявок, у стрелок — интенсивности переходов. Общее количество состояний будет равно $\frac{(N+1)(N+2)}{2}$. Значение вероятности нахождения в состоянии с i поступившими и j обработанными заявками можно найти из уравнения:

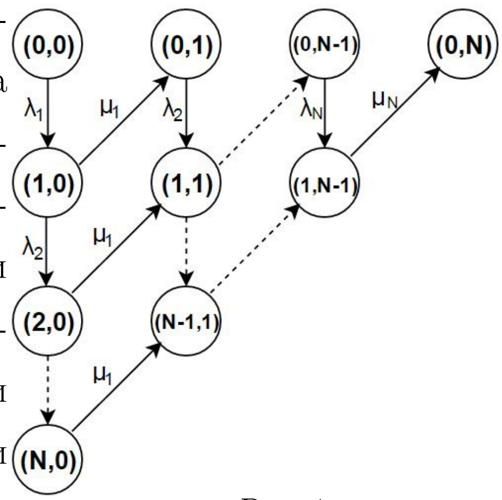

Рис. 1

$$\frac{dP_{i,j}(t)}{dt} = \theta(i) \times (P_{i-1,j}(t)\lambda_{i+j} - P_{i,j}(t)\mu_{j+1}) + \theta(j) \times P_{i+1,j-1}(t) \times \mu_j - \\ -\theta(N-i-j) \times P_{i,j}(t)\lambda_{i+j+1},$$

где $\theta(x) = \begin{cases} 1, & x > 0; \\ 0, & x \leqslant 0, \end{cases}$ $P_{i,j}(0) = \begin{cases} 1, & i+j = 0; \\ 0, & i+j \neq 0. \end{cases}$

Нестационарные системы обслуживания с относительными приоритетами

В данной работе рассматриваются нестационарные системы обслуживания с относительными приоритетами [4], это значит, что в НСО поступает несколько независимых пуассоновских потоков заявок. Заявки более высокого приоритета обрабатываются в первую очередь, но уже начатое обслуживание заявки, пусть даже более низкого приоритета не прерывается. Время обработки заявок, так же как и поступления, опять же, подчиняется экспоненциальному закону распределения с различными интенсивностями для каждой заявки. То есть если обозначить за λ интенсивность поступления заявки, то время, через которое она поступит, есть случайная величина с плотностью распределения равной $\lambda e^{-\lambda t}$, $t > 0$.

Постановка задачи

Системы с приоритетами также описываются системой уравнений вида (1). Для расчета характеристик переходного процесса НСО с известными интенсивностями поступления и обработки заявок требуется построить матрицу A, соответствующую возможным переходам между состояниями.

Систему (1) можно решить аналитически, если известны собственные числа матрицы A. Поскольку система не имеет возвратных состояний, существует возможность упорядочить состояния таким образом, чтобы матрица A была треугольной. В таком случае собственные числа будут в явном виде представлены на её диагонали.

К сожалению, граф, соответствующий НСО с приоритетами, в общем случае, не обладает свойством планарности, потому наглядно изобразить его возможно только для самых простых случаев. Так, например, на рис. 2 изображен граф, соответствующий НСО с всего лишь двумя заявками разных приоритетов. В вершинах обозначения даны следующим образом: в первой строке обозначено количество поступивших, но не обслуженных и обслуженных заявок высокого приоритета, во второй — то же самое для заявок низкого приоритета. Простыми стрелками обозначены переходы, соответствующие поступлению заявки, пунктирными —обслуживанию. Пример приведен для того, чтобы обозначить важное свойство НСО с относительными приоритетами — наличие состояний, отличающихся только лишь номером приоритета заявки, обслуживаемой в данный момент. Данная особенность значительно увеличивает количество состояний системы.

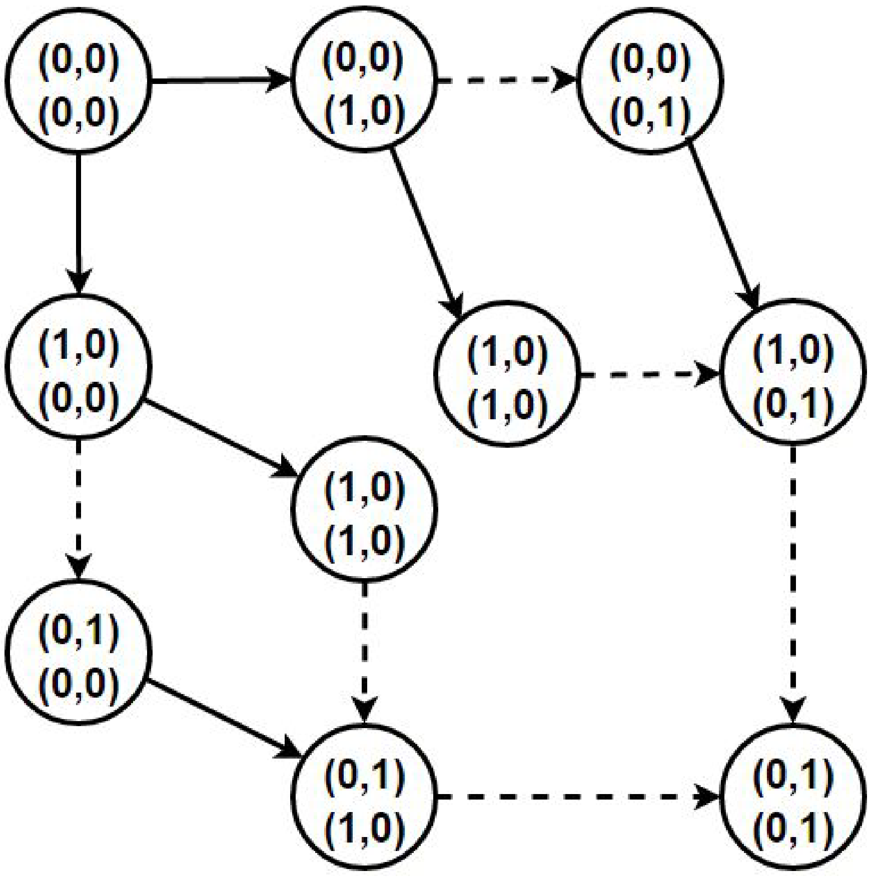

Рис. 2: Граф, соответствующий простейшей НСО с двумя приоритетами

Теория

Введем следующие обозначения:

- r — количество потоков заявок с различными приоритетами, заявки из потока с бо́льшим номером, имеют более высокий приоритет,
- N — вектор количества заявок каждого приоритета $N = (N_1, \ldots, N_r)$,
- i — вектор количества поступивших, но не обработанных заявок каждого приоритета, $i = (i_1, \ldots, i_r)$,
- j — вектор количества обработанных заявок каждого приоритета, $j = (j_1, \ldots, j_r)$,
- $\lambda_1, \ldots, \lambda_r$ — векторы интенсивностей поступления заявок, $\lambda_p = (\lambda_{p,1}, \ldots, \lambda_{p,N_p})$,
- $\mu_1, \ldots, \mu_r$ — векторы интенсивностей обработки заявок, $\mu_p = (\mu_{p,1}, \ldots, \mu_{p,N_p})$,
- d — приоритет заявки, обрабатываемой в текущий момент (для определенности будем считать, что $d = 0$ при $i = (0, \ldots, 0)$),
- $k(i, j, d)$ — функция, отображающая параметры состояния в его номер.

Таким образом, НСО с относительными приоритетами описывается векторами $\lambda_1, \ldots,\ \lambda_r,\ \mu_1, \ldots,\ \mu_r$, а каждое её состояние можно обозначить вектором $(i_1, \ldots, i_r, j_1, \ldots, j_r, d)$, или, короче, (i, j, d), где $0 \leqslant i_p + j_p \leqslant N_p$, $p = 1, \ldots, r$, $d = 0, 1, \ldots, r$.

В начальный момент времени система находится в нулевом состоянии, то есть без поступивших и обработанных заявок.

Из каждого состояния система может перейти в не более чем $r + 1$ состояний, где r возможных переходов — поступление новой заявки и ещё один — обработка. То есть возможны следующие переходы:

- $(i_1, \ldots, i_r, j_1, \ldots, j_r, d) \to (i_1, \ldots, i_p + 1, \ldots, i_r, j_1, \ldots, j_r, d)$, если $i_p + j_p < N_p$, $p = 1, \ldots, r$, причем интенсивность перехода будет равна $\lambda_{p, i_p + j_p + 1}$. Этот переход соответствует поступлению заявки приоритета p.

- $(i_1,\ldots,i_r,j_1,\ldots,j_r,d) \to (i_1,\ldots,i_d-1,\ldots,i_r,j_1,\ldots,j_d+1,\ldots,j_r,m)$, если $d \neq 0$, интенсивность перехода — μ_{d,j_d}, m равен наибольшему индексу среди ненулевых элементов вектора i нового состояния.

  Этот переход соответствует обработке заявки приоритета d.

Следовательно, при использовании следующих замен:
$\hat{k}(i,j,d,p) = k(i_1,\ldots,i_p-1,\ldots,i_r,j_1,\ldots,j_r,d\times\theta(\sum\limits_{l=1}^{r} i_l - 1))$,
$\bar{k}(i,j,d) = k(i_1,\ldots,i_p+1,\ldots,i_r,j_1,\ldots,j_p-1,\ldots,j_r,d)$,
k-е уравнение системы (1) примет следующий вид:

$$\dot{P}_{k(i,j,d)}(t) = \sum_{p=1}^{r}\left[\theta(i_p)\times\lambda_{p,i_p+j_p}\times P_{\hat{k}(i,j,d,p)}(t)\right] +$$
$$+\theta(d-\max_{i_p>0}(p)+1)\times\sum_{p=1}^{r}\left[\theta(j_p)\times\mu_{p,j_p}\times P_{\bar{k}(i,j,p)}(t)\right] -$$
$$-\left(\sum_{p=1}^{r}\left[\theta(N_p-i_p-j_p)\lambda_{p,i_p+j_p+1}\right]+\theta(d)\mu_{d,j_d+1}\right)\times P_{k(i,j,d)}(t)$$

Рано или поздно НСО заканчивает обработку всех заявок и остается в состоянии $(0,N,0)$, дальнейшие переходы невозможны.

Алгоритм составления матрицы A

Для того чтобы матрица A была нижнетреугольной, требуется чтобы состояния с меньшим номером не зависели от состояний с бóльшим. Для этого нужно разбить множество состояний на подмножества с одинаковым числом обработанных заявок наивысшего приоритета и упорядочить их по его возрастанию, затем каждое подмножество аналогичным образом разбить на подмножества и упорядочить их относительно количества поступивших, но ещё не обработанных заявок с наивысшим приоритетом. Затем следует повторять данную операцию для каждого приоритета, в порядке его убывания. Также, для определенности, будем считать, что состояния, отличающиеся лишь параметром d, будут также упорядочены по его возрастанию. То есть упорядочивание состояний идет по возрастанию параметров в следующем порядке: j_r, i_r, j_{r-1}, i_{r-1},..., j_1, i_1, d.

Алгоритм формирования матрицы A последовательно пробегает по всем состояниям системы и заполняет соответствующие им столбцы матрицы интенсивностями переходов в другие состояния в соответствии с принятой нумерацией.

Данный алгоритм был реализован на языке C#.

k	i_1	i_2	j_1	j_2	d
1	0	0	0	0	0
2	1	0	0	0	1
3	0	0	1	0	0
4	0	1	0	0	2
5	1	1	0	0	1
6	1	1	0	0	2
7	0	1	1	0	2
8	0	0	0	1	0
9	1	0	0	1	1
10	0	0	1	1	0

Таблица 1: Порядок состояний простейшей НСО с двумя приоритетами

Так, в табл. 1 приведен порядок состояний НСО с двумя поступающими заявками разных приоритетов, а матрица A примет следующий вид:

$$A = \begin{bmatrix} -(\lambda_1+\lambda_2) & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ \lambda_1 & -(\lambda_2+\mu_1) & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & \mu_1 & -\lambda_2 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ \lambda_2 & 0 & 0 & -(\lambda_1+\mu_2) & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & \lambda_2 & 0 & 0 & -\mu_1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & \lambda_1 & 0 & -\mu_2 & 0 & 0 & 0 & 0 \\ 0 & 0 & \lambda_2 & 0 & \mu_1 & 0 & -\mu_2 & 0 & 0 & 0 \\ 0 & 0 & 0 & \mu_2 & 0 & 0 & 0 & -\lambda_1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & \mu_2 & 0 & \lambda_1 & -\mu_1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & \mu_2 & 0 & \mu_1 & 0 \end{bmatrix}$$

Также стоит заметить, что нижнетреугольная матрица позволяет находить вероятность нахождения системы в состоянии с номером k не решая систему дифференциальных уравнений полностью, а лишь меньшую систему из не более чем k первых уравнений.

Сравнение результатов с имитационной моделью

В пакете MatLab была написана функция, имитирующая поведение НСО с девятью поступающими заявками, пять из которых имеют высокий приоритет. Параметры данной НСО: $\lambda_1 = (0.6, 0.8, 2, 1.5)$, $\mu_1 = (0.5, 0.7, 3, 1.2)$, $\lambda_2 = (1, 1.1, 1.5, 1.3, 0.7)$, $\mu_2 = (0.7, 1.2, 0.8, 1, 0.8)$. Имитационная модель запускается много раз, и значения вероятностей нахождения в определённых состояниях в каждый момент времени находятся отношением количества попаданий в эти состояния к общему числу запусков.

Для той же НСО при помощи реализации описанного алгоритма была найдена матрица A. Затем система (1) была решена методом, описанным в статье [5]. На рис. 3 представлено сравнение решения с результатами 100-кратной имитации на промежутке времени от 0 до 30 с. Сравниваются графики вероятности нахождения системы в свободном состоянии, то есть без заявок, которые ждут обработки. Пунктирная линия — решение системы (1), непрерывная — результаты имитации. Максимальная разница между значениями на рассматриваемом промежутке — 0.0734.

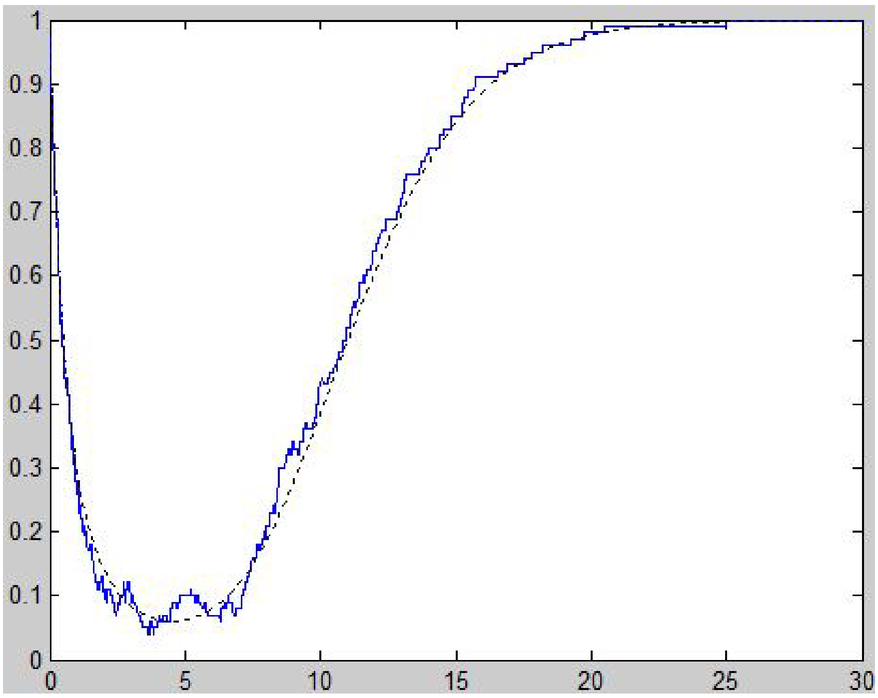

Рис. 3: Вероятность свободного состояния, максимальная разница между 100-кратной имитацией и решением системы ОДУ — 0.0734.

Для получения более точных результатов следующие сравнения проводились уже с результатами 100000-кратной имитации. Визуально решение и результаты имитации становятся неразличимы (см. рис. 4–6).

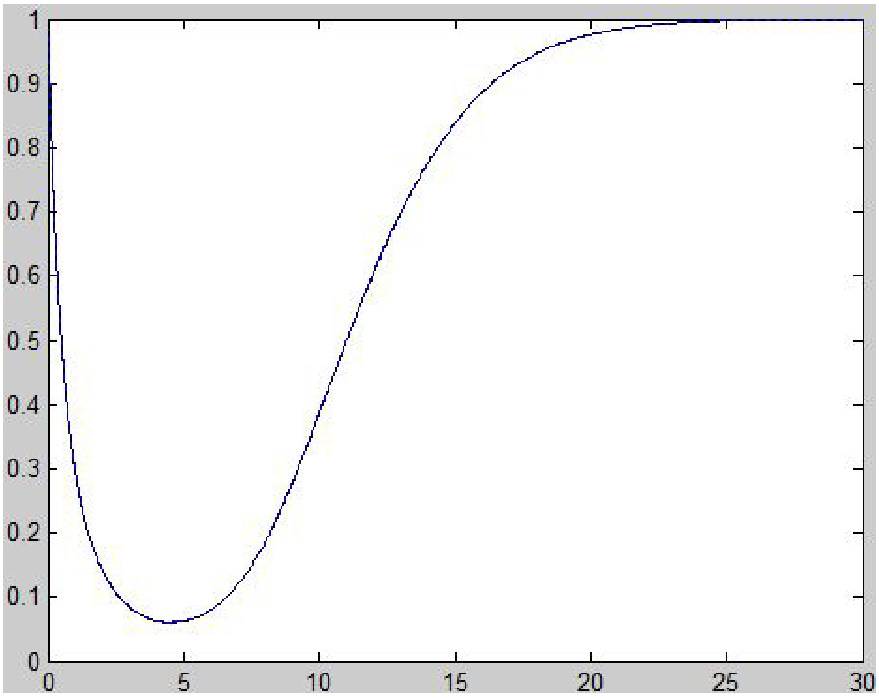

Рис. 4: Вероятность свободного состояния, максимальная разница — 0.0033

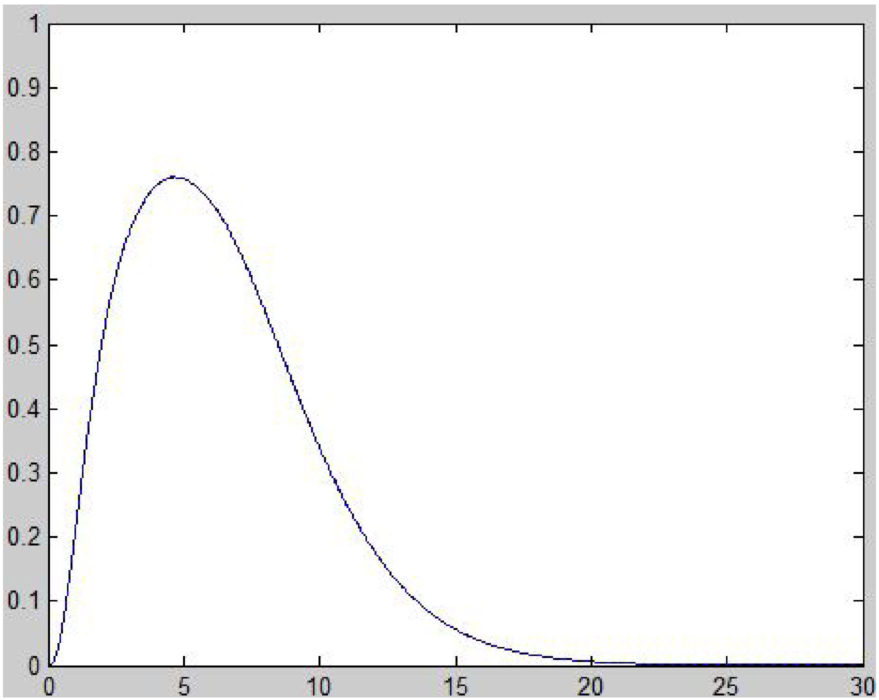

Рис. 5: Вероятность того, что суммарное количество заявок в очереди превысит три, максимальная разница — 0.0022

Так например, матожидание времени, за которое будут обслужены все заявки, вычисленное через решение системы (1), составило 11.7172 с, а среднее время обработки всех заявок по результатам имитации — 11.7203 с. Таким образом, абсолютная разница этих двух значений составляет менее 0.003 с, а относительная — менее 0.03%, что свидетельствует о правильности формирования матрицы A.

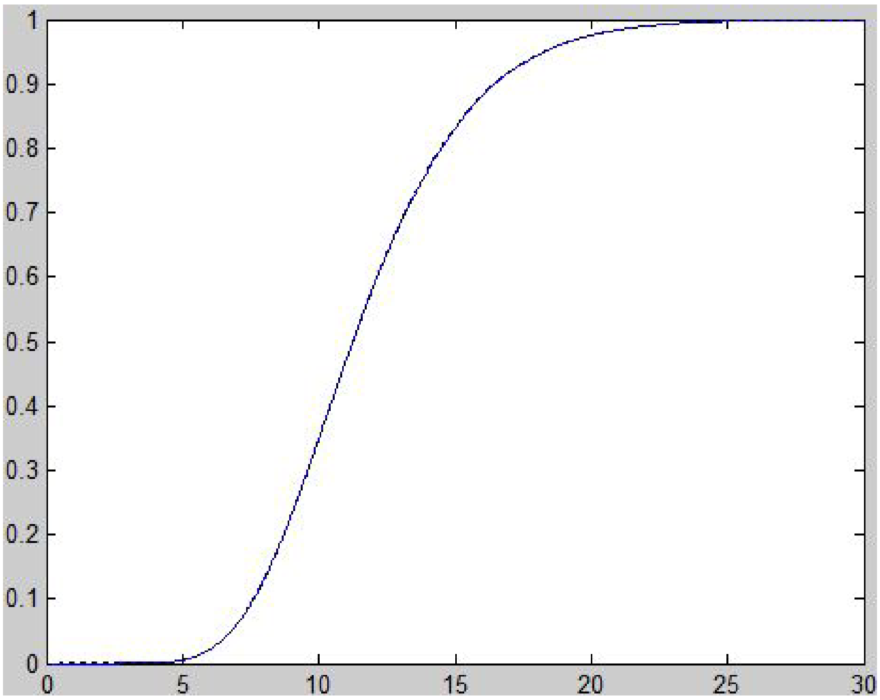

Рис. 6: Вероятность обработки всех заявок, максимальная разница — 0.0027

Заключение

В рамках данной работы был разработан алгоритм составления уравнений уравнений Чепмена — Колмогорова и написана его программная реализация, которая, в совокупности с реализацией метода, описанного в [5], может быть использована для расчета надежности и пропускной способности аппаратно-программных комплексов, у которых множество выполняемых заданий можно разделить на группы с разными приоритетами. В рассмотренном примере для простоты были использованы заявки только двух приоритетов, но сконструированный алгоритм применим к системам с любым количеством приоритетов. В будущем возможно расширение метода для применения к многоканальным системам и/или системам с неэкспоненциальными законами распределения времен поступления и обработки заявок.

Список литературы

[1] Клейнрок Л. Теория массового обслуживания; пер. с анг. И. И. Грушко, ред. В. И. Нейман. М.: Машиностроение, 1979. 432 с.

[2] Бубнов В. П., Сафонов В. И. Разработка динамических моделей нестационарных систем обслуживания. СПб.: Издательство «Лань», 1999. 64 с.

[3] Бубнов В. П., Тырва А. В., Еремин А. С. Комплекс моделей нестационарных систем обслуживания с распределениями фазового типа // Труды СПИИРАН. 2014. Вып. 37. С. 61–71.

[4] Матвеев В. Ф., Ушаков В. Г. Системы массового обслуживания. М.: Изд-во МГУ, 1984. 240 с.

[5] Бубнов В. П., Еремин А. С., Сергеев С. А. Особенности программной реализации численно-аналитического метода расчёта моделей нестационарных систем обслуживания // Труды СПИИРАН. 2015. Вып. 38. С. 218–232.