МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ

федеральное государственное автономное образовательное учреждение высшего образования

Санкт-Петербургский национальный исследовательский университет информационных технологий, механики и оптики

Мегафакультет трансляционных информационных технологий

Факультет информационных технологий и программирования

Лабораторная работа № 4
Модели массового обслуживания с ожиданием

По дисциплине «Прикладная математика»

Выполнили студенты групп
М*33011*
Лунев Илья Андреевич
Семенов Георгий Витальевич
Смирнов Сергей Викторович

Преподаватель:
Москаленко Мария Александровна

САНКТ-ПЕТЕРБУРГ

2021

ПОСТАНОВКА ЗАДАЧИ

В систему, состоящую из r рубительных машин, поступает простейший поток бревен с интенсивностью λ. Каждая машина имеет показательный закон рубки с интенсивностью μ. Если количество бревен, поступивших в рубку, больше числа машин, то образуется очередь, длина которой ограничена и не может превосходить m единиц. Требуется проанализировать работу цеха, как системы массового обслуживания с очередью конечной длины.

ПРАКТИЧЕСКАЯ ЧАСТЬ

По результатам выполненной лабораторной работы представляется отчет, в котором должны содержаться следующие пункты:

1. указание возможных состояний системы;
2. граф состояний;
3. система алгебраических уравнений, составленная относительно стационарных вероятностей; решение системы;
4. расчет стационарных характеристик работы системы;
5. система дифференциальных уравнений, неизвестными которой являются вероятности состояний, зависящие от времени;
6. вычисление нестационарных характеристик; определение коэффициентов загрузки и простоя системы как функций времени;
7. выводы по результатам исследований.

Вариант 2			
Интенсивность входящего потока заявок λ, час$^{-1}$	**Интенсивность обслуживания заявки μ, час$^{-1}$**	**Количество каналов r**	**Возможная длина очереди m**
4,46	2,7	1	2

Согласно кодировке, предложенной Д. Г. Кендаллом данная система массового обслуживания относится к классу М/М/1/2 – одноканальная система с очередью с пуассоновским входящим потоком заявок и экспоненциальным временем обслуживания.

1. указание возможных состояний системы

Возможные состояния системы:

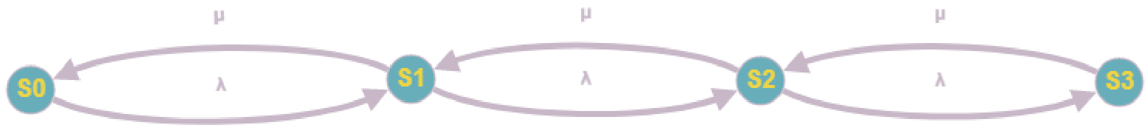
S_0 – в системе бревен нет, машины свободны от рубки

S_1 – в системе 1 бревно, одна машина занята рубкой

S_2 – в системе 2 бревна, одна машина занята рубкой, одно бревно находится в очереди

S_3 – в системе 3 бревна, одна машина занята рубкой, два бревна находятся в очереди

2. граф состояний



Переходы слева направо связаны с поступлением в систему очередного бревна, поэтому все интенсивности переходов одинаковы и равны $\lambda = 4{,}46$ час$^{-1}$. Переходы справа налево обусловлены окончанием рубки бревна и равны $\mu = 2{,}7$ час$^{-1}$.

3. система алгебраических уравнений, составленная относительно стационарных вероятностей; решение системы

Пусть p_k – стационарная вероятность пребывания системы в состоянии S_k, $k = 0,1,2,3$. Тогда имеет место следующая система алгебраических уравнений, описывающая стационарный режим:

$$\begin{cases} -\lambda p_0 + \mu p_1 = 0 \\ \lambda p_0 - (\mu + \lambda) p_1 + \mu p_2 = 0 \\ \lambda p_1 - (\mu + \lambda) p_2 + \mu p_3 = 0 \\ \lambda p_2 - \mu p_3 = 0 \\ p_0 + p_1 + p_2 + p_3 = 1 \end{cases}$$

$$\begin{cases} -4{,}46 p_0 + 2{,}7 p_1 = 0 \\ 4{,}46 p_0 - (2{,}7 + 4{,}46) p_1 + 2{,}7 p_2 = 0 \\ 4{,}46 p_1 - (2{,}7 + 4{,}46) p_2 + 2{,}7 p_3 = 0 \\ 4{,}46 p_2 - 2{,}7 p_3 = 0 \\ p_0 + p_1 + p_2 + p_3 = 1 \end{cases}$$

Программное решение:

```python
lmbds = [lmbd] * (states_count - 1) + [0.0]
mus = [min(r, i) * mu for i in range(states_count)]

print("Интенсивности перехода из i-го состояния в (i+1)-ое:", lmbds, sep="\n")
print("Интенсивности перехода из i-го состояния в (i-1)-ое:", mus, sep="\n")

A = np.zeros(shape=(states_count, states_count))

for i in range(states_count):
    if (i > 0):
        A[i, i - 1] = lmbds[i - 1]
    A[i, i] = -(lmbds[i] + mus[i])
    if (i < states_count - 1):
        A[i, i + 1] = mus[i + 1]

print("Система уравнений:", A, sep="\n")

# Теперь надо решить систему уравнений A = 0
# при этом p0 + p1 + ... + p_{m} = 1

# решим
As = [lmbds[0] / mus[1]]
for i in range(states_count - 2):
    As.append(As[i] * (lmbds[i + 1] / mus[i + 2]))
As = [1.0] + As

# Пусть у нас есть ответы (стационарная точка)
p = np.zeros(shape=states_count)

p_sum = np.sum(As, axis=0)
for i in range(len(p)):
    p[i] = As[i] / p_sum

print("Стационарное решение:", p)
```

Вычисленные стационарные вероятности в данном случае равны

$$p_0 = 0.1011$$
$$p_1 = 0.1671$$
$$p_2 = 0.2760$$
$$p_3 = 0.4558$$

4. расчет стационарных характеристик работы системы

Пусть X – число машин, занятых рубкой. Это есть случайная величина с возможными значениями: $0, 1$. Вероятности этих значений соответственно равны:

$$P(X = 0) = p_0 = 0.1011$$
$$P(X = 1) = p_1 + p_2 + p_3 = 0.1671 + 0.2760 + 0.4558 = 0.8989$$

Тогда среднее число машин, занятых рубкой, есть математическое ожидание случайной величины X, которое равно:

$$M(X) = 0 * P(X = 0) + 1 * P(X = 1) = 0.8989$$

Следовательно, среднее число работающих машин равно 0.8989 .

Пусть Y – число машин, свободных от рубки. Это есть случайная величина с возможными значениями: $0, 1$. Вероятности этих значений соответственно равны:

$P(Y = 0) = p_1 + p_2 + p_3 = 0.1671 + 0.2760 + 0.4558 = 0.8989$
$P(Y = 1) = p_0 = 0.1011$

Тогда среднее число машин, свободных от рубки, есть математическое ожидание случайной величины Y, которое равно

$$M(Y) = 0 * P(Y = 0) + 1 * P(Y = 1) = 0.1011$$

Следовательно, среднее число простаивающих машин равно 0.1011. Общее число занятых и свободных от рубки машин равно

$$M(X) + M(Y) = r = 1$$

Коэффициент загрузки машин равен отношению среднего числа загруженных машин к общему числу машин в цехе, т.е.

$$k_з = \frac{M(X)}{r}$$
$$k_з = 90\%$$

Коэффициент простоя машин равен отношению среднего числа машин, свободных от рубки, к общему числу машин в цехе, т.е.

$$k_п = \frac{M(Y)}{r}$$
$$k_п = 10\%$$

Пусть Z – число бревен в очереди. Это есть случайная величина с возможными значениями: 0, 1, 2. Вероятности этих значений соответственно равны
$P(Z = 0) = p_0 + p_1 = 0.1011 + 0.1671 = 0.2682$
$P(Z = 1) = p_2 = 0.2760$
$P(Z = 2) = p_3 = 0.4558$

Тогда среднее бревен в очереди есть математическое ожидание случайной величины Z, которое равно

$$M(Z) = 0 * P(Z = 0) + 1 * P(Z = 1) + 2 * P(Z = 2) = 0.2760 + 2 * 0.4558 = 1.1876$$

Таким образом, среднее число бревен, находящихся в очереди на рубку, равно 1.1876, т.е. менее двух бревен.

5. система дифференциальных уравнений, неизвестными которой являются вероятности состояний, зависящие от времени

Обозначим через $p_k(t)$ вероятность пребывания системы в момент времени t в состоянии S_k, $k = 0,1,2,3$. Это – переходные вероятности, изменяющиеся со временем.

Нестационарный режим описывается следующей системой дифференциальных уравнений, составленной по графу состояний

$$\begin{cases} p_0{}'(t) = -\lambda p_0(t) + \mu p_1(t) \\ p_1{}'(t) = \lambda p_0(t) - (\mu\ +\ \lambda) p_1(t) + \mu p_2(t) \\ p_2{}'(t) = \lambda p_1(t) - (\mu\ +\ \lambda) p_2(t) + \mu p_3(t) \\ p_3{}'(t) = \lambda p_2(t) - \mu p_3 \end{cases}$$

$$\begin{cases} p_0{}'(t) = -4{,}46 p_0(t) + 2{,}7 p_1(t) \\ p_1{}'(t) = 4{,}46 p_0(t) - (2{,}7\ +\ 4{,}46) p_1(t) + 2{,}7 p_2(t) \\ p_2{}'(t) = 4{,}46(t) - (2{,}7\ +\ 4{,}46) p_2(t) + 2{,}7 p_3(t) \\ p_3{}'(t) = 4{,}46 p_2(t) - 2{,}7 p_3 \end{cases}$$

Решение этой системы должно удовлетворять начальному условию

$p_0(0) = 1, p_1(0) = 0, p_2(0) = 0, p_3(0) = 0$

означающему, что в момент времени $t = 0$ система находится в состоянии S_0: бревен в цехе нет, рубительные машины простаивают и очередь отсутствует.

Программное решение:

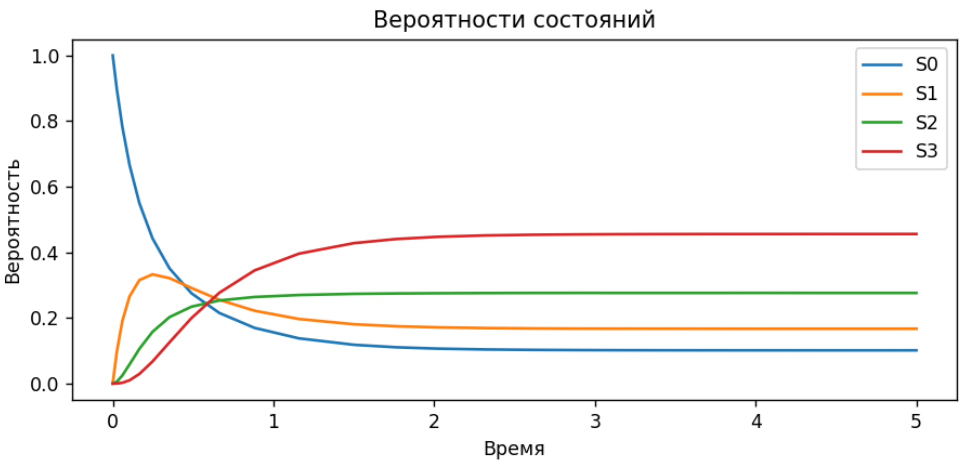
```
from scipy.integrate import solve_ivp
def rhs(s, v):
    equations = []
    for i in range(states_count):
        eq = 0.0
        for j in range(states_count):
            eq += A[i, j] * v[j]
        equations.append(eq)
    return equations

res = solve_ivp(rhs, (0, 5), [1.0] + [0.0] * (states_count - 1))

# нестационарное решение ( функции )
p_diff = np.zeros(shape=states_count)
for i in range(len(res.y)):
    states_ax.plot(res.t, res.y[i], label="S" + str(i))
    p_diff[i] = res.y[i, -1]

print("Нестационарное решение:", p_diff)
```

График:



6. вычисление нестационарных характеристик; определение коэффициентов загрузки и простоя системы как функций времени

Рассчитанные вероятности состояний системы позволяют определить коэффициенты загрузки и простоя системы в зависимости от времени. Для СМО вида $M/M/2/1$ указанные соотношения принимают вид

$$k_з = \frac{p_1(t) + p_2(t) + p_3(t)}{1}$$

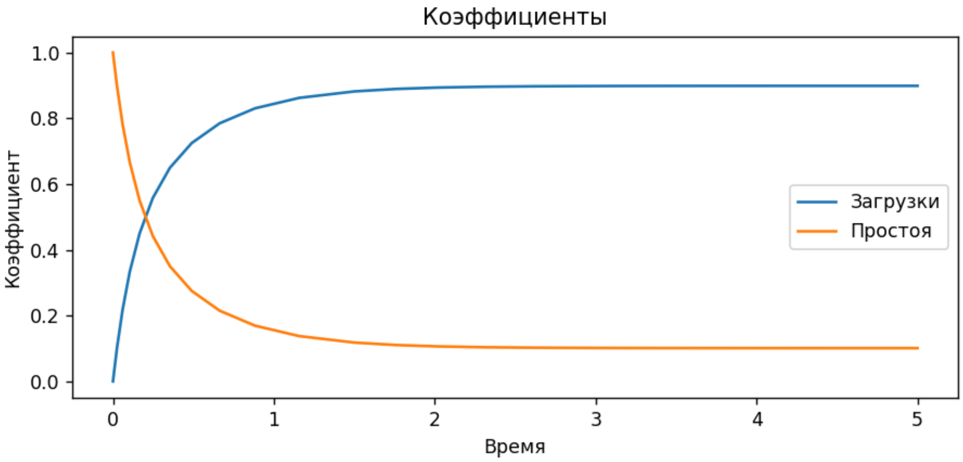
$$k_п = \frac{p_0(t)}{1}$$

Программное решение:

```
busy_coef_dyn = (
    sum(k * res.y[k] for k in range(r))
    + r * sum(res.y[k] for k in range(r, states_count))
) / r

free_coef_dyn = sum(k * res.y[r - k] for k in range(1, r + 1)) / r
```

Видим, что функции и с течением времени приближаются к своим стационарным значениям, равным 0,9 и 0,1 соответственно.



$$k_3(t) = \frac{M(X)}{r} = \frac{\sum_{k=0}^{r-1} k p_k(t) + r \sum_{k=r}^{r+m} p_k(t)}{r}.$$

$$k_П(t) = \frac{M(Y)}{r} = \frac{0 \cdot \sum_{k=r}^{r+m} p_k(t) + \sum_{k=1}^{r} k p_{r-k}(t)}{r}.$$

7. выводы по результатам исследований.

Из полученных графиков можно сделать следующие выводы.

С течением времени загрузка машин возрастает, и уже через 1.5 единиц времени становится близкой к 90%. Коэффициент простоя машин с течением времени убывает и через 1.5 единиц времени приближается к 10%. Это значит, что машины загружены достаточно сильно и вполне возможно осуществление в цехе таких мероприятий, как

- уменьшение потока бревен,
- установка в цехе еще одной рубительной машины,
- повышение эффективности обработки уже работающих машин

КОНТРОЛЬНЫЕ ВОПРОСЫ

λ *(интенсивность потока заявок)* — среднее количество заявок, поступающих в систему за единицу времени.

μ *(интенсивность обслуживания)* — среднее количество заявок, обслуживаемых приборов за единицу времени.

1. Что такое одноканальная система?

Каналом обслуживания называется устройство в СМО, обслуживающее заявку. СМО, содержащее один канал обслуживания, называется одноканальной, а содержащее более одного канала обслуживания – многоканальной (например, 3 кассы на вокзале).

2. Что такое однофазовая система?

СМО могут быть однофазными и многофазными. В первом случае заявка обслуживается только одним прибором, после чего покидает систему, например, покупатель билета в театре. Во втором случае заявка должна пройти некоторую последовательность приборов. Например, в сберкассе, прежде чем получить деньги, человек сначала должен быть обслужен контролером и только потом — кассиром.

3. Что такое очередь?

В системе с очередью (или с ожиданием), если все каналы заняты обслуживанием, то вновь пришедшая заявка становится в очередь (поступает в бункер или накопитель) емкостью m. При этом различают СМО с ограниченной и неограниченной очередью. Примером первых СМО может служить мойка для автомашин с маленькой стоянкой для ожидающих машин, а примером вторых СМО может служить билетная касса или метрополитен.

4. Что такое распределение времени обслуживания?

Закон распределения времени обслуживания — соотношение t времени обслуживания и вероятности P. Зачастую рассматриваются системы с экспоненциальным распределением.

G — распределение общего вида, Ek — распределение Эрланга порядка k, M — экспоненциальное распределение, D — распределение постоянной величины.

5. Что означает и как определяется среднее время в очереди?

$$T_{оч} = \frac{L_{оч}}{A} = \frac{L_{оч}}{p_{обс} \cdot \lambda}$$

$L_{оч}$ — среднее число заявок в очереди, A — абсолютная пропускная способность, $p_{обс}$ — относительная пропускная способность, λ — интенсивность поступающих заявок

6. Что означает и как определяется среднее время в системе?

$$T_{смо} = \frac{L_{смо}}{A}$$

$L_{СМО}$ — среднее время заявки в системе, $L_{оч}$ — среднее число заявок в очереди, $p_{обс}$ — относительная пропускная способность (вероятность обслуживания поступившей заявки), λ — интенсивность поступающих заявок, μ — интенсивность одного канала

7. Что означает и как определяется среднее число клиентов в очереди?

Среднее число заявок (клиентов) в очереди (если очередь есть); определяется как математическое ожидание случайной величины m – числа заявок, состоящих в очереди

$$\mathrm{L}_{оч} = M(m) = \sum_{i=1}^{m} i * p_{r+i}$$

где p_{r+i} - вероятность нахождения в очереди i заявок

8. Что означает и как определяется среднее число клиентов в системе?

$$L_{смо} = L_{оч} + \frac{\lambda p_{обс}}{\mu}$$

$L_{оч}$ — среднее число заявок в очереди, $p_{обс}$ — относительная пропускная способность (вероятность обслуживания поступившей заявки), λ — интенсивность поступающих заявок, μ — интенсивность одного канала.

9. Что означает и как определяется средний темп поступления заявок?

То же самое, что среднее число заявок в единицу времени – интенсивность λ.

10. Что означает и как определяется средняя длина очереди?

Математическое ожидание случайной величины Z — числа заявок в очереди $\mathrm{L}_{оч}$.

ПРИЛОЖЕНИЕ

Листинг кода

```
import numpy as np

lmbd = 4.46
mu = 2.7
r = 1
m = 2

# S0 ------> S1 ------> S2 ------> S3 ------> .... -----> S_{m}   (всего m+1 состояний)
#      lmbds[0]     lmbds[1]     lmbds[2]  ....         lmbds[m - 1]
#        lmbd         lmbd         lmbd                   lmbd
# S0 <------ S1 <------ S2 <------ S3 <------ .... <----- S_{m}   (всего m+1 состояний)
#       mus[0]        mus[1]       mus[2]  ....        mus[m - 1]
#         mu          2 mu         3 mu     ....       r * mu     r * mu

lmbds = []
mus = []

for i in range(m + 1):
    if i != m:
        lmbds.append(lmbd)
    else:
        lmbds.append(0.0)
    if i != 0:
        mus.append(min(r * mu, i * mu))
    else:
        mus.append(0.0)

print(lmbds)
print(mus)

A = np.zeros(shape=(m+1, m+1))

for i in range(m + 1):
    #for j in range(m + 1):
    if (i > 0):
        A[i, i - 1] = lmbds[i - 1]
    A[i, i] = -(lmbds[i] + mus[i])
    if (i < m):
        A[i, i + 1] = mus[i + 1]
```

```python
print(A)

# TODO: Теперь надо решить систему уравнений A = 0
# при этом p0 + p1 + ... + p_{m} = 1

# решим
As = [lmbds[0] / mus[1]]
for i in range(m - 1):
    As.append(As[i] * (lmbds[i + 1] / mus[i + 2]))
As = [1.0] + As

# Пусть у нас есть ответы (стационарная точка)
p = np.zeros(shape=(m + 1))

p_sum = np.sum(As, axis=0)
for i in range(len(p)):
    p[i] = As[i] / p_sum



from scipy.integrate import solve_ivp
def rhs(s, v):
    equations = []
    for i in range(m+1):
        eq = 0.0
        for j in range(m+1):
            eq += A[i, j] * v[j]
        equations.append(eq)
    return equations

res = solve_ivp(rhs, (0, 5), [1.0] + [0.0] * (m))

import matplotlib.pyplot as plt

# нестационарное решение (функции)
p_diff = np.zeros(shape=(m+1))
for i in range(len(res.y)):
    plt.plot(res.t, res.y[i], label="S" + str(i))
    p_diff[i] = res.y[i, -1]


print(p)
print(p_diff)

plt.legend()
plt.show()
```

Вывод программы

```
Интенсивности перехода из i-го состояния в (i+1)-ое:
[4.46, 4.46, 4.46, 0.0]
Интенсивности перехода из i-го состояния в (i-1)-ое:
[0.0, 2.7, 2.7, 2.7]
Система уравнений:
[[-4.46  2.7   0.    0.  ]
 [ 4.46 -7.16  2.7   0.  ]
 [ 0.    4.46 -7.16  2.7 ]
 [ 0.    0.    4.46 -2.7 ]]
Стационарное решение: [0.10113541 0.16706072 0.27595956 0.45584431]

Вероятность, что k машин заняты рубкой:
0 — 0.1011
1 — 0.8989
Вероятность, что все машины заняты и l брёвен находится в очереди:
0 — 0.2682
1 — 0.2760
2 — 0.4558
Среднее число занятых рубкой машин: 0.8989
Среднее число свободных от рубки машин: 0.1011
Коэффициент загрузки машин: 0.8989
Коэффициент простоя машин: 0.1011
Среднее число брёвен в очереди: 1.1876

Нестационарное решение: [0.10115555 0.16702711 0.2760063  0.45581104]
```