Лабораторна робота №2

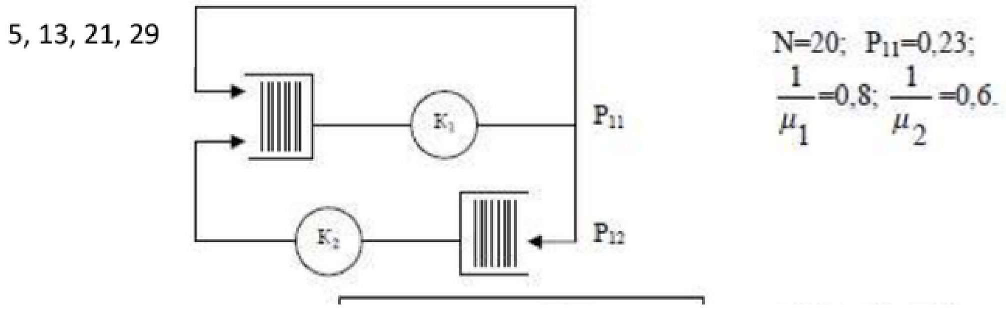
Дослідження мережі масового обслуговування аналітичними методами

Некряч Владислав, ТК-31

Варіант 13

Завдання до роботи

Дослідити обраний варіант мережі МО аналітичними методами і зробити висновки щодо ефективності її роботи.



Хід роботи

$$P_{11} = 0.23, P_{12} = 0.77$$

$$\mu_1 = \frac{5}{4}, \mu_2 = \frac{5}{3}$$

• **Коефіцієнти передачі**

$$e_1 = 1$$

$$e_2 = p_{12}e_1 + p_{22}e_2 = 0.77e_1$$

Розв’язок СЛАР:

$$\{e_1 = 1 \ \ e_2 = 0.77\}$$

• **Нормуючий множник**

Для випадку з 2 робочими станціями пошук нормуючого множнику зводиться до знаходження наступної суми:

$$C(N) = \left(\sum_{(k_1,k_2):\, k_1+k_2=N} p_1(k_1)p_2(k_2) \right)^{-1} = \left(\sum_{\alpha=0}^{N} p_1(\alpha)p_2(N-\alpha) \right)^{-1}.$$

$$C(N) \ = \ 36.64639369578203$$

- **Розрахунок $P_{CMO}(k)$**

$$p_{\text{CMO}_1}(j) = \sum_{(j, N-j)} p(j, N-j) = p(j, N-j) = C(N) p_1(j) p_2(N-j).$$

$P_{CMO_1}(k) \ = \ (where\ k\ is\ from\ 0\ to\ N)$

[
7.19234902e-06,
1.24542840e-05,
2.15658598e-05,
3.73434801e-05,
6.46640348e-05,
1.11972355e-04,
1.93891523e-04,
3.35742898e-04,
5.81372983e-04,
1.00670646e-03,
1.74321466e-03,
3.01855352e-03,
5.22693250e-03,
9.05096536e-03,
1.56726673e-02,
2.71388178e-02,
4.69936239e-02,
8.13742406e-02,
1.40907776e-01,
2.43996149e-01,
4.22504154e-01
]

$P_{CMO_2}(k) = \ (where\ k\ is\ from\ 0\ to\ N)$

[
0.4225041535815594,
0.2439961486933505,
0.1409077758704099,
0.08137424056516171,
0.04699362392638087,
0.027138817817484954,
0.01567266728959756,
0.009050965359742588,
0.005226932495251344,
0.0030185535160076513,
0.0017432146554944184,
0.0010067064635480266,
0.0005813729826989853,
0.0003357428975086639,
0.00019389152331125337,
0.0001119723547122488,
6.466403484632368e-05,
3.7343480123751925e-05,
2.1565859771466733e-05,
1.2454284018022035e-05,
7.1923490204077245e-06
]

Для кожної робочої станції сума коеф. $\mathbf{P_{CMO}(k)} = 1$, тобто розрахунки були проведені правильно.

<u>Обчислення основних показників ефективності функціонування розімкнутої ММО</u>

• **Пошук середньої кількості вимог у черзі СМО*ᵢ***

$$L_i = \sum_{j=r_i+1}^{N} (j - r_i) \cdot P_{CMOi}(j)$$

$$L_1 \approx 17.633, L_2 \approx, \ 0.789$$

• **Пошук середньої кількості зайнятих пристроїв у СМО*ᵢ***

$$R_i = r_i - \sum_{j=0}^{r_i-1} (r_i - j) P_{CMOi}(j)$$

$$R_1 \approx 0.9999928, R_2 \approx \ 0.577495$$

• **Пошук середньої кількості вимог, які перебувають у СМО*ᵢ***

$$M_i = L_i + R_i, i = \overline{1, n}$$

$$M_1 \approx 18.63334, M_2 \approx 1.36665$$

• **Пошук інтенсивності вихідного потоку СМО*ᵢ***

$$\lambda_i = R_i \mu_i$$

$$\lambda_1 \approx 1.24999 \ , \lambda_2 \approx 0.96249$$

• **Пошук середнього часу очікування в СМО*ᵢ***

$$Q_i = \frac{L_i}{\lambda_i}, i = \overline{1, n}$$

$$Q_1 \approx 14.90678, Q_2 \approx \ 1.41991$$

• **Пошук середнього часу перебування вимог у СМО*ᵢ***

$$T_i = \frac{M_i}{\lambda_i}, i = \overline{1, n}$$

$$T_1 \approx 14.10678, T_2 \approx 0.81991$$

• **Вивід програми та код**

```
e_i coefficients: [1.   0.77]
Normalizing factor: 36.64639369578203
Average queries in queue per CMO: [17.633349737231256, 0.7891616086993221]
Average working machines per CMO: [0.9999928076509796, 0.5774958464184405]
Average queries per CMO: [18.633342544882236, 1.3666574551177626]
Intensity of output streams: [1.2499910095637246, 0.9624930773640676]
Average query waiting time per CMO : [14.906781250679314, 1.4199140619905133]
Average query waiting time in queue per CMO: [14.106781250679314, 0.8199140619905133]
```

```python
import math

import numpy as np
from scipy import linalg

N = 20
mu_list: list[float] = [5/4, 5/3]
r_list: list[int] = [1, 1]

def get_e_list():
    coefficients = [
            [0.77, -1],
            [1, 0]
    ]
    result = [0, 1]
    return linalg.solve(coefficients, result)

def get_p_i(k, machine_number, e_list):
    machine_number -= 1
    conditional_factor = \
```

```
        1 / math.factorial(k) if k <=
r_list[machine_number] \
        else 1 / (math.factorial(r_list[machine_number])
* r_list[machine_number] ** (k-r_list[machine_number]))


    return (e_list[machine_number] /
mu_list[machine_number]) ** k * conditional_factor



def get_normalizing_factor(e_list):
    result = 0
    for alpha in range(N+1):
        result += get_p_i(k=alpha, machine_number=1,
e_list=e_list) * \
                    get_p_i(k=N-alpha, machine_number=2,
e_list=e_list)
    return 1 / result



def get_average_queries_in_queue(p_cmo_list):
    average_queries_in_queue = []

    for cmo in range(2):
        result = 0
        for j in range(r_list[cmo] + 1, N + 1):
            result += (j - r_list[cmo]) *
p_cmo_list[cmo][j]
        average_queries_in_queue.append(result)
    return average_queries_in_queue
```

```
def get_p_cmo_list():
    p_cmo_list = [np.zeros(N + 1), np.zeros(N + 1)]
    for j in range(N + 1):
        p_cmo_list[0][j] = normalizing_factor * \
                          get_p_i(k=j,
machine_number=1, e_list=e_list) * \
                          get_p_i(k=N - j,
machine_number=2, e_list=e_list)
    for j in range(N + 1):
        p_cmo_list[1][j] = normalizing_factor * \
                          get_p_i(k=N - j,
machine_number=1, e_list=e_list) * \
                          get_p_i(k=j,
machine_number=2, e_list=e_list)
    return p_cmo_list


def get_average_working_machines(p_cmo_list):
    average_working_machines = []
    for cmo in range(2):
        result = r_list[cmo]
        for j in range(r_list[cmo] + 1):
            result -= (r_list[cmo] - j) *
p_cmo_list[cmo][j]
        average_working_machines.append(result)
    return average_working_machines
```

```python
if __name__ == "__main__":
    e_list: list[float] = get_e_list()
    normalizing_factor = get_normalizing_factor(e_list)
    p_cmo_list = get_p_cmo_list()

    # for list in p_cmo_list:
    #     for el in list:
    #         print(el)
    #     print()
    # print(p_cmo_list)
    # print(sum(p_cmo_list[0]), sum(p_cmo_list[1]))

    average_queries_in_queue =
get_average_queries_in_queue(p_cmo_list=p_cmo_list)
    average_working_machines =
get_average_working_machines(p_cmo_list=p_cmo_list)
    average_queries = [
        queue_queries + working_machines
        for queue_queries, working_machines in
        zip(average_queries_in_queue,
average_working_machines)
    ]

    intensities = [
        mu * working_machines_cmo
        for mu, working_machines_cmo in
        zip(mu_list, average_working_machines)
    ]
```

```python
    average_cmo_wait_time = [
        average_queries_cmo / intensity
        for intensity, average_queries_cmo in
        zip(intensities, average_queries)
    ]

    average_cmo_queue_wait_time = [
        average_queries_in_queue_cmo / intensity
        for intensity, average_queries_in_queue_cmo in
        zip(intensities, average_queries_in_queue)
    ]


    print(f"e_i coefficients: {e_list}")
    print(f"Normalizing factor: {normalizing_factor}")

    print(f"Average queries in queue per CMO:
{average_queries_in_queue}")
    print(f"Average working machines per CMO:
{average_working_machines}")
    print(f"Average queries per CMO: {average_queries}")
    print(f"Intensity of output streams: {intensities}")
    print(f"Average query waiting time per CMO :
{average_cmo_wait_time}")
    print(f"Average query waiting time in queue per CMO:
{average_cmo_queue_wait_time}")
```

Теоретичні питання:

1) За яких припущень розглядаються аналітичні моделі мереж масового обслуговування?

Для аналітичного моделювання ММО роблять такі припущення:

1) ВВ «час обробки вимоги каналом обслуговування СМО» розподілена за експоненціальним законом з відомим параметром, який рівний інтенсивності обробки вимоги одним каналом СМО.
2) ВВ «час надходження вимоги в ММО» розподілена за експоненціальним законом з параметром, який рівний інтенсивності надходження вимог до ММО.
3) Усі черги необмежені.
4) Відомі ймовірності слідування вимоги з однієї СМО до іншої або до неї ж самої. Допускаються складні маршрути, не допускаються блокування маршрутів.

2) Які параметри задають розімкнуту мережу МО, замкнуту мережу МО?

Вхідні змінні розімкнутої ММО:

- n – кількість систем масового обслуговування ММО;
- λ_0 – інтенсивність надходження вимог до мережі МО;
- (p_{ij}) – матриця ймовірностей, p_{ij} – ймовірність переходу вимоги з $СМО_i$ у $СМО_j$;
- λ_i – інтенсивність вхідного потоку;
- r_i – кількість каналів обслуговування у $СМО_i$;
- μ_i – інтенсивність оброблених вимог кожним каналом $СМО_i$.

Для замкнутої:

- n – ***кількість СМО*** у ММО;
- N – ***кількість вимог*** у замкненій ММО;
- (p_{ij}) – ***матриця ймовірностей***, p_{ij} – ***ймовірність переходу*** вимоги з $СМО_i$ у $СМО_j$;
- r_i – ***кількість каналів*** обслуговування у $СМО_i$.;
- μ_i – ***інтенсивність обслуговування*** вимог кожним каналом $СМО_i$.

3) Як розраховується нормуючий множник у випадку розімкнутої мережі МО, у випадку замкнутої мережі МО? Як перевірити правильність розрахунку нормуючого множника?

Розімкнута ММО:

$$C_i = \left(\left(\frac{e_i\lambda_0}{\mu_i}\right)^{r_i} \cdot \frac{1}{r_i!\left(1-\frac{e_i\lambda_0}{\mu_i r_i}\right)} + \sum_{k=0}^{r_i-1}\left(\frac{e_i\lambda_0}{\mu_i}\right)^{k} \cdot \frac{1}{k!} \right)^{-1}$$

Перевірка - підставити у рівність:

$$C_i\left(\sum_{k=0}^{r_i}\left(\frac{\lambda_i}{\mu_i}\right)^k\frac{1}{k!} + \sum_{k=r_i+1}^{\infty}\left(\frac{\lambda_i}{\mu_i}\right)^k\frac{1}{r_i!\,r_i{}^{k-r}}\right) = 1.$$

Замкнута (перевірка + як знайти):

$$\sum_{(k_1,\dots,k_n)} C(N)\prod_{i=1}^{n} p_i(k_i) = 1,$$

$$C(N) = \left(\sum_{(k_1,\dots,k_n)}\prod_{i=1}^{n} p_i(k_i)\right)^{-1}.$$

4) Що таке коефіцієнти передачі? Як вони знаходяться у випадку розімкнутої мережі МО, у випадку замкнутої мережі МО?

Розімкнута МО:

Введемо коефіцієнт передачі e_i:

$$\lambda_i = e_i \lambda_0 \, , i = \overline{1, n}.$$

Це доля вхідного потоку λ_0 вимог, що надходять у СМО$_i$.

З іншого боку λ_i – це середня кількість раз проходження вимогою СМО$_i$.

Тоді для пошуку e_i отримаємо систему:

$$e_i \lambda_0 = \sum_{j=0}^{n} p_{ji} \lambda_j \, , \qquad e_i = p_{0i} + \sum_{j=1}^{n} p_{ji} e_j \, , i = \overline{1, n}.$$

Замкунта МО:

Коефіцієнт передачі e_i вводиться так:

$$\lambda_i = \mathrm{e}_i \lambda_1, i = \overline{2, n}.$$

Звідси зрозуміло, що $\mathrm{e}_1 = 1$, а e_i – середня кількість раз проходу вимоги через СМО$_i$, *i=2,3,…n* з моменту виходу цієї вимоги з СМО$_1$, і до мометнту повернення її у СМО$_1$.

Середня кількість вимог, що надходить до СМО$_i$, дорівнює середній кількості вимог, що залишають її, тому вхідний потік вимог у СМО$_i$ буде

$$\lambda_i = \sum_{j=1}^{n} p_{ji} \lambda_j \;\; , i = \overline{1, n}.$$

Враховуючи, що $\lambda_i = \mathrm{e}_i \lambda_1$ для e_i отримаємо

$$e_i \lambda_1 = \sum_{j=1}^{n} p_{ji} \lambda_j \;\; , i = \overline{1, n} \, ,$$

$$\begin{cases} e_i = \sum_{j=1}^{n} p_{ji} e_j \, , i = \overline{2, n} \\ \qquad e_1 = 1 \end{cases}$$

5) Як знаходяться ймовірності станів розімкнутої мережі МО, замкнутої мережі МО?

Ймовірність перебування у $СМО_i$ з необмеженою чергою *k* вимог:

$$p_i(k) = \left(\frac{e_i\lambda_0}{\mu_i}\right)^k \cdot C_i \cdot \begin{cases} \frac{1}{k!} & \text{при} \quad k \le r_i \\ \frac{1}{r_i! \cdot r_i^{k-r_i}} & \text{при} \quad k > r_i \end{cases}$$

$\boldsymbol{\lambda_i}$ – інтенсивність вхідного потоку;

$\boldsymbol{r_i}$ – кількість каналів обслуговування у $СМО_i$;

$\boldsymbol{\mu_i}$ – інтенсивність оброблених вимог кожним каналом $СМО_i$;

$\boldsymbol{C_i}$ – нормуючий множник.

У розімкнутій ММО всі СМО поводять себе незалежно одна від одної. Тому ймовірність того, що у $СМО_1$ k_1 вимог, і т.д., а у $СМО_n$ – k_n вимог буде

$$p(k_1, \dots, k_n) = \prod_{i=1}^{n} p_i(k_i)$$

У замкненій ММО окремі СМО залежні між собою, тому

$$p(k_1, \dots, k_n) \neq \prod_{i=1}^{n} p_i(k_i),$$

а

$$p(k_1, \dots, k_n) = C(N) \prod_{i=1}^{n} p_i(k_i).$$

Тут $C(N)$ – нормуючий множник, а функції $p_i(k)$ не є ймовірностями і можуть приймати довільне значення:

$$p_i(k) = \left(\frac{e_i}{\mu_i}\right)^k \cdot \begin{cases} \frac{1}{k!} \text{ при } k \le r_i \\ \frac{1}{r_i! \cdot r_i^{k-r_i}} \text{ при } k > r_i \end{cases}$$

$$P_{CMO_i}(k) = \boldsymbol{\Sigma}_{(k1,\, k2,\, \dots\, j,\, \dots,\, kn):\ k1+k2+\dots+kn+j\, =\, N}\ p(k_1, k_2, \dots\, j, \dots, k_n)$$

6) Які вихідні характеристики розімкнутої мережі МО? Як вони розраховуються?

1) середня ***кількість вимог у черзі*** СМО$_i$ знаходиться як математичне сподівання випадкової величини "кількість вимог у черзі"

$$L_i = \sum_{j=r_i+1}^{\infty} (j - r_i) p_i(j), \quad i = \overline{1, n}.$$

2) середня ***кількість зайнятих пристроїв*** у СМО$_i$ знаходиться як відношення інтенсивності надходження вимог до інтенсивності обслуговування вимог одним пристроєм

$$R_i = \frac{e_i \lambda_0}{\mu_i}.$$

3) середня ***кількість вимог у СМО$_i$*** – сума вимог у черзі і у пристроях

$$M_i = L_i + R_i$$

4) середній ***час очікування в черзі*** СМО$_i$ знаходять за ***2-ю формулою Літтла***

$$Q_i = \frac{L_i}{e_i \lambda_0} = \frac{L_i}{\lambda_i}.$$

5) середній ***час перебування вимог у СМО$_i$*** знаходиться за ***1-ю формулою Літтла***

$$T_i = \frac{M_i}{e_i \lambda_0} = \frac{M_i}{\lambda_i}.$$

---

6) середній ***час перебування вимог у ММО***

$$T = \sum_{i=1}^{n} e_i T_i,$$

e_i – коефіцієнт передачі, або кількість проходів вимогою СМО$_i$.

7) Які вихідні характеристики замкнутої мережі МО? Як вони розраховуються?

1) **середня кількість вимог** у черзі СМО$_i$ розраховується як математичне сподівання випадкової величини „кількість вимог у черзі”:

$$L_i = \sum_{j=r_i+1}^{N} (j - r_i) \cdot P_{CMOi}(j)$$

2) **середня кількість зайнятих пристроїв** у СМО$_i$ розраховується як загальна кількість пристроїв мінус середня кількість вільних пристроїв:

$$R_i = r_i - \sum_{j=0}^{r_i-1} (r_i - j) P_{CMOi}(j)$$

3) **середня кількість вимог** у СМО$_i$ розраховується як сума кількостівимог у черзі і у пристроях:

$$M_i = L_i + R_i$$

4) **інтенсивність вихідного потоку** вимог у СМО$_i$:

$$\lambda_i = R_i \mu_i$$

5) середній час перебування вимоги в СМО$_i$ розраховується за **другою формулою Літтла**:

$$T_i = \frac{M_i}{\lambda_i}$$

6) середній час очікування в черзі СМО$_i$ розраховується за **першою формулою Літтла**:

$$Q_i = \frac{L_i}{\lambda_i}$$

8) Як перевірити правильність розрахунків?

Якщо довжина черги значна, а кількість зайнятих пристроїв - ні, то допущена помилка.

9) Які висновки можна зробити на основі зроблених розрахунків вихідних характеристик мережі МО?

1. Які СМО перевантажені/недовантажені
2. Чи не перевантажена сама ММО
3. Де можна прибрати зайві потужності/додати нові потужності (або збільшили/зменшити потік вимог).

Висновки

Результати розрахунку відповідають розумному змісту моделі, є адекватними. Перший пристрій перевантажений, тому можна запропонувати збільшити кількість пристроїв в першій СМО.

Перевагою аналітичних методів дослідження ММО є простота отримання якісних результатів та відповідної інтерпретації ефективності роботи ММО.

Недоліком аналітичних методів дослідження ММО є використання середніх значень величин, що на практиці може не відповідати реальним результатам роботи ММО.