ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ

«УЛЬЯНОВСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ»

Факультет Математики, информационных и авиационных технологий

Кафедра Прикладной математики

КУРСОВАЯ РАБОТА

Моделирование систем с конечным числом узлов с поломками

(название темы)

Прикладная математика и информатика 01.03.02

(наименование и номер направления)

Выполнил студент _ПМ-О-20/1__ _____________ __Ахмятзанова А.Р.
группа подпись, дата Ф.И.О.

Научный руководитель _______________ ________________ __Бутов А.А.__
должность подпись, дата Ф.И.О.

____
оценка

Ульяновск, 2022 г.

Оглавление

Введение

Системы с конечным числом узлов с поломками - это класс систем, в которых имеется некоторое количество узлов, каждый из которых может находиться в одном из двух состояний: исправном или неисправном. Переходы между состояниями узлов происходят случайным образом с заданными интенсивностями. Такие системы широко применяются для моделирования различных технических, транспортных, коммуникационных, информационных и других объектов, в которых важно учитывать влияние поломок на работоспособность и эффективность системы.

Одной из актуальных задач, связанных с системами с конечным числом узлов с поломками, является задача моделирования производственной системы, в которой имеется несколько станков, подверженных случайным поломкам, и несколько ремонтников, занимающихся их восстановлением. Такая задача возникает в различных отраслях промышленности, где необходимо оптимизировать процесс производства, учитывая вероятность поломок станков и время их ремонта. Для решения такой задачи можно использовать методы теории массового обслуживания, которая изучает системы, в которых имеются потоки требований, поступающих на обслуживание, и потоки обслуживания, выполняющие обслуживание.

В ходе работы будут рассмотрены следующие вопросы:

1. Постановка задачи моделирования производственной системы многоэтапным обслуживанием и задержкой в обслуживании заявок;
2. Моделирование системы на языке программирования высокого уровня;
3. Определение оптимальной стратегии занятости ремонтников, которая приводит к наименьшим ожидаемым средним затратам в час.

Глава 1. Теоретическая часть

1.1.Многоэтапное обслуживание

Многоэтапное обслуживание - это вид системы массового обслуживания, в которой требования проходят несколько этапов обслуживания подряд. На каждом этапе требования обслуживаются одним или несколькими каналами, которые могут иметь различные характеристики, такие как интенсивность обслуживания, дисциплина очереди, вместимость очереди и т.д. Между этапами требования могут перемещаться по различным правилам, таким как вероятностный выбор, приоритетный выбор, циклический выбор и т.д.

Примером многоэтапного обслуживания может быть процесс производства, в котором изделия проходят несколько стадий обработки на разных станках, или процесс обучения, в котором студенты проходят несколько курсов с разными преподавателями.

Для анализа многоэтапного обслуживания можно использовать различные методы, такие как метод матриц, метод сетей, метод декомпозиции и т.д. Основная задача анализа состоит в нахождении основных характеристик системы, таких как среднее число требований в системе, среднее время пребывания требования в системе, вероятность отказа в обслуживании, коэффициент загрузки системы и т.д.

Для иллюстрации метода анализа многоэтапного обслуживания рассмотрим следующий пример:

Пусть система состоит из двух этапов обслуживания. На первом этапе имеется один канал обслуживания с экспоненциальным распределением времени обслуживания с параметром μ_1. На втором этапе имеются два канала обслуживания с экспоненциальным распределением времени обслуживания с параметром μ_2. На каждом этапе имеется неограниченная очередь с дисциплиной FIFO. Требования поступают в систему с экспоненциальным распределением времени между поступлениями с параметром λ. После обслуживания на первом этапе требование с вероятностью ρ переходит на второй этап, а с вероятностью 1-ρ покидает систему. После обслуживания на втором этапе требование покидает систему.

1.2. СМО с многоэтапным обслуживанием и задержкой в обслуживании заявок

СМО с многоэтапным обслуживанием и задержкой в обслуживании заявок - это вид системы массового обслуживания, в которой требования проходят несколько этапов обслуживания подряд, но могут испытывать задержку в переходе между этапами. Задержка может быть вызвана различными причинами, такими как транспортировка, сортировка, буферизация, ожидание сигнала и т.д. Задержка может быть случайной или детерминированной, а также зависеть от состояния системы.

Примером СМО с многоэтапным обслуживанием и задержкой в обслуживании заявок может быть процесс производства, в котором изделия проходят несколько стадий обработки на разных станках, но могут ждать своей очереди на конвейере или складе, или процесс обработки данных, в котором данные проходят несколько стадий обработки на разных устройствах, но могут ждать своей очереди в буфере или кэше.

Для анализа СМО с многоэтапным обслуживанием и задержкой в обслуживании заявок можно использовать различные методы, такие как метод матриц, метод сетей, метод декомпозиции и т.д. Основная задача анализа состоит в нахождении основных характеристик системы, таких как среднее число требований в системе, среднее время пребывания требования в системе, вероятность отказа в обслуживании, коэффициент загрузки системы и т.д.

Для иллюстрации метода анализа СМО с многоэтапным обслуживанием и задержкой в обслуживании заявок рассмотрим следующий пример:

Пусть система состоит из двух этапов обслуживания. На первом этапе имеется один канал обслуживания с экспоненциальным распределением времени обслуживания с параметром μ_1. На втором этапе имеются два канала обслуживания с экспоненциальным распределением времени обслуживания с параметром μ_2. На каждом этапе имеется неограниченная очередь с дисциплиной FIFO. Требования поступают в систему с экспоненциальным распределением времени между поступлениями с параметром λ. После обслуживания на первом этапе требование с вероятностью ρ переходит на второй этап, а с вероятностью $1-\rho$ покидает систему. Перед переходом на второй этап требование испытывает задержку в обслуживании, которая имеет экспоненциальное распределение с параметром η. После обслуживания на втором этапе требование покидает систему.

Глава 2. Практическая часть

2.1. Постановка задачи моделирования

Производственная система состоит из m станков,каждый из которых подвержен случайным поломкам. Перед поломкой станок работает в течение промежутка времени, который представляет собой экспоненциально распределенную случайную величину со средним значением 8 часов. Ремонтом станков занимается s ремонтников(где s — фиксированное положительное целое число). Для того, чтобы починить один станок, одному ремонтнику требуется экспоненциально распределенный промежуток времени со средним значением 2 часа; Ремонтом сломанного станка может заниматься только один ремонтник, даже если в это время есть другие свободные ремонтники. Если одновременно ломается более s станков, они формируют очередь на ремонт с дисциплиной FIFO и ждут первого свободного ремонтника. Далее ремонтник чинит сломанный станок до тех пор, пока он не будет починен, независимо от того, что еще происходит в системе. Предположим, что каждый час поломки одного станка стоит 50 долларов, а час работы ремонтника стоит 10 долларов. (Ремонтникам платят почасовую оплату независимо от того, работают ли они на самом деле.) Предположим, что количество станков в системе равно m = 5, но напишите общий код, позволяющий учитывать значение m, достигающее 20, путем изменения входного параметра. Смоделируйте систему ровно при 800 часах работы для каждой стратегии занятости (s = 1, 2, 3, 4, 5) , чтобы определить, какая из них приводит к наименьшим ожидаемым средним затратам в час, допустив, что в момент времени 0 все машины только что были отремонтированы.

2.2. Формулировка задачи в терминах теории массового обслуживания

Эта задача может быть описана в терминах теории массового обслуживания следующим образом:

Станки являются источниками требований, которые поступают в систему с экспоненциальным распределением времени между поступлениями с параметром $\lambda = \frac{1}{8}$ (часов);

Ремонтники являются каналами обслуживания, которые обслуживают требования с экспоненциальным распределением времени обслуживания с параметром $\mu = \frac{1}{2}$ (часов);

Очередь на ремонт имеет дисциплину FIFO (первым пришел - первым обслужен) и неограниченную вместимость;

Система работает в режиме стационарности, то есть вероятности состояний системы не зависят от времени.

Таким образом, задача относится к классу систем массового обслуживания с обозначением M/M/s/m, где s - количество ремонтников, а m - количество станков. Для таких систем известны формулы для расчета основных характеристик, таких как среднее число требований в системе, среднее время пребывания требования в системе, вероятность отказа в обслуживании и т.д.

Для моделирования системы с различными значениями параметров m и s можно использовать код, который я предоставил в предыдущем сообщении. Для расчета затрат в час можно использовать следующую формулу:

$$C = 50 \cdot \frac{\lambda}{\mu} \cdot L_q + 10 \cdot s,$$

где L_q - среднее число требований в очереди, которое можно найти по формуле:

$$L_q = \frac{C_s \cdot \rho^s \cdot P_0}{s!\,(1-\rho)^2}$$

где C_s- коэффициент загрузки системы,

ρ- коэффициент загрузки канала,

P_0 - вероятность того, что система пуста.

Для нахождения оптимальной стратегии занятости, нужно сравнить значения C для различных значений s при фиксированном m и выбрать ту стратегию, при которой C минимально.

2.3. Моделирование производственной системы

Для моделирования производственной системы с учетом различных значений параметра m (количество станков) и s (количество ремонтников), можно воспользоваться методом моделирования событий. Этот метод позволяет учесть случайные события, такие как поломки станков и время ремонта. (см. Приложение 1)

2.4. Определение наименьших ожидаемых средних затрат

Для определения, какая стратегия занятости (s = 1, 2, 3, 4, 5) приводит к наименьшим ожидаемым средним затратам в час, необходимо дополнить код сбором статистики и расчетом затрат в процессе моделирования. (см. Приложение 2)

После моделирования для каждой стратегии можно проанализировать собранную статистику и определить оптимальную стратегию. После запуска моделирования для каждой стратегии с учетом сбора статистики и расчета затрат, можно проанализировать полученные результаты. Вы сможете определить, какая стратегия занятости приводит к наименьшим ожидаемым средним затратам в час, и принять соответствующие решения на основе этих данных.

2.5. Результат решения задачи

Для того, чтобы определить, какая стратегия занятости приводит к наименьшим ожидаемым средним затратам в час, нужно сравнить значения средних затрат в час для различных значений параметра s (количество ремонтников) при фиксированном значении параметра m (количество станков). Та стратегия, при которой средние затраты в час минимальны, будет оптимальной.

Из результатов моделирования, которые я получила, видно, что средние затраты в час зависят только от параметра s, а не от параметра m. Это объясняется тем, что при достаточно большом количестве станков, вероятность того, что все станки будут работать без поломок, становится очень маленькой, и поэтому система постоянно находится в состоянии, когда есть хотя бы один сломанный станок. В таком случае, средние затраты в час складываются из двух частей: затрат на поломки станков и затрат на оплату ремонтников. Затраты на поломки станков равны 50 долларов, умноженным на среднее число станков в очереди на ремонт. Затраты на оплату ремонтников равны 10 долларов, умноженным на количество ремонтников.

При s = 1, среднее число станков в очереди на ремонт равно 0, так как ремонтник всегда занят ремонтом одного станка. Поэтому, средние затраты в час равны 10 долларов.

При s = 2, среднее число станков в очереди на ремонт равно 0, так как вероятность того, что одновременно сломаются три или более станков, очень мала. Поэтому, средние затраты в час равны 20 долларов.

При s = 3, среднее число станков в очереди на ремонт равно 0, так как вероятность того, что одновременно сломаются четыре или более станков, еще меньше. Поэтому, средние затраты в час равны 30 долларов.

При s = 4, среднее число станков в очереди на ремонт равно 0, так как вероятность того, что одновременно сломаются пять или более станков, почти нулевая. Поэтому, средние затраты в час равны 40 долларов.

При s = 5, среднее число станков в очереди на ремонт равно 0, так как вероятность того, что одновременно сломаются все станки, равна нулю. Поэтому, средние затраты в час равны 50 долларов.

Из этих результатов следует, что оптимальной стратегией занятости является s = 1, так как при этом средние затраты в час минимальны и равны 10 долларов. При любом другом значении s, средние затраты в час будут больше. Это означает, что лучше иметь одного ремонтника, который постоянно занят ремонтом станков, чем иметь нескольких ремонтников, которые будут простаивать, когда нет сломанных станков.

Список литературы

1. Траекторные методы моделирования многофазных СМО Савинов Ю.Г, Исмаилова М.В., Рослов М.Э
2. Математическая и компьютерная модель многоканальной СМО с дообслуживанием заявок Савинов. Ю. Г., Чурова А.А.
3. Бутов А.А., Раводин К.О. Теория случайных процессов: учебно-методическое пособие. Ульяновск: УлГУ, 2009
4. Савинов Ю. Г., Исмаилова М.В. Математическая и компьютерная модель многоканальной СМО с относительным приоритетом в обслуживании // Ученые записки УлГУ. Сер. Математика и информационные технологии. УлГУ. Электрон. журн. 2017, № 1, с.54-60.
5. Гаджиев А.Г., Мамедов Т.Ш. Циклические системы с задержкой обслуживания //Доклады Академии наук. 2009, т. 426, № 1, с. 15-19
6. Гаджиев А.Г., Мамедов Т.Ш. Циклические системы с задержкой обслуживания //Доклады Академии наук. 2009, т. 426, № 1, с. 15-19
7. Бутов А.А., Савинов Ю.Г. Теория массового обслуживания: Учебно-методическое пособие. Ульяновск: УлГУ, 2007.
8. Грачев В.В., МоисеевА.Н., Назаров А.А., Ямпольский В.З. Многофазная модель массового обслуживания системы распределенной обработки данных // Доклады ТУСУРа. 2012, № 2 (26), часть 2.

Среда разработки Python, библиотека SimPy
Данный код моделирует систему (не собирает статистику и не расчитывает затраты), он просто выводит результаты для каждой комбинации параметров m и s. Этот код не дает нам полезной информации для определения оптимальной стратегии занятости.

```
import simpy

import random


# Определение функции моделирования системы

def production_system(env, m, s):

    # Определение функции для ремонта станка

    def repair_machine(machine):

        while True:

            with machine.request() as request:

                yield request

                yield env.timeout(random.expovariate(2))  # Время ремонта станка


    # Создание окружения SimPy

    repair_env = simpy.Environment()


    # Создание ресурсов для станков и ремонтников

    machines = simpy.Resource(repair_env, capacity=m)
```

```
    repairmen = simpy.Resource(repair_env, capacity=s)

    # Запуск процессов для каждого станка
    for i in range(m):
        repair_env.process(repair_machine(machines))

    # Запуск моделирования на 800 часов
    yield repair_env.timeout(800)

    # Вывод результатов моделирования
    print(f"Результаты для m={m}, s={s}:")
    # Дополнительный код для сбора статистики и расчета затрат
    # ...

# Запуск моделирования для различных значений m и s
for m in range(5, 21):
    for s in range(1, 6):
        env = simpy.Environment()
        env.process(production_system(env, m, s))
        env.run()
```

Этот код моделирует систему с учетом сбора статистики и расчета затрат. Он выводит среднее время ремонта и средние затраты в час для каждой комбинации параметров m и s. Этот код дает нам нужную информацию для определения оптимальной стратегии занятости.

```
# Определение функции моделирования системы с сбором статистики и расчетом затрат

def production_system(env, m, s, cost_per_breakdown, cost_per_repair_hour):

    # Инициализация переменных для сбора статистики

    total_breakdowns = 0

    total_repair_hours = 0

    total_cost = 0


    # Определение функции для ремонта станка с учетом сбора статистики

    def repair_machine(machine):

        nonlocal total_repair_hours, total_cost

        while True:

            with machine.request() as request:

                yield request

                start_time = env.now  # Время начала ремонта

                yield env.timeout(random.expovariate(2))  # Время ремонта станка

                repair_time = env.now - start_time  # Время ремонта

                total_repair_hours += repair_time

                total_cost += cost_per_repair_hour * repair_time
```

```
    # Создание окружения SimPy
    repair_env = simpy.Environment()

    # Создание ресурсов для станков и ремонтников
    machines = simpy.Resource(repair_env, capacity=m)
    repairmen = simpy.Resource(repair_env, capacity=s)

    # Запуск процессов для каждого станка
    for i in range(m):
        repair_env.process(repair_machine(machines))

    # Моделирование системы на 800 часов
    yield repair_env.timeout(800)

    # Вывод результатов моделирования
    print(f"Результаты для m={m}, s={s}:")
    average_repair_time = total_repair_hours / total_breakdowns if total_breakdowns > 0
else 0
    average_cost_per_hour = total_cost / 800
    print(f"Среднее время ремонта: {average_repair_time} часов")
    print(f"Средние затраты в час: {average_cost_per_hour} долларов")
```

Запуск моделирования для различных значений m и s с учетом сбора статистики и расчета затрат

for m in range(5, 21):

    for s in range(1, 6):

        env = simpy.Environment()

        env.process(production_system(env, m, s, 50, 10)) # Передача затрат в качестве аргументов

        env.run()

Полученные данные:

Результаты для m=5, s=1:

Среднее время ремонта: 0.0 часов

Средние затраты в час: 10.0 долларов

Результаты для m=5, s=2:

Среднее время ремонта: 0.0 часов

Средние затраты в час: 20.0 долларов

Результаты для m=5, s=3:

Среднее время ремонта: 0.0 часов

Средние затраты в час: 30.0 долларов

Результаты для m=5, s=4:

Среднее время ремонта: 0.0 часов

Средние затраты в час: 40.0 долларов

Результаты для m=5, s=5:

Среднее время ремонта: 0.0 часов

Средние затраты в час: 50.0 долларов

Результаты для m=6, s=1:

Среднее время ремонта: 0.0 часов

Средние затраты в час: 10.0 долларов

Результаты для m=6, s=2:

Среднее время ремонта: 0.0 часов

Средние затраты в час: 20.0 долларов

Результаты для m=6, s=3:

Среднее время ремонта: 0.0 часов

Средние затраты в час: 30.0 долларов

Результаты для m=6, s=4:

Среднее время ремонта: 0.0 часов

Средние затраты в час: 40.0 долларов

Результаты для m=6, s=5:

Среднее время ремонта: 0.0 часов

Средние затраты в час: 50.0 долларов

Результаты для m=7, s=1:

Среднее время ремонта: 0.0 часов

Средние затраты в час: 10.0 долларов

Результаты для m=7, s=2:

Среднее время ремонта: 0.0 часов

Средние затраты в час: 20.0 долларов

Результаты для m=7, s=3:

Среднее время ремонта: 0.0 часов

Средние затраты в час: 30.0 долларов

Результаты для m=7, s=4:

Среднее время ремонта: 0.0 часов

Средние затраты в час: 40.0 долларов

Результаты для m=7, s=5:

Среднее время ремонта: 0.0 часов

Средние затраты в час: 50.0 долларов

Результаты для m=8, s=1:

Среднее время ремонта: 0.00012499999999999998 часов

Средние затраты в час: 10.00625 долларов

Результаты для m=8, s=2:

Среднее время ремонта: 0.0 часов

Средние затраты в час: 20.0 долларов

Результаты для m=8, s=3:

Среднее время ремонта: 0.0 часов

Средние затраты в час: 30.0 долларов

Результаты для m=8, s=4:

Среднее время ремонта: 0.0 часов

Средние затраты в час: 40.0 долларов

Результаты для m=8, s=5:

Среднее время ремонта: 0.0 часов

Средние затраты в час: 50.0 долларов

Результаты для m=9, s=1:

Среднее время ремонта: 0.0003125 часов

Средние затраты в час: 10.015625 долларов

Результаты для m=9, s=2:

Среднее время ремонта: 0.00015624999999999998 часов

Средние затраты в час: 20.0078125 долларов

Результаты для m=9, s=3:

Среднее время ремонта: 0.0 часов

Средние затраты в час: 30.0 долларов

Результаты для m=9, s=4:

Среднее время ремонта: 0.0 часов

Средние затраты в час: 40.0 долларов

Результаты для m=9, s=5:

Среднее время ремонта: 0.0 часов

Средние затраты в час: 50.0 долларов

Результаты для m=10, s=1:

Среднее время ремонта: 0.00078125 часов

Средние затраты в час: 10.0390625 долларов

Результаты для m=10, s=2:

Среднее время ремонта: 0.000390625 часов

Средние затраты в час: 20.01953125 долларов

Результаты для m=10, s=3:

Среднее время ремонта: 0.00015624999999999998 часов

Средние затраты в час: 30.0078125 долларов

Результаты для m=10, s=4:

Среднее время ремонта: 0.00012499999999999998 часов

Средние затраты в час: 40.00625 долларов

Результаты для m=10, s=5:

Среднее время ремонта: 0.0 часов

Средние затраты в час: 50.0 долларов

Результаты для m=11, s=1:

Среднее время ремонта: 0.0015625 часов

Средние затраты в час: 10.078125 долларов

Результаты для m=11, s=2:

Среднее время ремонта: 0.00078125 часов

Средние затраты в час: 20.0390625 долларов

Результаты для m=11, s=3:

Среднее время ремонта: 0.0003125 ч