Надёжность информационных систем. Лабораторная работа №1. Оценка и статистическое моделирование показателей безотказности нерезервированных систем

Макаров П. А.

Содержание

1 Введение **1**

2 Краткая теория **2**

3 Практическое задание **3**
3.1 Системный анализ и актуализация данных 4
3.2 Аналитический расчёт 4
3.3 Статистическое моделирование на Python 4

4 Требования к отчёту **6**

5 Контрольные вопросы **6**

Список источников **6**

1 Введение

В учебной литературе существует множество пособий и книг, посвящённых теоретическим аспектам надёжности информационных систем [1—4]. К ним, в частности, относится и достаточно неплохой конспект лекций Расуловой С. С. [5]. Его большими достоинствами являются компактность и ясность изложения, а также широта охвата материала. В то же время наблюдается значительный недостаток литературы, которую можно было бы использовать в качестве методического сопровождения при проведении лабораторных работ и практических занятий по курсу «Надёжность информационных систем». Это не значит, что подобных источников не существует в принципе! Данные книги, конечно же, имеются (см., например, [6—8]), однако все они написаны с поистине энциклопедическим масштабом, что само по себе замечательно, но сильно затрудняет их использование в реалиях очень ограниченного времени, отведённого на лабораторные занятия.

В связи с этим было принято компромиссное решение, состоящее в том, что предлагаемый студентам набор лабораторных работ (начинающийся с данной работы) опирается только на ту теоретическую базу, которая изложена в пособии [5]. Тем не менее,

надо понимать, что источник [5] не лишён и недостатков. Среди главных из них я бы выделил недостаточную актуальность приведённых фактологических и справочных данных[1] (что, конечно же, не являляется сколько-нибудь важной проблемой для учебного пособия, опубликованного порядка двадцати лет назад). Вместе с тем представляется весьма интересным обратить отмеченную выше особенность [5] в нашу пользу. Это тем более важно, что актуализация фактологических данных о надёжности современных компьютеров в форме самостоятельного исследования является хорошей базой именно для практической части работы студентов.

Дополнительно отметим, что изложение материала в [5] базируется на стандарте [9]. И хотя с точки зрения юридического статуса этот документ является действующим, следует учитывать, что с момента его разработки прошло уже порядка сорока лет. В связи с этим, необходимо иметь в виду существование более актуального на данный момент государственного стандарта [10].

Переходя к теме непосредственно данной лабораторной работы отметим, что при рассмотрении показателей надёжности невосстанавливаемых систем в качестве примеров в [5] используются типичные значения интенсивности отказов λ компонентов, характерных для вычислительной техники 1980 – 1990-х годов (например, перфолент, магнитных лент и т. п.). В то же время ни для кого не является секретом, что реализация современных информационных систем к настоящему моменту претерпела радикальные изменения. В частности, например, на смену механическим носителям информации пришли NVMe-накопители (Non-Volatile Memory Express); вместо классических устройств оперативной памяти используются микросхемы DDR5/LPDDR5 с ECC (Error-Correcting Code); а периферийные устройства, как правило, интегрируются в систему на уровне SoC (System-On-a-Chip).

Цель работы

1. Изучить базовые вероятностные показатели безотказности невосстанавливаемых информационных систем ($P(t)$, $Q(t)$, λ, $T_{\text{ср}}$).

2. Провести *верификацию и актуализацию* справочных данных из лекционного курса [5], заменив устаревшие компоненты на современные узлы серверной архитектуры.

3. Выполнить аналитический расчёт надёжности современного серверного узла.

4. Разработать программу на языке Python для статистического моделирования потока отказов и визуализации теоретических и эмпирических показателей надёжности.

2 Краткая теория

Как известно [5], надёжность невосстанавливаемого объекта оценивается вероятностными показателями, основными из которых являются следующие.

- **Вероятность безотказной работы (ВБР)** $P(t)$ — вероятность того, что объект не откажет к моменту времени t.

[1] Справедливости ради, подчеркнём, что эта проблема встречается почти во всех отмеченных выше источниках. В качестве единственного исключения здесь может рассматриваться разве что книга [8], да и то не в полной мере.

- **Вероятность отказа**
$$Q(t) = 1 - P(t). \tag{1}$$

- **Частота отказов** — это плотность распределения времени безотказной работы (производная от вероятности безотказной работы)
$$\nu(t) = \frac{\mathrm{d}Q(t)}{\mathrm{d}t} = -\frac{\mathrm{d}P(t)}{\mathrm{d}t}. \tag{2}$$

- **Интенсивность отказов** $\lambda(t)$ — условная плотность вероятности возникновения отказа
$$\lambda(t) = \frac{\nu(t)}{P(t)}. \tag{3}$$
Под *условностью* здесь подразумевается, что вероятность отказа оценивается **при обязательном условии:** к данному моменту времени объект ещё не вышел из строя (то есть проработал безотказно до времени t).

- **Средняя наработка до отказа** (MTTF — Mean Time To Failure)
$$T_{\text{ср}} = \int_0^\infty P(t)\,\mathrm{d}t. \tag{4}$$

Для электронных вычислительных систем в период нормальной эксплуатации (участок II U-образной кривой интенсивности отказов) справедливо **экспоненциальное распределение**, при котором $\lambda(t) = \lambda = \text{const}$. В этом случае основные показатели рассчитываются по формулам:
$$P(t) = e^{-\lambda t}, \quad Q(t) = 1 - e^{-\lambda t}, \quad T_{\text{ср}} = \frac{1}{\lambda}. \tag{5}$$

Для системы, состоящей из N элементов, соединённых последовательно (с точки зрения теории надёжности), результирующая интенсивность отказов равна:
$$\lambda_{\text{sys}} = \sum_{i=1}^{N} \lambda_i. \tag{6}$$

Проблема достоверности данных. В Таблице 1, приведённой в Лекции 4 [5], указаны значения λ для устаревших компонентов (например, $\lambda = 350 \cdot 10^{-6}$ 1/ч для памяти на магнитных лентах). Использование этих данных для расчёта современных ЦОД (центров обработки данных) может привести к колоссальным погрешностям и не даст верных оценок их надёжности. Студенту необходимо самостоятельно найти актуальные данные (например, из отчётов Backblaze, Google Datacenter Reliability или спецификаций производителей серверного оборудования) и пересчитать λ для современных комплектующих.

3 Практическое задание

Непосредственное задание данной лабораторной работы включает три элемента: системный анализ, аналитический расчёт и компьютерное моделирование.

3.1 Системный анализ и актуализация данных

Рассмотрите архитектуру современного вычислительного узла (сервера) для обработки больших данных. Замените устаревшие данные из Таблицы 1 в Лекции 4 [5] на современные, используя открытые источники (отчёты о надёжности HDD/SSD, спецификации RAM, CPU и т. п.). Заполните аналог данной таблицы в рамках вашего отчёта. Для каждого компонента укажите его современное наименование, количество в узле и актуальную интенсивность отказов λ_i (в 1/ч или FIT).

FIT (Failures In Time) — это отраслевой стандарт измерения интенсивности отказов электронных компонентов: 1 FIT $= 10^{-9}$ 1/ч (или 1 отказ на 1 миллиард часов суммарной наработки устройства). Эта единица используется вместо неудобных малых дробей в современных спецификациях производителей серверного оборудования (Intel, Samsung и др.) и отчётах о надёжности ЦОД.

Пример для поиска: интенсивность отказов современных enterprise NVMe SSD (например, PCIe Gen5) составляет порядка $10 \div 50$ FIT, а серверных CPU — около $20 \div 100$ FIT на кристалл.

3.2 Аналитический расчёт

Используя данные из сформированной вами таблицы, рассчитайте следующие величины.

1. Результирующую интенсивность отказов серверного узла λ_{sys}.
2. Вероятность безотказной работы узла $P(t)$ за 1 месяц (730 часов), 1 год (8 760 часов) и 3 года.
3. Среднюю наработку до отказа (MTTF) данного серверного узла в часах и годах.

3.3 Статистическое моделирование на Python

Напишите скрипт на Python, моделирующий эксплуатацию парка из $N_0 = 1\,000$ идентичных серверных узлов в течение $T_{\max} = 8\,760$ часов (1 год).

Требования к скрипту

- Для определения времени отказа каждого узла используйте генератор случайных чисел, подчиняющихся экспоненциальному закону распределения.
- Разбейте время моделирования на дискретные интервалы Δt (например, 100 часов).
- Рассчитайте статистические оценки ВБР $\hat{P}(t)$ и интенсивности отказов $\hat{\lambda}(t)$ по формулам из Лекции 2 [5].
- Постройте следующие графики:
  1. сопоставление теоретической $P(t)$ и статистической $\hat{P}(t)$ ВБР;
  2. гистограмма времени отказов в сравнении с теоретической функцией плотности распределения $\nu(t) = \lambda e^{-\lambda t}$.

Пример кода

```
import numpy as np
import matplotlib.pyplot as plt

# --- Исходные данные (подставляются после Этапа 3.2) ---
lambda_sys = 1.5e-5  # Примерное значение 1/ч (заменить на своё)
N0 = 1000            # Начальное количество серверов
T_max = 8760         # Время моделирования (1 год в часах)
dt = 100             # Шаг интервала времени для статистики

# 1. Генерация времени отказов (экспоненциальное распределение)
np.random.seed(42)
failure_times = np.random.exponential(scale=1/lambda_sys, size=N0)
# Отсекаем отказы, произошедшие позже T_max (для статистики наблюдений)
failure_times = failure_times[failure_times <= T_max]

# 2. Расчёт статистических оценок
t_bins = np.arange(0, T_max + dt, dt)
t_centers = (t_bins[:-1] + t_bins[1:]) / 2

# Гистограмма отказов
n_failures, _ = np.histogram(failure_times, bins=t_bins)

# N(t) - число исправных объектов концу каждого интервала
# Добавляем N0 в начало, чтобы получить массив длины len(t_bins)
N_t = N0 - np.concatenate(([0], np.cumsum(n_failures)))

# Статистическая вероятность безотказной работы P_hat(t)
P_hat = N_t[1:] / N0  # берём значения после каждого интервала

# Статистическая интенсивность отказов lambda_hat(t)
N_cp = (N_t[:-1] + N_t[1:]) / 2  # Среднее число исправных на интервале
lambda_hat = n_failures / (N_cp * dt)

# 3. Теоретические значения
P_theory = np.exp(-lambda_sys * t_centers)
f_theory = lambda_sys * np.exp(-lambda_sys * t_centers)

# 4. Построение графиков
fig, axs = plt.subplots(1, 2, figsize=(14, 5))

# График 1: ВБР
axs[0].plot(t_centers, P_theory, 'r-', label='Теоретическая $P(t)$')
axs[0].plot(t_centers, P_hat, 'bo', markersize=3, label='Статистическая
↪  $\\hat{P}(t)$')
axs[0].set_xlabel('Время $t$, часы')
axs[0].set_ylabel('Вероятность безотказной работы $P(t)$')
axs[0].set_title('Оценка ВБР серверного узла')
axs[0].legend()
axs[0].grid(True)
```

```
# График 2: Плотность распределения
axs[1].plot(t_centers, f_theory, 'r-', label='Теоретическая $\\nu(t)$')
axs[1].bar(t_centers, n_failures / (N0 * dt), width=dt*0.8, alpha=0.6,
           label='Статистическая плотность')
axs[1].set_xlabel('Время $t$, часы')
axs[1].set_ylabel('Плотность распределения отказов')
axs[1].set_title('Гистограмма времени наработки до отказа')
axs[1].legend()
axs[1].grid(True)

plt.tight_layout()
plt.savefig('reliability_simulation.png', dpi=300)
plt.show()
```

4 Требования к отчёту

Отчёт должен содержать:

1. Титульный лист, цель работы и краткие теоретические положения.
2. Таблицу с актуализированными данными по надёжности элементов архитектуры современного сервера (с указанием источников, откуда взяты значения λ или FIT).
3. Результаты аналитических расчётов с подробными выкладками.
4. Листинг программного кода с комментариями и подставленными рассчитанными значениями.
5. Графики, сгенерированные скриптом (с анализом расхождения теоретических и статистических кривых). Почему на ранних этапах времени статистика может "шуметь" и как это связано с величиной Δt)?
6. Выводы о применимости экспоненциальной модели для современных нерезервированных вычислительных систем.

5 Контрольные вопросы

1. Дайте определение понятиям «надёжность», «безотказность» и «наработка на отказ». В чём разница между восстанавливаемыми и невосстанавливаемыми системами?
2. Почему для электронных компонентов информационных систем в период нормальной эксплуатации принимается допущение $\lambda(t) = \text{const}$?
3. Как связаны между собой вероятность безотказной работы $P(t)$ и интенсивность отказов λ при экспоненциальном законе распределения?
4. В чём заключается метод статистической оценки интенсивности отказов $\hat{\lambda}(t)$? Почему в знаменателе формулы (11) из Лекции 2 в [5] используется $N_{\text{ср}}$, а не N_0?
5. Почему справочные данные 1980-х годов (Таблица 1 в Лекции 4 из [5]) некорректно применять для расчёта надёжности современных облачных и серверных систем без верификации?

Список источников

1. *Матуско В. Н.*, *Лебедев Н. С.* Надёжность информационных систем. — Новосибирск : Сибирская государственная геодезическая академия, 2006. — С. 129.
2. *Ващенко Г. В.*, *Добронец Б. С.* Надёжность информационных систем. — Красноярск : Сибирский федеральный университет, 2008. — С. 208.
3. Надёжность информационных систем / Ю. Ю. Громов [и др.]. — Тамбов : Тамбовский государственный технический университет, 2010. — С. 160.
4. *Гнеденко Б. В.*, *Беляев Ю. К.*, *Соловьев А. Д.* Математические методы в теории надёжности: Основные характеристики надёжности и их статистический анализ. — Изд. 3, стереотип. — Москва : URSS, 2024. — С. 584.
5. *Расулова С. С.* Надёжность информационных систем. — Ташкент : Ташкентский университет информационных технологий, 2007. — С. 118.
6. *Jain R.* The Art of Computer Systems Performance Analysis: Techniques for Experimental Design, Measurement, Simulation, and Modeling. — John Wiley & Sons, 1991. — P. 720.
7. *Modarres M.*, *Kaminskiy M. P.*, *Krivtsov V.* Reliability Engineering and Risk Analysis: A Practical Guide. — 3rd Edition. — Boca Raton : CRC Press, 2016. — P. 522.
8. *Breneman J. E.*, *Sahay C.*, *Lewis E. E.* Introduction to Reliability Engineering. — 3rd Edition. — John Wiley & Sons, 2022. — P. 640.
9. ГОСТ 24.701-86. Единая система стандартов автоматизированных систем управления. Надёжность автоматизированных систем управления. Основные положения. — Москва, 2009.
10. ГОСТ Р 27.102-2021. Надёжность в технике. Надёжность объекта. Термины и определения. — Москва, 2021.

Актуальные отраслевые отчёты

- Backblaze Drive Stats Data (Ежеквартальные отчёты о надёжности HDD/SSD).
- Google Datacenter Reliability Reports (Данные о FIT для серверных CPU, RAM и NVMe).
- Спецификации производителей (например, Intel Xeon Scalable Processors Datasheet, Samsung Enterprise SSD MTBF[2] specs и т. п.).

[2]Здесь MTBF (Mean Time Between Failures) — это «наработка на отказ», термин применяемый для характеристики надёжности восстанавливаемых систем.