Министерство образования и науки Российской Федерации
Федеральное государственное автономное образовательное учреждение
высшего профессионального образования

Санкт-Петербургский национальный исследовательский университет

информационных технологий, механики и оптики

Факультет Компьютерных технологий и управления
Кафедра ПБКС

Конспект по дисциплине

Защищённые информационные системы

XeTeX

Санкт-Петербург, 2016

Термины и определения

Функциональные качества технических устройств, в.т.ч. информационных систем, безопасность данных систем в большой степени зависит от их надёжности. Под ИС мы будем понимать сложную программно-аппаратную систему, включающую в свой состав эргатические звенья, технические средства и ПО.

Говоря о надёжности информационных систем следует учитывать две основные составляющие - надёжность аппаратных средств и надёжность ПО.

Теория надёжности опирается на перечень различных ГОСТов. Основной ГОСТ 27002-89.

- Под объектом по теории надёжности подразумевается техническое изделие определённого назначения, рассматриваемое в периоды проектирования, производства , испытания и эксплуатации. Объектами также могу быть системы и их элементы.
- Под системой подразумевается объект, представляющий собой совокупность элементов, связанных между собой определёнными отношениями, и взаимодействующих таким образом, чтобы обеспечить выполнение системой некоторых достаточно сложных функций.

С точки зрения надёжности выделяют 4 состояния объекта

1. Исправность - состояние объекта, при котором он соответствует всем требованиям, установленным в нормативно-технической документации (обычное состояние защищённой ИС).
2. Неисправность - состояние объекта, при котором он не соответствует хотябы одному из требований, установленных нормативно-технической документации (ЗИС - угроза безопасности)
3. Работоспособность - состояние объекта, при котором он способен выполнять заданные функции, сохраняя значения основных параметров, установленных в нормативно-технической документации
4. Неработоспособность - состояние объекта, при котором значение хотя-бы одного из параметров, характеризующего способность исполнять заданные функции не соответствует требованиям, установленным в нормативно-технической документации (DoS для ЗИС)

С точки зрения теории надёжности различают 6 различных переходов объекта в заданные состояния:

1. Повреждение - событие, заключающееся в нарушении исправности объекта при сохранении его работоспособности
2. Отказ - событие, заключающееся в нарушении работоспособности объекта
3. Критерий отказа - отличительный признак или совокупность признаков, согласно которым устанавливается факт отказа
4. Восстановление - процесс обнаружения и устранения отказа с целью восстановления объектом его работоспособности

Восстанавливаемый обьект - объект, работоспособность которого после отказа подлежит восстановлению в заданных условиях

Невосстанавливаемый - объект, работоспособность которого после отказа не подлежит восстановлению в заданных условиях

Рассмотрим следующие временные характеристики:

- Наработка - Продолжительность работы объекта. Объект может работать как непрерывно, так и в временными интервалами. Во втором случае будет учитываться суммарная наработка.
- Технический ресурс - наработка объекта от начала его эксплуатации до достижения предельного состояния.
- Срок службы объекта - календарная продолжительность эксплуатации объекта от её начала или возобновления после ремонта до наступления предельного состояния.
- Эксплуатация объекта - стадия его существования в распоряжении потребителя при условии применения объекта по назначению, что может чередоваться с хранением, транспортировкой, техобслуживанием и ремонтом, если это осуществляется потребителем.

Надёжность - (по ГОСТ 27002) - свойство объекта сохранять во времени в установленных пределах значения всех параметров, характеризующих способность выполнять требуемые функции в заданных режимах и условиях применения.

С точки зрения ИБ надёжность представляет собой способность ИС противостоять внешним или внутренним угрозам ИБ.

Факторы, определяющие надёжность ИС

Для построения ИС используются различные типы обеспечения: экономическое, временное, организационное, структурное, технологическое, эксплуатационное, социальное, эргатическое, алгоритмическое, синтаксическое и семантическое.

Под обеспечением можно характеризовать совокупность факторов, способствующих достижению заданной цели.

Организационное, временное и экономическое обеспечение, обуславливаемое необходимостью материальных и временных затрат используется для поддержания достоверности результатов работы ИС

Структурное обеспечение ИБ должно обеспечивать надёжность функционирования комплексов и эргатических звеньев, а также ИС в целом. Здесь обосновывается рациональное построение ИС, её структуры, зависящее от выбора структуры техпроцесса преобразования информации, обеспечения взаимосвязи между отдельными элементами системы, резервированию и использованию устройств, осуществляющих процедуры контроля.

Надёжность и технологическое обеспечения связана с выбором для конструктивных решений отдельных комплексов, входящих в состав системы, технологий и протоколов реализации информационных процессов.

Эргатичесекое обеспечение включает комплекс фактов, связанных с рациональной организацией работы человека в системе - правильное расположение функций между людьми и технологическими устройствами.

Надёжность алгоритмического обеспечения связана с обеспечением высокого качества и безошибочности алгоритмов и программ преобразования информации и реализации контроля достоверности информации.

Информационное. синтаксическое и семантическое обеспечение должно обеспечить специальную информационную избыточность, избыточность данных и смысловую избыточность, обуславливающей возможность поведения контроля достоверности информации.

(hint: http://sdo2.irgups.ru/course/view.php?id=69)

Виды ошибок:

- 1
- Ошибки совместимости (с ОС)
- Ошибки сопряжения

Основные показатели надёжности ПО

Если рассматривать отказавшее программное обеспечение без учёта его восстановления, а также случайный характер отказов - то модель надёжности будет принимать вид невосстанавливаемой информационной системы и, следовательно, основными показателями будут следующие величины:

- P(t) - вероятность, что ошибки программы не проявятся в интервале (0;t)
- Вероятность события отказа ПО - q(t) - вероятность, что ошибки программы проявятся в интервале (0;t)
- Интенсивность отказа $\lambda(t)$
- Время наработки на отказ T

При определении характеристик надёжности ПО учитывается тот факт, что возникающие при работе программ ошибки устраняются, количество ошибок уменьшается $\Rightarrow$ интенсивность отказов уменьшается, и наработка на отказ должна увеличиваться.

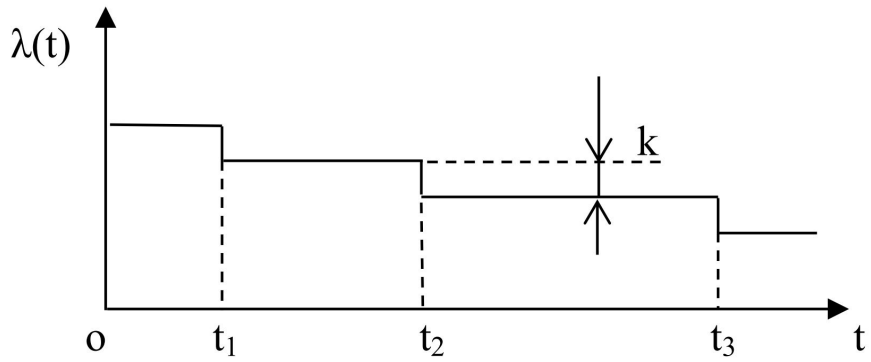
В связи с такими предположениями рассматривается несколько моделей надежности про- граммного обеспечения:

Модель с дискретно понижающейся частотой ошибок ПО

(ref: http://sdo2.irgups.ru/pluginfile.php/41173/mod_resource/content/1/Лекция%20№%2011.pdf)

В этой модели полагается, что интенсивность отказов $\lambda(t)$ является постоянной величиной до обнаружения возникшей ошибки. После этого значение интенсивности уменьшается, и данная величина становится опять постоянной.

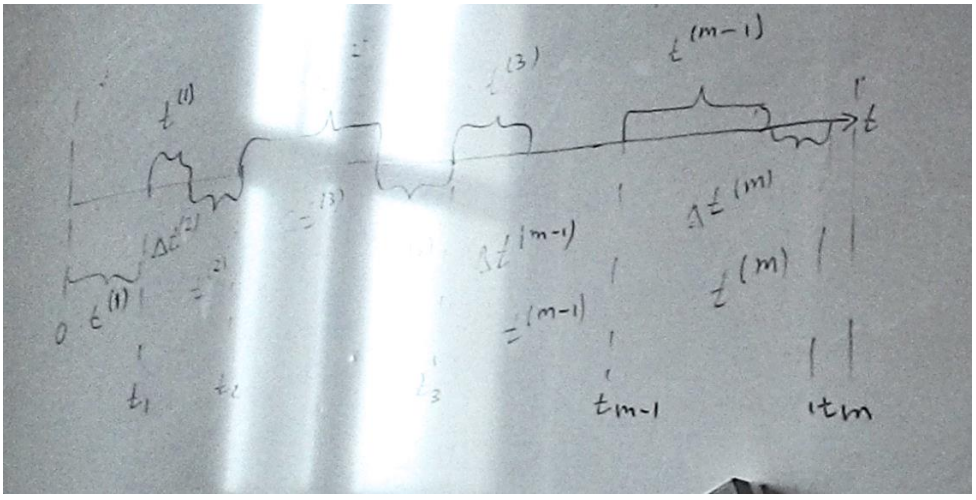
В данной модели интенсивность отказов $\lambda(t)$ можно выразить формулой $\lambda(t) = k(M - i) = \lambda$, где M - первоначальное число ошибок; i - число обнаруженных ошибок, зависящее от времени t; k - некоторая константа.



Плотность распределения времени обнаружения i - й ошибки в момент времени t_i определяется соотношением $f(t_i) = \lambda_i e^{-\lambda_i t_i}$, а параметры k и M будут устанавливаться на основе наблюдения интервалов между ошибками.

На практике же условия данной модели не соблюдаются, так как при устранении ошибок интенсивность отказов уменьшается на одну и ту же величину k, но разные ошибки имеют разный вес. Довольно часто возникают ситуации, когда исправление старых ошибок вызывает новые ошибки. Не всегда удаётся устранить причину ошибки, и ПО продолжают использовать, так как при других исходных данных ошибка может себя и не проявлять.

Модель с дискретным увеличением времени наработки на отказ



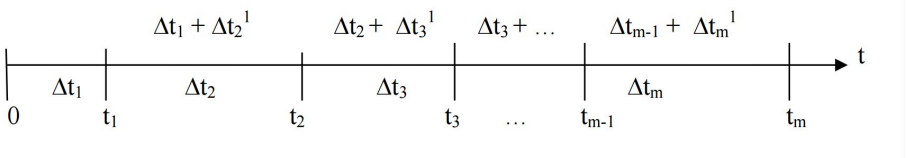


Рисунок показывает временные интервалы наработки на отказ. Величины t_1,t_2,t_3... t_m - случайные моменты возникновения первого, второго, третьего и так далее – m-го отказов Величины $t^{(1)}$,$t^{(2)}$,$t^{(3)}$... $t^{(m)}$ - случайные интервалы времени между возникновением соседних отказов.

Пусть первая ошибка, появившаяся при работе программы, происходит в момент времени t_1 и была устранена. Наработка до первого отказа и возникшей ошибки равна интервалу времени $t^{(1)}$, так как после перезапуска системы у нас возникает вторая ошибка через интервал времени t_2, с наработкой системы на отказ, равной $t^{(2)}$. В соответствии с предположением, этот интервал больше, чем Δt_1, так как после перезапуска программа проработала время до возникновения первой устраненной ошибки, а затем продолжила работу до новой второй ошибки.

Тогда $t^{(2)} = t^{(1)} + \Delta t^{(2)}$, где $\Delta t^{(2)}$ - дополнение до $\delta t^{(1)}$.

Случайное время возникновения ошибки $i-1$ в интервал времени t_i всегда отсчитывается с момента времени $t = 0$. Время на ликвидацию ошибки в расчёт не берётся. В этом случае для всех случайных моментов времени возникновения ошибки и временных интервалов между соседними ошибками можно записать:

$$
\begin{aligned}
t_1 &= t^{(1)} \\
t_2 &= t^{(1)} + t^{(1)} + \Delta t^{(2)} \\
t_2 &= t^{(1)} + t^{(1)} + \Delta t^{(2)} + t^{(1)} + \Delta t^{(2)} + \Delta t^{(3)} \\
&\dots \\
t_m &= m \cdot t^{(1)} + (m-1) \cdot \Delta t^{(2)} + (m-2)\Delta t^{(3)} + ... + 2\Delta t^{(m-1)} + \Delta t^{(m)}
\end{aligned}
$$

Учитывая, что от момента t_0 до момента t_1 не выявлено ни одной ошибки и что интервал t_1 сравнительно невелик, так как ошибки программы в начале эксплуатации происходят довольно часто можно представить интервал наработки на отказ как δt_i

Как видим, с последующим запуском программы после обнаружения и устранения ошибки временной интервал между соседними отказами постоянно увеличивается. Следовательно, увеличивается средняя наработка на отказ. Величину наработки на отказ программы можно оценить как:

$$t_{\text{ср}} = \frac{\sum\limits_{i-1}^{m} t^{(i)}}{m}$$

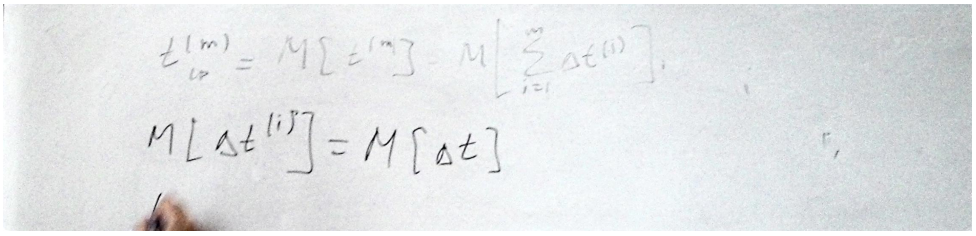
Теперь рассмотрим значения Δt_i $t^{(2)}$

Естественно, для любого i большего (мудак, бля) можем записать

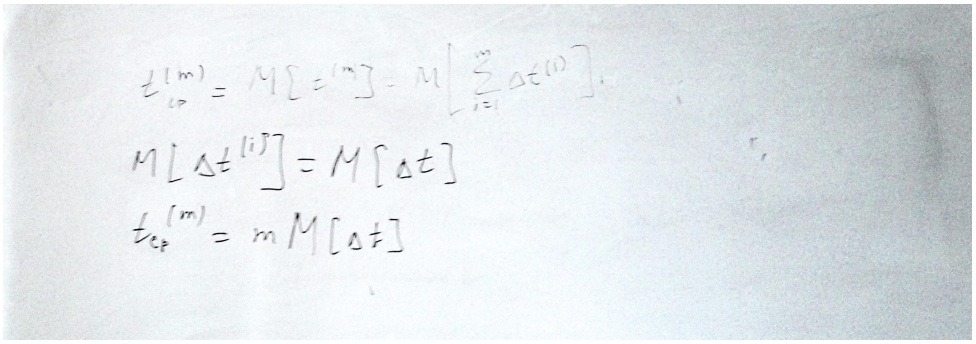
$$t^{(m)} = \sum_{i-1}^{m} \Delta t^{(i)}$$

Можем заметить. что $\Delta t^{(i)}$ равна матожиданию t_m, а $t^{(m)} = M[t^{(m)}]$

Но для любого i это матожидание равно

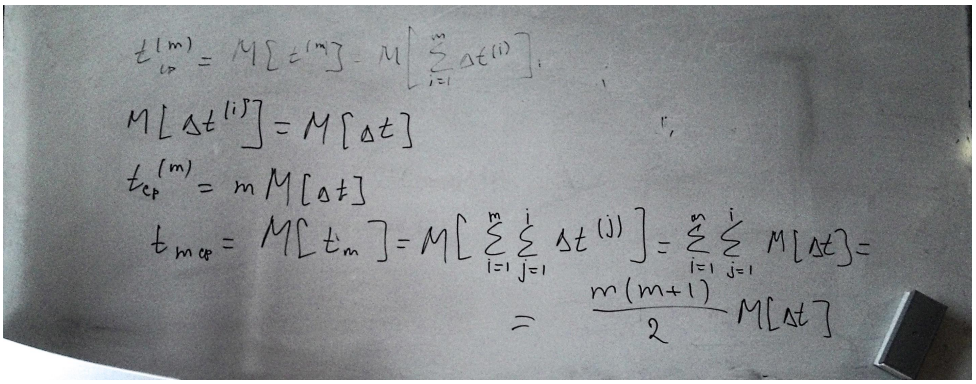


Приводя это упрощение, можем выразить среднее время наработки на отказ

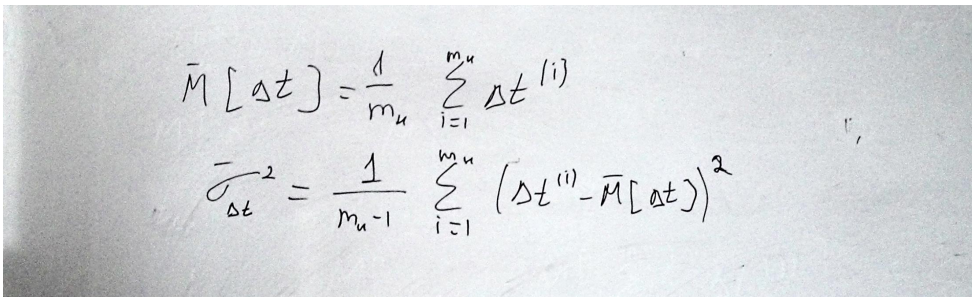


То-же самое мы можем провести с атаками на информационные системы.

Отсюда видно, что с увеличением числа ошибок увеличивается и средняя наработка между двумя отказами. Рассмотрим среднюю наработку до возникновения m-го отказа $t_{\text{ср}}^{(m)} =$



Как и в предыдущем случае здесь видно, что средняя наработка до отказа возрастает с увеличением числа отказов. Оценки матожидания и дисперсии для данных величин выглядят следующим образом:



$\sigma_{\Delta t}^2 = \ldots$ Где M_n - это число отказов за интервал времени от 0 до M.