

МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ
федеральное государственное бюджетное образовательное учреждение
высшего образования
«Тольяттинский государственный университет»

Институт математики, физики и информационных технологий

(наименование института полностью)

Кафедра «Прикладная математика и информатика»

(наименование кафедры)

01.04.02 Прикладная математика и информатика

(код и наименование направления подготовки)

Математическое моделирование

(направленность (профиль))

МАГИСТЕРСКАЯ ДИССЕРТАЦИЯ

на тему Математическая модель оценки надёжности веб-сайтов

Студент

Т.С. Касимова

(И.О. Фамилия)

(личная подпись)

Научный
руководитель

О.М. Гущина

(И.О. Фамилия)

(личная подпись)

Руководитель программы

д.ф.-м.н., доцент, С.В. Талалов

(ученая степень, звание, И.О. Фамилия)

(личная подпись)

« _____ »

20 _____ г.

Допустить к защите

Заведующий кафедрой к.т.н., доцент, А.В. Очеповский

(ученая степень, звание, И.О. Фамилия)

(личная подпись)

« _____ »

20 _____ г.

Тольятти 2018

ОГЛАВЛЕНИЕ

ВВЕДЕНИЕ	3
Глава 1 СРАВНЕНИЕ МЕТОДОВ ОЦЕНКИ НАДЁЖНОСТИ ВЕБ-САЙТА	7
1.1 Критерии надёжности веб-сайта.....	7
1.2 Методы анализа надёжности веб-сайта	20
1.3 Анализ видов и последствий и критичности отказов (АВПКО).....	22
1.4 Анализ деревьев неисправностей	32
1.5 Анализ схем функциональной целостности (СФЦ).....	41
1.6 Вейбулл-анализ (анализ эмпирических данных из испытаний и эксплуатации)	43
Глава 2 МАТЕМАТИЧЕСКАЯ МОДЕЛЬ ОЦЕНКИ НАДЁЖНОСТИ ВЕБ-САЙТОВ	50
2.1 Постановка задачи для модели оценки надёжности веб-сайтов.....	50
2.2 Составление математической модели оценки надёжности веб-сайтов	52
Глава 3 РЕАЛИЗАЦИЯ И ПРИМЕНЕНИЕ МОДЕЛИ ОЦЕНКИ НАДЁЖНОСТИ	56
3.1 Анализ программного обеспечения для реализации математической модели.....	56
3.2 Анализ инструментальной среды «Maple»	59
3.3 Анализ инструментальной среды «Mathematica»	61
3.4 Анализ инструментальной среды «MathCad»	65
3.5 Сравнение средств для реализации математической модели	72
3.6 Реализация математической модели оценки надёжности веб-сайтов..	75
3.7 Доказательство гипотезы о возможности построения математической модели поиска численного значения вероятности отказа в работе сайта.....	83
ЗАКЛЮЧЕНИЕ	86
СПИСОК ИСПОЛЬЗУЕМОЙ ЛИТЕРАТУРЫ.....	88

ВВЕДЕНИЕ

В настоящее время веб-сайт является неотъемлемой частью имиджа практически любой компании, а также необходимым инструментом ведения бизнеса для различного рода интернет-компаний. Для успешного функционирования веб-сайта необходимо четкое понимание его возможностей, чтобы свести до минимума вероятность отказа в обслуживании данного сайта. В связи с этим для сайта как системы, которая рассчитана на постоянную бесперебойную работу, надёжность играет серьёзную роль.

Под надёжностью сайта будем понимать его эксплуатационное качество, которое характеризуется вероятностью безотказной работы сайта в определенный период времени с сохранением параметров, заданных в техническом задании на сайт.

Одним из основных показателей надёжности сайта является вероятность безотказной работы. И, чтобы уменьшить вероятность отказа в обслуживании сайта, следует знать численное значение данной характеристики.

Таким образом, **актуальность работы** заключается в том, что в настоящее время создается достаточно много сайтов и, чтобы избежать возможных рисков, связанных с тем, что фактическое число пользователей превысило ожидаемое, уже на этапе проектирования сайта следует оценить его надёжность.

В настоящий момент существует достаточно много методов для оценки надёжности сайта, наиболее распространенными из которых являются:

- анализ видов, последствий и критичности отказов;
- анализ деревьев неисправностей;
- вейбулл-анализ;
- анализ схем функциональной целостности.

Анализ перечисленных выше методов показал, что ни один из перечисленных выше методов не дает способа получения численного значения

вероятности отказа в работе сайта. Это определяет **проблему исследования**. Из данной проблемы следует цель работы.

Целью работы является разработка математической модели оценки надёжности веб-сайтов.

Обозначенная цель определила объект и предмет исследования. **Объект исследования** – оценка надёжности веб-сайта.

Предмет исследования – модель оценки надёжности.

Гипотезой диссертации является предположение о том, что можно получить численное значение вероятности отказа в работе сайта на этапе его проектирования, если будет построена математическая модель.

Для достижения поставленной цели потребовалось решить следующие **задачи**:

1. Проанализировать методы оценки надёжности веб-сайтов.
2. Разработать модель оценки надёжности веб-сайтов.
3. Провести апробацию модели оценки надёжности веб-сайтов.
4. Найти численное значение вероятности безотказной работы веб-сайта.
5. Применить данную модель для нахождения максимально возможный поток посетителей веб-сайта при заданном времени задержки пользователя на странице и максимально возможным числом одновременных посетителей веб-сайта.
6. Доказать гипотезу о том, что можно получить численное значение вероятности отказа в работе сайта на этапе его проектирования, если можно построить математическую модель.

Теоретической основой исследований послужили научные труды в области оценки надёжности зарубежных и отечественных ученых, таких как Hyun K. C., Ikram M. A., Hussain F. K., Robertazzi T. G., Sekar K. R., Sethuraman J., Александровской Л., Афанасьева Т. А., Гусейнова Р. В., Дормидонтовой Т. В., Кирьякова В. В., Султановой Л. М., Канн С. К., Карманова Ф. И., Острейковского В. А., Линдигрина А. Н., Розенберга В. Я., Прохорова А. И.

Яковлева Б. С., Бойченко О. В., Тупота Е. С. и т.д.

Методы исследования. Решение указанных задач планируется проводить на основе системного анализа, математического моделирования систем массового обслуживания, объектно-ориентированного программирования. Для программной реализации алгоритмов планируется использовать аппарат численного математического моделирования и пакеты прикладных программ компьютерной математики.

На защиту выносятся:

1. Математическая модель оценки надёжности веб-сайтов.
2. Результаты эксперимента, проведённого с использованием данной модели.
3. Результаты апробации предложенной модели.

Исследование состояло из следующих этапов:

1. Изучение литературы по теме исследования, а именно анализ методов оценки надёжности веб-сайтов. Определение состояния проблемы на данный момент.
2. Построение математической модели оценки надёжности веб-сайтов с использованием теории массового обслуживания.
3. Проведения эксперимента для нахождения численного значения вероятности безотказной работы веб-сайта.

Апробация работы была произведена на следующих конференциях, где были рассмотрены результаты исследования:

1. IV Международная научно-практической конференция (школа-семинар) молодых ученых «Прикладная математика и информатика: современные исследования в области естественных и технических наук» (23-25 апреля 2018г., Тольятти).
2. Международная научно-практическая конференция «Взаимодействие науки и общества: проблемы и перспективы», стр. 7-9 (21 апреля 2018г, Новосибирск).
3. Научно-практическая конференция «Студенческие дни науки в

ТГУ» (апрель 2018г., Тольятти).

Новизна исследования состоит в том, что была построена модель, которая не только позволит оценить надёжность веб-сайта, но и получить численное значение вероятности отказа в обслуживании.

Теоретическая значимость исследования:

1. Расширено знание о нахождении численного значения вероятности безотказной работы веб-сайта.
2. Проведен анализ методов оценки надёжности применительно к веб-сайтам.

Практическая значимость исследования состоит в том, что полученные результаты позволят оценить надёжность проектируемого веб-сайта и принять верные решения для его развития.

На защиту выносятся математическая модель оценки надёжности веб-сайта, а также эксперимент, проведённый с целью нахождения численного значения вероятности безотказной работы веб-сайта.

Работа состоит из трех глав:

1. В первой главе описываются и сравниваются основные методы анализа надёжности веб-сайтов.
2. Во второй главе, в соответствие с гипотезой, строится математическая модель оценки надёжности веб-сайтов.
3. В третьей главе проводится анализ программного обеспечения для построения модели, проводится эксперимент над данной математической моделью для расчета количественного значения вероятности безотказной работы сайта, а также приводится доказательство гипотезы диссертационного исследования.

Диссертационное исследование состоит из введения, трех глав, заключения и списка используемой литературы. Работа изложена на 94 страницах, содержит 9 рисунков, 6 таблиц.

Глава 1 СРАВНЕНИЕ МЕТОДОВ ОЦЕНКИ НАДЁЖНОСТИ ВЕБ-САЙТА

1.1 Критерии надёжности веб-сайта

Под надёжностью Брумштейн Ю. М. и Бондарев А. А. в своей работе [15] понимают свойство некоторого объекта сохранять во времени и в установленных пределах значения всех параметров, которые характеризуют его способность выполнять необходимые функции в заданных режимах и условиях функционирования, технического обслуживания, а также хранения и транспортирования.

Надёжность недостаточно качественно (высокая, низкая, приемлемая и т. д.) – её необходимо оценивать количественно, вводя, например, критерии или показатели надёжности, являющиеся количественными характеристиками единичных характеристик (одной или нескольких), с помощью которых представляется возможным определить надёжность объекта.

В целях определения надёжности, разработки мер по её обеспечению и оптимизации требуется знать оценку комплексного свойства надёжности или её количественные характеристики.

Показателем надёжности является количественная характеристика или оценка одного или нескольких свойств, определяющих его надёжность. Показатель надёжности может быть размерной или безразмерной величиной.

В данный момент в теории надёжности применяют вероятностные показатели. Каждый объект описывается вектором единичных и комплексных характеристик. Так как при сравнении одна из характеристик может быть оценена выше альтернативной по одному показателю и ниже по другому, то среди характеристик выбирают ту, которая в определённых условиях использования наиболее лучшим образом показывает свойство надёжности и придаёт ему назначение критерия надёжности. Обычно именно этот показатель нормируется в техническом задании на разработку и в технической документации. Можно утверждать и противоположное: нормируемый показатель надёжности применяют в качестве критерия надёжности. Не следует

считать эти понятия аналогичными, поскольку нормироваться может один показатель, а при сравнении вариантов использоваться другой.

Следует различать критерий надёжности и критерии отказа и предельного состояния. Под критерием отказа понимают признак или совокупность признаков неработоспособного состояния объекта, обозначенных в технической документации этого объекта. Аналогично с приведенным выше определением критерия отказа, критерий предельного состояния является признаком или совокупностью признаков предельного состояния.

Все главные показатели надёжности разделяют на два типа: единичные и комплексные показатели.

Единичные показатели надёжности описывают только лишь одно свойство, определяющее надёжность объекта.

Комплексные показатели надёжности определяют некоторую совокупность свойств, составляющих надёжность объекта.

Рассматривая различные показатели надёжности нельзя не обратить внимание на их наименования и численные значения. Также важными характеристиками показателей являются их математические формулировки и определения. Анализируя численные значения, необходимо помнить, что данный показатель надёжности может изменять своё значение в зависимости от условий его создания и применения, от рассматриваемого периода его существования. Для нахождения значения (численного) надёжности используют либо теоретический, либо практический метод, который определяется с помощью формулировки или математического определения данного показателя. Учитывая, что отказы объектов – величина случайная, то лишь применение курса теории вероятностей и математической статистики позволяют математически определить показатели надёжности. Итак, чтобы определить показатель надёжности некоего объекта, необходимо показать его в виде определенного соотношения (статистического или вероятностного). Вместе с этим нельзя забывать, что существуют такие показатели надёжности, которые являются также и параметрами распределения случайных величин.

В некоторых случаях применяют схему использования или испытаний невосстанавливаемых объектов, учитывая, что некоторые образцы таких объектов работают до отказа (полного) для выявления так называемых статистических показателей надёжности описанных выше объектов.

В таком случае в пике с увеличением числа объектов для испытаний статистические показатели будут сходиться по вероятности к соответствующим показателям вероятности, являющимся некоторой абстракцией в математике. Однако большая часть показателей надёжности более явно показываются в виде не статистических, а вероятностных соотношений. Это делает такие показатели более полезными на практике. В добавок к этому, на этапе проектирования объектов априорные вычисления характеристик надёжности есть возможность проводить исключительно с применением показателей вероятности.

Необходимо анализировать последовательность случайных событий, которые представляют из себя отказы, чередующиеся с интервалами функционирования и восстановления, чтобы выявить математическое определение показателя надёжности восстанавливаемого объекта. Поток отказов называют такую последовательность отказов, которые появляются по очереди в неопределённые моменты времени. При наличии потока отказов объект может быть в нескольких типах состояний:

1. Работоспособность.
2. Полный отказ.
3. Частичный отказ.

При этом работоспособные состояния соответствуют режимам работы объекта, которые установлены в технической документации.

Значениями наработки на отказ называют последовательности случайных величин, которые являются моментами появления отказов с условием использования восстанавливаемых объектов. Именно поэтому для проведения оценки показателей надёжности восстанавливаемых объектов используются условные распределения наработки в интервал времени между отказами, либо

параметры потока отказов.

Когда в потоке с отказом появляется важное последствие, то выявление условных распределений наработки в интервале между отказами оказывается очень важным.

Выделяют три вида показателей:

1. Мгновенные.
2. Интервальные.
3. Числовые.

Эти показатели определяются по типу численных значений главных показателей надёжности от этого промежутка или момента наработки (времени).

Частью заданного периода наработки (времени) являются показатели t_1 и t_2 , называемые интервальными.

Мгновенные показатели соответствуют заданному моменту наработки или значению t .

Числовые показатели не зависят от заданного промежутка или момента наработки (времени).

При рассмотрении такого качества сайта как надёжность со стороны бизнеса надёжность сайта можно называть способностью сайта в заданный промежуток времени составлять конкуренцию сайтам со схожей тематикой и ценовой категорией.

Очевидно, что надёжность сайта неразрывно связана с отсутствием отказов в работе этого сайта. Это отношение и называют определением надёжности в «узком смысле».

В «узком смысле» надёжность сайта называется его свойство сохранять работоспособность в заданный промежуток времени или определённой наработки. Если анализировать это определение, то надёжность сайта заключается в невозможности появления неожиданного ухудшения его качества во время эксплуатации. Такой понятие как надёжность невозможно представлять отдельно от разных сторон процесса эксплуатации.

Надёжность «в широком смысле» является свойством, сочетающим в себе совокупность других свойств, которое может составлять в себе, свойства безотказности, сохраняемости, долговечности, пригодность к ремонту, и т.д

Надёжность включает в себя ряд критериев, которые сейчас рассмотрим.

Источники [16, 44] выделяют следующие критерии надёжности сайта:

1. Стабильность безошибочной работы сайта «из коробки».
2. Возможность получения критических ошибок или ошибок, влияющих на работоспособность сайта при изменении настроек в административной панели (без правки кода).
3. Возможность получить неработоспособность сайта или его критическую ошибку при обновлении CMS программного кода.
4. Возможность получить ошибку после обновления установленных легальных компонентов, плагинов, шаблонов.
5. Возможности резервного копирования сайта. Работоспособность создаваемых бекапов.
6. Возможность получить ошибку, которая нарушает работоспособность сайта, при проведении установки плагина, компонента, дополнения, шаблона.
7. Наличие информирования и предупреждений пользователя при работе с настройками и правке кода в административной панели, когда ошибки могут вывести сайт из строя.
8. Сложность восстановления сайта при нарушении его работы.
9. Присутствие информирования пользователей о необходимости настройки, защиты сайта и рекомендаций по установке полезных компонентов.
10. Возможность системы уведомлять администратора об обнаружении критических уязвимостей и необходимости их неотложного исправления.
11. Преемственность версий (CMS) и возможность безпроблемного обновления с минимальными рисками.
12. Команда разработки сайта, регулярность обновлений и устранения ошибок.

Далее будем рассматривать первый критерий, стабильность безошибочной работы сайта после установки, то есть в нормальном режиме без непредвиденных обстоятельств, поскольку данный критерий является, пожалуй, самым важным.

1.1.1 Стабильность безошибочной работы сайта «из коробки» (вероятность безотказной работы)

Это самый первый из рассматриваемых критериев надежности сайта. При нестабильной работе системы сразу после установки необходимо задуматься о целесообразности её использования.

К сожалению, как показывает практика [32], не все сайты могут стабильно работать без отказов. Причины этого могут быть самыми разными - от ошибок разработчиков, так или иначе связанных с их обслуживанием и наполнением, так и до проблем, возникающих на стороне хостинга (старое оборудование, DDoS-атаки на сервера, либо ошибки самого хостинга). Для проверки работоспособности сайта существует много различных online-сервисов. Также проверить стабильность работы сайта поможет, например, Яндекс.Метрика (раздел "Мониторинг" - "Результаты проверки"). Будет показана информация, показывающая, когда и по какой причине сайт был недоступен.

Важно понимать, что если речь идет о так называемом релизе, предназначенном исключительно для разработчиков, то ошибки вполне допустимы, но при выборе стабильной версии для пользователей ошибок быть не должно.

В стабильной системе не должно быть возможности получения ошибки, которая влияет на работоспособность сайта, либо вовсе критической ошибки, при изменении администратором сайта настроек в административной панели без правки кода.

Стабильная система не должна выдавать ошибок при любых комбинациях настроек или же должно быть указано, что такие-то опции могут

конфликтовать с тем-то и могут вызывать такие-то ошибки.

Однако некоторые системы ведут себя непредсказуемо при настройке. Например, магазинная система Prestashop [58] версии 1.5.5 при включении ЧПУ ссылок перестает отображать изображения товаров в магазине.

Существует ряд сервисов для проверки стабильности работы сайта, таких как LoadImpact [12], BrowserMob [5], Alertra [57], site24x7 [54], You get signal [63], HOST-TRACKER (больше других используют в России – 3,7% посещений) и другие [56]. На рисунках 1.1 и 1.2, расположенных ниже, можно увидеть посещаемость этих сервисов, кроме BrowserMob (на сайте не имеется открытых метрик), определенную по открытым метрикам с помощью сервиса Алекса (сайт, на котором происходит сбор статистика о посещаемости других сайтов) [30].

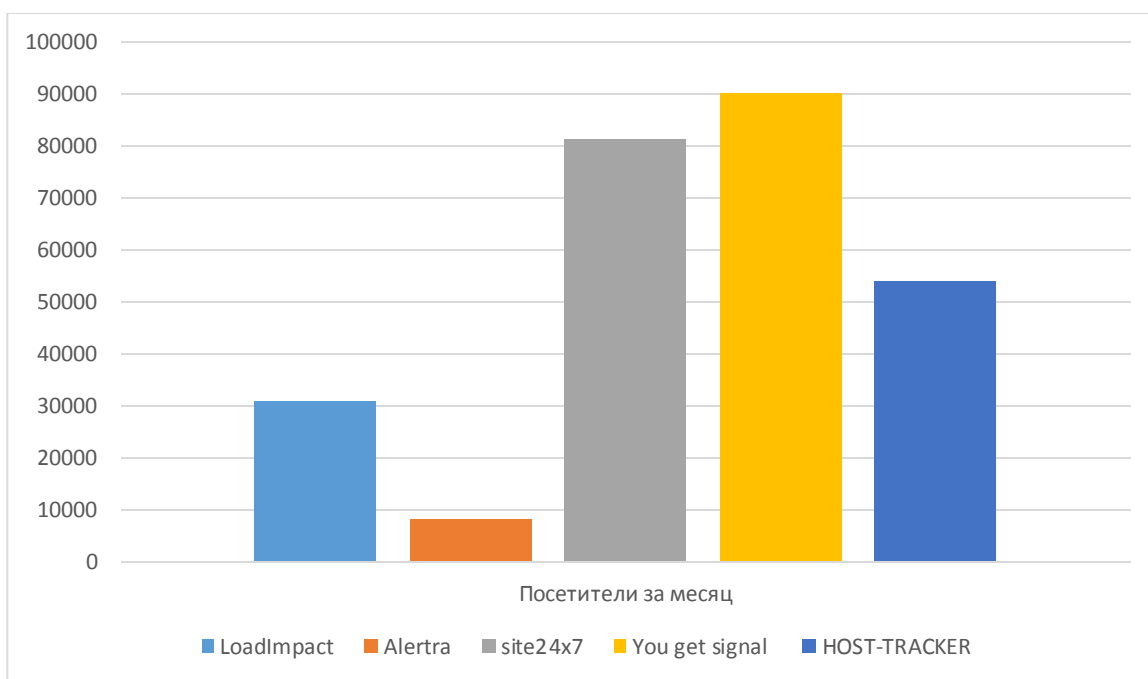


Рисунок 1.1 – Количество посетителей сервисов для проверки стабильности работы сайта в месяц

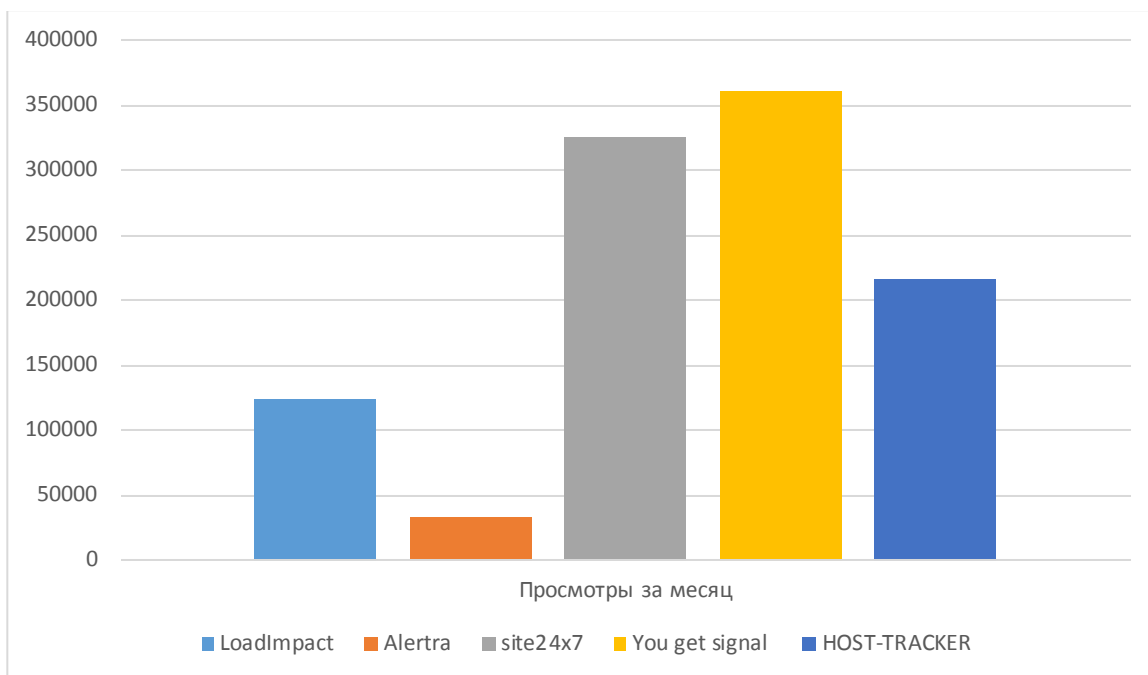


Рисунок 1.2 – Количество просмотров сервисов для проверки стабильности работы сайта в месяц

После исследования практических примеров нарушений в работе сайтов и изучения источников, описывающих работу сервисов по проверке стабильности сайтов можно сделать вывод, что надежность сайта необходимо оценивать уже при его проектировании, чтобы в дальнейшем не возникало проблем, связанных с отказом сайта при повышении потока посетителей. Далее рассмотрим подробнее критерии надёжности веб-сайтов и причины, которые могут привести к нарушению работы сайта из-за нарушения этих критериев.

1.1.2 Возможность возникновения критической ошибки, которая приведет к неработоспособности сайта при обновлении до новой версии

Данный критерий надёжности применим только к CMS и относится к обновлениям, выпущенным разработчиками. CMS или, иначе, система управления контентом сайта – это специализированное программное обеспечение, с помощью которого можно управлять сайтом и наполнять его контентом, не владея при этом специальными знаниями и навыками программирования. В данный момент существует достаточно много разнообразных CMS, все они делятся на платные и бесплатные системы

управления, а также на CMS с открытым и закрытым кодом.

Горнаков С. [20] среди самых популярных на данный момент CMS выделяет такие, как 1С-Битрикс, UMI.CMS, WordPress, Joomla.

Все CMS, как и любое другое программное обеспечение, периодически выпускают свои обновленные версии. В обновленных движках присутствуют не только новые функции и возможности для сайта и его развития, но и усовершенствованная система безопасности.

Ошибки в системе безопасности устаревшей CMS сайта играют роль некоторого пригласительного билета для хакеров. Именно сайты на устаревшей CMS в первую очередь подвергаются атакам и взломам. Как правило, взломы совершаются с целью принудительно перенаправить посетителя сайта на абсолютно сторонний ресурс или с целью распространения вируса среди посетителей. Особое внимание владельцам сайтов здесь следует обратить внимание на то, что сайты, зараженные вирусом, блокируются поисковыми системами. По статистике около 70% взломов приходится именно на сайты с устаревшей CMS.

Одновременно с процессом совершенствования программного обеспечения свои навыки улучшают и взломщики, появляются новые вредоносные программы и вирусы. По этой причине, чтобы защитить своих пользователей, разработчики сайтов вынуждены выпускать более безопасные версии программного обеспечения.

Автор [44] приводит в пример, что при установке обновлений через автообновление в системе Joomla версии 2.5.10-2.5.12 происходила подмена файла robots.txt на стандартный. Бывают случаи, когда обновить систему не получается по причине несовместимости некоторых расширений, плагинов и т.д., как например при обновлении Joomla 2.5 до Joomla 3.1. В итоге высока возможность неработоспособности сайта.

Почти все современные системы управления сайтом в автоматическом режиме проверяют наличие обновлений системы и дают возможность администратору выполнить обновление из административной панели. Но, к

сожалению, встречаются случаи, когда после обновления сайт становится неработоспособным, либо происходит нарушение его важных функций.

Во избежание проблем, которые могут привести к дополнительным затратам, владельцам сайтов необходимо все пристальнее следить за выходом новых версий ПО и своевременно его обновлять до актуальной версии.

К сожалению, данная проблема появляется, поскольку были допущены некоторые ошибки при разработке, и она может решиться только на стороне разработчика.

1.1.3 Возможность возникновения ошибки при обновлении установленных легальных компонентов, плагинов, шаблонов (CMS или фреймворк)

Необходимо заметить, что далеко не все системы управления сайтом имеют централизованную систему обновлений компонентов и отображением обновлений, и вовсе не все разработчики сайтов поддерживают свои продукты. Порой администратору приходится самому проверять настройку отдельных плагинов или расширений на наличие обновлений, или даже проводить постоянный мониторинг сайтов данных расширений на предмет наличия обновлений, что является крайне неудобным.

Например, в CMS Joomla нужно отдельно проверять обновления самой системы, а отдельно обновления компонентов. Но даже при этом ряд установленных компонентов может отсутствовать в списке. Так, компонент Акееба Backup, который описывается в источнике [62], который помогает при резервном копировании, проводит проверку обновлений через интерфейс самого компонента и не интегрирован в общую систему обновлений

Однако в ряде систем обновление установленных компонентов организовано грамотно. К примеру, в CMS WordPress присутствует единая система обновлений, которая включает обновления CMS, плагинов и тем оформления.

При разработке сайтов с использованием подобных систем разработчику необходимо всегда проверять совместимость плагинов между собой и с версией используемой системы.

1.1.4 Возможности резервного копирования сайта. Работоспособность создаваемых бекапов

Бэкап (от английского back up – запас) – создание полной или частичной резервной копии данных, таких как операционные системы, сайты или какое-либо программное обеспечение. Вся суть данного термина сводится к копированию данных, FTP-аккаунтов, файлов сайта, почтовых данных и других параметров хостинга. Иными словами, происходит сохранение всего сайта и его настроек в отдельном месте, и при необходимости администратор сайта может вернуть сайт к версии, которая была сохранена. При этом копирование данных может осуществляться на текущий («бэкапный» или дополнительный) сервер, который располагается отдельно от серверов провайдера, либо вовсе в другом дата-центре. Копирование производится на тот случай, когда что-то случится с сервером, на котором располагается сам сайт. На рисунке 1.3 изображена схема работы с backup.

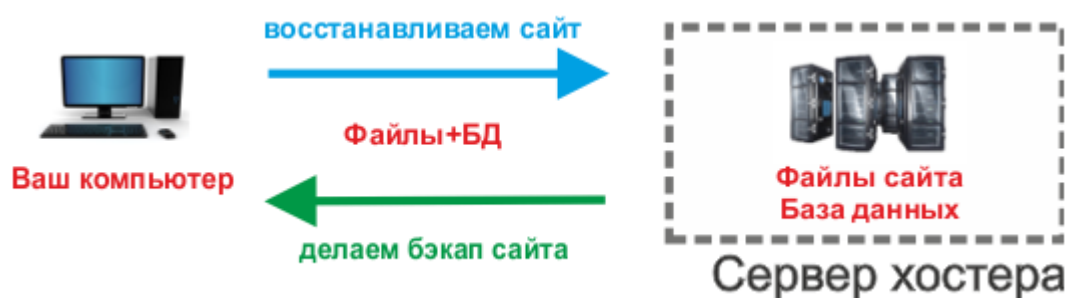


Рисунок 1.3 – Схема работы backup

Проблема резервного копирования рассматривается во многих источниках [14, 11, 28].

Существует возможность восстановить выполненный хостинг-провайдером бэкап при использовании исключительно панели управления хостинг-аккаунтом, или обратившись к провайдеру, после чего можно начинать

работу над повышением уровня безопасности данного веб-проекта, а не заниматься «по-старинке» восстановлением сайта по частям.

Однако может возникнуть и более сложная проблема – может произойти какое-либо стихийное бедствие, в результате чего работоспособность сервера нарушится, а он, как и все остальные сайты, и соответственно содержащиеся в нём бэкапы тоже перестанет работать. Для подобных непредвиденных случаев существует сервис двойного резервного копирования.

Хостинг-провайдер имеет возможность оперативного восстановления работы своих сервисов и выполнения восстановления резервных копий данных пользователей путём взятия их с другого бэкапного сервера, расположенного в другом дата-центре, и на который соответственно не могли повлиять эти непредвиденные обстоятельства, которые и нарушили работу сервера и сайтов.

По этой причине перед приобретением хостинга следует узнавать у провайдера, предлагает ли он заказчикам сервис резервного копирования данных и как регулярно он выполняет это резервное копирование и сколько времени у него хранятся копии. Именно от этого зависит бесперебойная работа интернет-проекта, размещённого на том хостинге.

Желательно делать резервное копирование сайта каждый день. Зачастую сам хостинг создает резервные копии в автоматическом режиме, и эти копии хранятся там около двух недель. Рекомендуется загружать их себе на диск (или облачное хранилище) примерно 1-2 раза в месяц. Для большего количества сайтов это будет хороший залог бесперебойной работы и эффективности.

Многие CMS не имеют надежной системы резервного копирования. Так, в WordPress существует возможность копировать только ваш контент сайта, но не всю базу данных и файлы сайта. Возможности резервного копирования в данной системе описывает в своей работе [25]. Конечно, проблема решается установкой плагинов резервного копирования.

Стоит отметить, что созданные бекапы всегда целесообразно проверять на предмет работоспособности. Не проверенный бекап равнозначен его отсутствию. Для большинства сайтов оптимальным называют количество

резервных копий, которые создаются за один год. Если делать их 1-2 раза в месяц, то оптимальным считается 12-25 копий в год.

1.1.5 Сложность восстановления сайта при нарушении его работы

Еще один критерий надежности сайта — это сложность восстановления с после возникшей ошибки, приведшей к его неработоспособности.

Не обращая внимание на то, что мир интернета постоянно движется в сторону совершенствования, все равно продолжают свою работу хакеры, возникают сбои в работе сайтов и разного рода ошибки. И так называемая «поломка» сайта, есть приостановление его работоспособности, является вполне реальным явлением, которое может причинить множество неудобств для владельцев ресурсов, снизив эффект ведения любого бизнеса, связанного с сайтами. Когда произошёл сбой в работе интернет-магазина, без восстановления сайта невозможно обойтись. Исключив тот момент, когда сайт только создан и на нём практически нет контента, тогда создание сайта заново не вызовет больших трудностей, если сравнивать с давно функционирующим сайтом. Но если веб-проект достаточно долго функционирует, он богат различного рода контентом, то при нарушении его работоспособности восстановление сайта без потери данных является очень важным моментом.

Встаёт самый важный вопрос, как же происходит восстановление сайта. Этот процесс может отнять довольно много времени, если им заниматься самостоятельно и не иметь при этом необходимых знаний. Такой процесс как восстановление может оказаться и вовсе невозможным, если не были сохранены резервные копии последней версии проекта. Резервные копии часто приходится делать вручную при том, что существуют хостинги, которые автоматически создают бэкапы сайта и баз данных, что является достаточно удобным для владельцев ресурсов. Ведь, например, используя резервные копии восстановление сайта будет довольно нетрудоёмким процессом. Конечно, по сути, это будет являться новым сайтом, но с абсолютно идентичным наполнением, что и было в предыдущем.

Восстановление сайта из резервной копии происходит в несколько основных этапов:

- первым этапом будет являться создание резервной копии на случай, если в процессе восстановления будут происходить очередные сбои;
- на втором этапе необходимо изменить ключи доступа и пароль к сайту перед началом процесса чистки интернет-ресурса. Это важно, по причине того, что важна уверенность, что у злоумышленников не имеется несанкционированного доступа;
- на третьем этапе необходимо запустить сканеры сайта, которые позволяют выявлять повреждённые зоны;
- при восстановлении сайта на четвёртом этапе необходимо переустановить движок данного сайта;
- на пятом этапе необходимо провести проверку папки контента и темы, так как в них может содержаться что-либо подозрительное, что вредит работе данного ресурса;
- и на последнем шестом этапе следует проверить и, если потребуется, переустановить все необходимые плагины.

Чем меньше усилий требует восстановление сайта, тем выше уровень его надёжности. Так, в CMS Wordpress, если после установки плагина в его работе наблюдаются ошибки, следует просто удалить папку плагинов через ftp-сервер и скорее всего сайт восстановит свою работоспособность.

1.2 Методы анализа надёжности веб-сайта

Такая сфера как надёжность на этапе проектирования содержит в себе ряд инструментов и практических советов, а также порядок применения тех советов, которыми не следует пренебрегать организации для обеспечения высокого уровня надёжности и ремонтпригодности продукта, который разрабатывает эта организация, для достижения достойного показателя готовности продукта, минимизации вложений и максимизации срока эксплуатации продукта. Зачастую на первом этапе работы в данном

направлении происходит нормирование показателей надёжности. Данный показатель должен быть заранее задан, как бы спроектирован, в системе. На этапе проектирования системы задаются условия, которым должна соответствовать надёжность верхнего уровня, далее они делятся на определённые части разработчиками, инженерами по надёжности и конструкторами, которые работают в команде. Первым этапом в проектировании надёжности является разработка модели. В данном процессе используют деревья неисправностей, при помощи которых показывается связь между разными компонентами системы, или схемы надёжности.

Резервирование или введение избыточности является одной из самых важных техник проектирования. Резервированием называется метод обеспечения надёжности продукта за счет добавочных средств и (или) возможностей, которые являются избыточными по отношению к в наименьшей степени необходимым для выполнения обязательного функционала [1, 2, 3]. В том числе и системы с низким уровнем надёжности по одному каналу могут обладать высокой надёжностью по всем каналам в целом, если ввести избыточность в связке с организованным на достаточном уровне мониторингом отказов. Впрочем, введение избыточности в сложной системе на высоком уровне (к примеру, на уровне двигателя самолета) влечет за собой увеличение затрат и трудоемкости, что дает ограничение на такое резервирование. На более низком уровне системы резервирование происходит достаточно быстро и не влечет за собой больших трудозатрат, так, к примеру, использование дополнительного соединения болтом.

Имеется достаточное количество методов анализа надёжности, которые обладают особенностями для конкретных отраслей промышленности и приложений. Под самыми общими выделяют следующие методики [23].

1. Анализ видов и последствий отказов (АВПО).
2. Имитационное моделирование надёжности.
3. Анализ схем функциональной целостности (СФЦ).
4. Анализ опасностей (Hazard analysis).

5. Анализ структурных схем надежности (RBD).
6. Анализ деревьев неисправностей.
7. Ускоренные испытания.
8. Анализ роста надежности.
9. Вейбулл-анализ (анализ эмпирических данных из испытаний и эксплуатации).
10. Анализ смеси распределений.
11. Устранение критичных отказов.
12. Планирование технического обслуживания, обеспечивающего надёжность (RCM).
13. Анализ диагностики отказов.
14. Анализ ошибок человека-оператора.

Для выявления идеального баланса между различными ограничениями и требованиями и надёжностью используются инженерные исследования. При проведении инженерного анализа существенную помощь могут оказать программные комплексы для расчета надёжности.

Далее приведём анализ некоторых из приведённых выше методов анализа надёжности.

1.3 Анализ видов и последствий и критичности отказов (АВПКО)

Метод анализа видов и последствий и критичности отказов (АВПКО) широко применяется в различных сферах деятельности. Подобное достаточно широкое распространение этого метода оправдывается его хорошей способностью приспосабливаться к индивидуальным чертам конкретной сферы деятельности, условиям эксплуатации и характеру продукции, в которых его нужно использовать.

Такой метод как АВПКО представляется возможным использовать как основу при проведении исследований по обеспечению вместо точечного результата измерения показателей надёжности интервального.

Следующими стандартами в России регламентируется проведение

анализа данным методом:

- ГОСТ 27.310-95 «Анализ видов, последствий и критичности отказов» (АВПКО); используется для проведения оценки технических объектов всех видов;
- ГОСТ Р 51901.12-2007 «Менеджмент риска. Метод анализа видов и последствий отказов»; дает общие сведения к проведению FMEA-анализа;
- ГОСТ Р 51814.2-2001 «Системы качества в автомобилестроении. Метод анализа видов и последствий потенциальных дефектов»; распространяется на технические объекты автомобилестроения.

Смысл этого метода состоит в оценивании надёжности системы с помощью расчета критичности отказов каждой отдельной части этой системы, и затем сравнения критичности отказа данной части с пороговым значением, которое было установлено до проведения оценки.

Согласно стандарту ГОСТ Р 51901.12-2007, критичностью отказа является совокупность частоты появления и тяжести последствий или других свойств отказа как показатель необходимости идентификации причин, источников и сокращения количества или частоты появлений этого отказа и минимизация тяжести его последствий.

Критичность (C_i) вычисляется с помощью формулы:

$$C_i = B_{1i} + B_{2i} + B_{3i}, \quad (1.1)$$

где B_{2i} - балльная оценка вероятности возникновения i -го отказа за время эксплуатации (B_{1i}),

B_{2i} - балльная оценка тяжести последствий i -го отказа,

B_{3i} - балльная оценка вероятности обнаружения i -го отказа до поставки изделия потребителю.

Процесс оценивания методом АВПКО влечет за собой необходимость обработки и формирования большого объёма данных, которые обязательны для выполнения следующих этапов: выявление причин и описание отказов, расчет вероятности возникновения отказов, определение последствий этих отказов и их критичности.

Алгоритм проведения анализа состоит из ряда этапов, представленных ниже [10]:

1. Построение и анализ функциональной и иерархической схем объекта.
2. Анализ взаимодействия элементов объекта.
3. Выделение наиболее ответственных элементов объекта. Здесь используется информация об отказах объектов-аналогов и метод диаграммы Парето.
4. Составление перечня отказов объекта и их причин. На этом этапе используются классификаторы отказов и и НТД на объект, априорная информация и инженерный анализ КД.
5. Оценка степени серьезности и частоты отказов с использованием экспертного метода.
6. Расчет чисел критичности для отказов (ГОСТ 27.310-95).
7. Распределение отказов по уровням критичности с использованием диаграммы Исикавы.
8. Установление корректирующих действий и рекомендаций.

Этапы 1-5 представляют собой качественный анализ, последующие этапы 6 и 7 относятся к количественной оценке критичности отказов. Формирование экспертной группы предшествует началу анализа.

Рассмотрим более подробно шестой этап, который связан с расчетами на критичность отказов.

Главная идея для расчета критичности отказа включает в себя три фактора:

- вероятность (частота) отказов;
- возможность обнаружить дефект до ввода продукта в эксплуатацию;
- последствия отказа.

Эти три фактора вместе и составляют критичность отказа [10].

Предельно ясно, что, чем выше значение хотя бы одного из перечисленных

выше факторов, тем выше значение критичности отказа.

Коэффициенты B_{1b} , B_{2i} и B_{3i} определяют эксперты по шкале от одного до десяти, рекомендуемым ГОСТ 27.310-95 [17] (таблицы 1.1, 1.2, 1.3), или по шкалам, разрабатываемым специально для конкретного вида продукции. Согласно источнику [48] число экспертов в группе варьируется от 4 до 8 человек, согласованность оценок которых определяется с помощью коэффициента конкордации.

Таблица 1.1 – Оценка вероятности отказов в баллах (ГОСТ 27.310-95) [17]

Виды отказов по вероятности возникновения за время эксплуатации	Ожидаемая вероятность отказов, оцененная расчетом или экспериментальным путем	Оценка вероятности отказа в баллах B_1
Отказ практически невероятен	Менее 0,00005	1
Отказ маловероятен	От 0,00005 до 0,001	2
Отказ имеет малую вероятность, обусловленную только точностью расчета	От 0,001 до 0,005	3
Умеренная вероятность отказа	От 0,005 до 0,01	4
Отказы возможны, но при испытаниях или в эксплуатации аналогичных сайтов не наблюдались	От 0,001 до 0,005	5
Отказы возможны, наблюдались при испытаниях и в	От 0,001 до 0,005	6

Виды отказов по вероятности возникновения за время эксплуатации	Ожидаемая вероятность отказов, оцененная расчетом или экспериментальным путем	Оценка вероятности отказа в баллах В1
эксплуатации аналогичных сайтов		
Отказы вполне вероятны	От 0,005 до 0,01	7
Высокая вероятность отказов	От 0,01 до 0,10	8
Вероятны повторные отказы	Более 0,11	10

Таблица 1.2 – Оценка последствий отказов (ГОСТ 27.310-95) [17]

Описание последствий отказа	Оценка последствий в баллах В ₂
Отказ не приводит к заметным последствиям, пользователь вероятно не обнаружит наличие неисправности	1
Последствия отказа незначительны, но пользователь может выразить недовольствие его появлением	2-3
Отказ приводит к заметному для потребителя снижению эксплуатационных характеристик и / или к неудобству использования сайта для пользователя	4-6

Описание последствий отказа	Оценка последствий в баллах B_2
Высокая степень неудобства для пользователя, сайт не может быть использован по назначению, но угрозы безопасности отказ не представляет	7-8
Отказ представляет угрозу безопасности (например, возможна кража персональных данных, утечки информации или кража денежных средств со счета)	9-10

В случае наличия достаточного объема статистической информации об отказах элементов сайта или аналитического определения вероятности их возникновения рассматриваемому отказу присваиваются баллы в соответствии с интервалом значений, в котором находится значение вероятности обнаружения или возникновения отказа. Если указанные данные отсутствуют, то коэффициенты определяются экспертами, при этом эксперту необходимо опираться на собственные знания в данной области.

Согласно мнению И.Н. Ефстафьева [27], при оценке вероятности, особенно экспертным путем, можно устанавливать не вероятность, частоту событий (предполагаемую) за некоторый отрезок времени, к примеру, за срок эксплуатации или за один год, поскольку люди имеют способность достаточно достоверно оценивать частоту, однако довольно трудно поддается оценке вероятность события. Если речь идет о новом продукте, то производится пробная версия для определения вероятности отказа (версия для тестирования) [25].

Далее значения критичности, которые рассчитываются для каждого отказа C_i (формула (1.1)) сравниваются с пороговым значением $C_{кр}$, которое было принято ещё до проведения анализа надёжности.

В том случае, когда $C_i > C_{кр}$, необходимо принять решение о необходимости введения действий по корректировке, поскольку признается значимым данный отказ [26].

Источники [10] рекомендуют в качестве предельного значения назначать $C_{кр} = 125 (5 \cdot 5 \cdot 5)$, но организация, которая выпускает продукт, может назначить сама значения коэффициента критичности при проведении анализа.

В принципе, на основании найденного значения критичности можно осуществить принятие решения для повышения качества продукции, выпускаемой организацией. Например, можно существенным образом изменить вариант (схему) технического обслуживания. Также возможен вариант, при котором организация примет решение осуществить дополнительный контроль, а также многие другие решения, которые будут направлены на повышение качества продукта. Опираясь на результаты АВПКО, осуществляют сертификацию объектов при проведении определенных проверок уровня принятых для обеспечения безопасности мер, определенных на этапе разработки.

Необходимо помнить, что не исключена возможность негативного влияния совершенных ошибок на этапе нахождения численного значения критичности отказов на формирование решения о повышении качества выпускаемого продукта. Причины возникновения таких ошибок могут быть достаточно разнообразны. Основная причина – определенные несовершенства системы АВПКО (процедуры ее проведения). Рассмотрим такую ситуацию подробнее.

Применяемые в данной работе методы имеют следующее несовершенство: оценки критичности, определяемые экспертами, зачастую оказываются субъективными. Такое допущение часто способствует неправильному нахождению численного значения критичности C_i . Другими словами, возникает так называемая погрешность ΔC , способы нахождения которой в настоящее время еще не описаны.

Из-за возникновения такого рода погрешностей есть определенные риски,

что управленцы организации примут ошибочное решение, которое приведет к существенным денежным затратам и производителя продукта, и потребителя этого продукта.

Нельзя забывать и о таком важном факторе, как недостаточная априорность информации об определенных свойствах элементов (системы), которые необходимо объединять, когда производится анализ надежности с использованием анализа видов, критичности отказов и последствий.

Производя процесс нахождения описанных выше коэффициентов В, необходимо обладать пониманием структуры системы, которая предложена к рассмотрению, а также знать отдельно о каждом из элементов этой системы, их свойствах. В таких случаях получается описывать данный процесс, используя деревья отказов или неисправностей (ДО – дерево отказов / ДН – дерево неисправностей). Применение таких деревьев осуществляют с использованием ГОСТ 27.310-95.

Применение ГОСТ 27.310-95 для технически сложных систем АВПКО осуществляется с использованием аппарата функциональных методов, применение которых имеет следующие основные требования:

1. В обязательном порядке производится процесс идентификации каждой функции, которую выполняет объект, определенный для исследования данным методом.

2. Производится процесс определения вариантов вероятных ошибок (отклонений) с помощью таких методов как априорные данные, исследовательский опыт, инженерный анализ, а также любым другим способом, применимым к конкретной ситуации.

3. Для каждого определенного выше отклонения производится оценка рисков (ущерба), а также в некоторых случаях качественная оценка того, насколько тяжелы будут эти последствия для организации, которая производит данный продукт.

4. Среди найденных значений отбирают критические отклонения, потери, от воздействия которых будут существеннее сказываться на

организации, чем это планомерно установлено предварительным анализом.

5. Для каждого такого критического отклонения, выявленного на предыдущей итерации, производится формирование (построение) дерева отказов, который описывает отказ элементов системы на каждом уровне. В том числе и на самом нижнем уровне, определенном предварительным анализом. Критическое отклонение в таком дереве отказов является так называемым «вершинным событием».

6. Применяя дерево отказов, определяют элементы, приводящие к критическим отклонениям в работе веб-сайта. Также определяют сочетание таких элементов, наложение которых друг на друга также приводит к критическому отклонению.

7. Осуществляется процесс определения вероятности отказа элементов сайта, а также возможность отказа определенного сочетания этих элементов. С помощью найденных значений вероятностей определяют показатели критичности комбинация отказов или единичных отказов.

Следует учитывать, что в описанном выше стандарте ГОСТ 27.310-95, а также в других подобных источниках [9, 10] отсутствует полноценная информация о корреляции уровня дерева отказов и достоверности определения оценки надежности.

В ГОСТ 27.310-95 указаны только общие пояснения о том, что должна содержать в себе методика оценки информации о достоверности применяемых данных, а также об уровне, на котором построение дерева должно быть прекращено. Основными критериями, с помощью которых определяют нижние уровни, являются результаты проведенного предварительного анализа, наличие требуемых данных, информация о цикле жизни продукта и т.д.

Выполнение мероприятий по оценке надежности в настоящее время имеет низкое качество. Это связано с несовершенством методик, а именно с тем, что в них нет полноценных рекомендаций.

Используя при анализе надежности таких методов как анализ видов, последствий и критичности отказов, необходимо понимать, что сам этот

процесс можно принять за технические измерения. Экспертов, принимающих участие в таком анализе можно принять за так называемые средства измерения. В таком случае результаты проведения такой оценки надежности можно определить в следующую форму:

$$P C_i - \Delta C_i < C_i < C_i + \Delta C_i = \dots, \quad (1.2)$$

где P – принятая доверительная вероятность;

C_i – точечная оценка критичности, она определяется как среднее арифметическое значение критичности, которые определяют эксперты для каждого отказа i ,

ΔC_i – граница погрешности точечной оценки критичности для каждого i -го отказа, которую можно рассчитать по формуле ниже:

$$\Delta C_i = \mp t * S_{C_i}, \quad (1.3)$$

где t – относительная ширина доверительного интервала погрешности значения критичности. Данная величина зависит от принятой ранее доверительной вероятности, от вида закона распределения вероятностей, а также от результатов оценки коэффициентов B .

S_{C_i} – оценка СКО (среднего квадратического отклонения) величины критичности C_i ,

$C_i - \Delta C_i < C_i < C_i + \Delta C_i$ – доверительный интервал значений параметра C_i .

Значение оценки СКО можно определить, используя формулы (1.1), следующей зависимостью:

$$S_{C_i} = \sqrt{\sum_{j=1}^3 \frac{\partial C_i}{\partial B_{ij}} * S_{B_{ij}}^2} = \sqrt{B_{2i} * B_{3i} * S_{B_{i1}}^2 + B_{1i} * B_{3i} * S_{B_{i2}}^2 + B_{1i} * B_{2i} * S_{B_{i3}}^2}, \quad (1.4)$$

где i – номер рассматриваемого отказа,

j – номер коэффициентов по порядку ($j=1\dots 3$),

$S_{B_{ij}}$ – оценки средних квадратических отклонений средних

арифметических значений коэффициентов B_{1i}, B_{2i}, B_{3i} для рассматриваемого i -го отказа.

$S_{B_{ij}}$ определяется по следующей формуле:

$$S_{B_{ij}} = \frac{\sum_{i=1}^n (B_{ij} - B_{ij})^2}{n}, \quad (1.5)$$

где n – количество экспертов в группе,

B_{ij} – среднее арифметическое значение результатов оценок коэффициентов B_{1i}, B_{2i}, B_{3i} , которое предоставлено экспертами.

Необходимо понимать, что основным критерием, который оказывает существенное влияние на то, насколько эффективно была произведена оценка надежности, является квалификация экспертной группы. Это обусловлено тем, что в основе данного метода лежит экспертная оценка. Также важным критерием является априорная информация, так как на ее основании и происходит оценка, производимая экспертной группой.

Также необходимо отметить, что, прикладывая метод АВПКО для анализа надежности, применяется информация об уже выявленных отказах. Безусловно, в уже существующей работающей системе с помощью данного метода возможно выявить уровень надёжности, но при проектировании системы оценка надёжности методом АВПКО будет довольно сложным процессом, так как он основывается на экспертном мнении.

1.4 Анализ деревьев неисправностей

Анализ дерева отказов (неисправностей) или АДО, в английской терминологии FTA, является методом анализа отказов сложных систем, в котором отказы системы или нежелательные состояния подвергаются анализу с использованием методов булевой алгебры, соединяя последовательность отказов низшего уровня (нижестоящих событий), приводящих к отказу всей системы.

Согласно источнику [9], анализ дерева отказов интенсивно используется в различных сферах, к примеру, в машиностроении для понимания того, как

система может выйти из строя, определить метод минимизации рисков или определения частоты системного отказа.

Такой метод как анализ дерева неисправностей широко используется в различных отраслях, связанных с высокой степенью риска: химической, атомной энергетике, аэрокосмической, перерабатывающей, нефтехимической, фармацевтической и других.

Данный метод активно используется и описывается в следующих источниках [43, 40, 53].

В некоторых отраслях используется более широкий термин «условие отказа системы», с помощью которого обозначается «нежелательное состояние» верхнего события дерева отказов.

Эти условия классифицируются по степени последствий. Для описания самых тяжелых условий необходимо составление наиболее широкого анализа дерева неисправностей.

Обычно классификация условий отказа системы и сами условия заранее определяются в функциональном анализе рисков и опасностей появления отказов.

Метод анализа деревьев неисправностей широко используется для:

1. Четкого понимания логики, которая ведет к нежелательному состоянию (отказу системы) или верхнему событию.
2. Определения соответствия с требованиями к надёжности или системой безопасности.
3. Определения ранга участников, которые ведут к вершине, что является критерием для создания важного оборудования (запчастями или списками событий).
4. Мониторинга и контроля показателей состояния сложных систем. К примеру, безопасность полета на самолете при неисправностях топливного клапана.
5. Минимизации и оптимизации ресурсов.

6. Помощи в проектировании системы. АДО может быть использован в качестве средства проектирования, которое помогает создавать требования.

7. FTA может быть применен для диагностики при выявлении и исправлении причин верхнего события. Это может оказать помощь при создании процессов и диагностических руководств.

Использование дерева отказов в качестве инструмента, помогающего проанализировать отказы по степени надежности, очень распространено в настоящее время.

Анализ дерева отказов состоит из логических схем, которые позволяют определить состояние, в котором пребывает система. АДО строят с применением методов графического проектирования.

Учитывая жесткие требования к квалификации, т.е. к знаниям анализируемой системы, основными разработчиками такой методики были инженеры.

Анализ дерева отказов зачастую содержит некие события неисправности, такие как изнашивание материала, фактов истории, периодом отказа систем и т.д.

Также существует практика ввода человеческого фактора (некий процент механической или логической ошибки человека, к которому относится неправильный ввод информации человеком или неправильный ввод из-за его вмешательства в работу в систему в неподходящее время). Однако, данный фактор не является основной частью АДО.

Анализ дерева отказов является ценным инструментом, применяемым к определению вероятности потенциальных отказов и исключаящим дорогостоящие конструктивные изменения в продукте.

Нередко анализ дерева отказов применяют как некий инструмент диагностики, который с некоторыми допущениями предсказывает вероятность ошибок системы, если возникнет сбой.

Проводя сравнение с деревом успехом, стоит отметить, что проведение ФТА с помощью анализа дерева отказов гораздо менее трудоемкий процесс определения вероятности отказа.

Однако, несмотря на это эта методика является трудоемкой и дорогостоящей, если пытаться применить ее ко всей системе. Поэтому зачастую ограничиваются использованием анализа дерева отказов в рамках подсистем, а не систем в целом.

Производя анализ для подсистем, упрощается системный анализ расчета, что обозначает уменьшенную вероятность ошибки, которую можно допустить, совершая такие расчеты. В дальнейшем, если это необходимо, расчеты, произведенные для подсистемы, можно сгруппировать и внедрить (интегрировать) для формирования большой системы анализа.

Так называемым главным событием (иногда называют корневым событием) дерева логики принимается нежелательное последствие. Логическая цепочка, предназначенная для формирования пути до события, находящегося в самом вершине этой цепочки, может быть достаточно разнообразной.

В таком случае будет достаточно полезным приложение методов функционального анализа опасности. Его основным критерием является «опыт». Принцип такого дерева иерархический, то есть существует некое основное событие, от которого все задачи идут вниз.

Следом за задачами следуют логические выражения, которые основаны на ситуациях, приводящих к данному эффекту. Затем производится наполнение деревьев отказа реальными вероятностями неудачных исходов. Все вышеперечисленное позволяет компьютеру произвести вычисление вероятности отказа из дерева отказов (неисправностей).

Данное дерево обычно состоит из цепочки обычных логических символов. Такое дерево обязательно включает в себя сечение, которое по сути является путем от события до инициатора события. Самый короткий такой маршрут является минимальным сечением.

Зачастую во многих отраслях промышленности применяют деревья и отказов, и событий. Работа в такой системе происходит следующим образом: нежелательный инициатор, которым может быть, как потеря питания, так и отказ конкретных компонентов, порождает событие дерева, которое в дальнейшем двигается по вероятным последующим событиям системы через ряд окончательных последствий или результатов.

Разделяют два типа символов, которые применяют для построения дерева отказов:

1. Символы событий.
2. Символы элементов.
3. Символы передачи.

Рассматривая символы событий, необходимо отметить, что они в основном коррелируют с первичными, а также с промежуточными событиями.

Первичные события не развиваются в дереве отказов. Промежуточные же события получают развитие в системе (дереве отказов) и располагаются на выходе элементов.

Любое событие в дереве имеет функциональную зависимость со статистической вероятностью. Иными словами, вероятность появления любого события может быть определена исключительно экспериментальным путем.

Приведем пример. Зачастую сбои в работе некоторых компонентов связаны с некоторой постоянной интенсивностью этих сбоев λ .

Данный случай наиболее простой, так как вероятность отказа в таком случае коррелирует с интенсивностью λ и временем t по экспоненциальному закону (1.6).

$$P = 1 - \exp(-\lambda t), \quad (1.6)$$

где $P \approx \lambda t, \lambda t < 0,1$.

Другими словами, вероятность отказа узла или отдельного компонента в течение t часов равна выражению, вычисляемому по формуле (1.6).

Существует определенное ограничение дерева отказов временными интервалами, например, если речь идет о часовом времени полета или неким

среднем временем полета.

Похожая функциональная зависимость существует также между вероятностью события и временным интервалом, охарактеризованным в предыдущем абзаце.

Зачастую, для упрощения построения логических диаграмм (логических символов) применяют бинарный код, который применяют для обозначения входного и выходного значений (0 - ложь, 1 – правда). Вполне логично и, как оказалось, правдиво предположение о том, что вероятность события на входе имеет непосредственное влияние на вероятность выходного события.

Между бинарными кодами, как правило, применяют определенные логические операторы, например, «И», который обозначает нахождения справа и слева от него событий, не оказывающих влияние друг на друга. То есть между ними отсутствует корреляция на входе и на выходе. Другими словами, прикладывая к этому рассуждения теорию множеств, обнаружим, что описанное выше условие, по сути, однозначно с пересечением входных событий (т.е. у событий одинаковый вход). Для такого случая вероятность выхода можно рассчитать, применив формулу (1.7).

$$P(A \text{ and } B) = P(A \cap B) = P(A)P(B) \quad (1.7)$$

Символ «ИЛИ» в отличие от символа «И» соответствует объединению. В этом случае вероятность выхода можно определить, используя формулу (1.8).

$$P(A \text{ or } B) = P(A \cup B) = P(A) + P(B) - P(A \cap B) \quad (1.8)$$

Учитывая небольшую вероятность отказа в дереве отказа (менее 0,01), вероятность события выхода обычно принимает достаточно маленькое значение. Таким образом можно принять допущение, что входные события являются взаимоисключающими. В этом случае вероятность выхода можно определить, используя формулу (1.9).

$$P(A \text{ or } B) \approx P(A) + P(B), P(A \cap B) \approx 0 \quad (1.9)$$

По сути в таком случае символ «ИЛИ» становится исключаящим и в случае с двумя входами можно определить вероятность того, что активен только один вход: первый или второй. Оба входа одновременно в этом случае

активны быть не могут. В этом случае вероятность выхода можно определить, используя формулу (1.10).

$$P(A \text{ xor } B) = P(A) + P(B) - 2P(A \cap B) \quad (1.10)$$

Учитывая малое численное значение вероятности, исключаящий символ «ИЛИ» используют в дереве отказов с ограничениями.

При моделировании ФТА применяют различные подходы. Наиболее часто применяемые подходы включают в себя несколько основных шагов.

Применение одиночного дерева отказов ограничено только одним нежелательным событием (верхним событием). В дальнейшем оно перемещается в другое дерево неисправностей, где превращается в основное событие в данном дереве.

Следует понимать, что несмотря на различную природу нежелательных событий, встречающихся в реальной жизни, при моделировании с использованием ФТА используется унификация природ нежелательных событий. Другими словами, для этой методики нет различия в задержке длительностью 0,25 мсек при получении электроэнергии и в пожаре в грузовом отсеке.

Применение ФТА достаточно затратное. Поэтому данную методику применяют исключительно для серьезных нежелательных последствий.

Ниже перечислены пять основных шагов ФТА:

1. Для начала необходимо определить нежелательное событие. Несмотря на очевидность и простоту некоторых событий, определить нежелательное событие достаточно трудно. Необходимо помнить, что никакие два различных события не могут быть применимы к одному и тому же дереву отказа.

2. После однозначного определения нежелательного события необходимо провести анализ всех событий, которые оказывают влияние на нежелательное событие и которые имеют вероятность больше 0. Зачастую можно получить события с 0 вероятностью просто потому, чтобы определить

точное значение вероятности затратно по времени и деньгам, так что проще такую вероятность не определять. Применение компьютерного моделирования позволяет снизить подобные затраты и получить требуемое численное значение вероятности. Использование системного анализа позволяет облегчить понимание всей системы. Разработчики таких систем понимают их устройство, что позволяет им досконально понимать причины, которые вызывают рассматриваемые нежелательные события, а также события, которые оказывают на них влияние. Далее для конкретного события производится нумерация причин, затем их группировка в порядке возникновения. На основании этой группировки изображается дерево отказов.

3. Далее производится непосредственное построение дерева отказов. После тщательного выбора и анализа нежелательного события и системы, которая его вызывает и влияет на него, определяются вероятности эффектов. На основании вышеперечисленного происходит построение дерева отказов. Оно основано на описанных ранее символах «И» и «ИЛИ», которые определяют основные характеристики данного дерева неисправностей.

4. Далее необходимо произвести оценку и анализ дерева отказов с целью поиска возможностей дальнейшего улучшения системы, если это необходимо. Данный этап является подготовкой к последнему шагу, на котором будет произведена идентификация опасности. Другими словами, на предпоследнем этапе происходит окончательное определение рисков, которые могут повлиять на работоспособность системы.

5. Далее следует шаг, на котором происходит контроль выявления рисков. На основании данного шага, после определения и контроля за возникающими рисками будет происходить процесс, направленный на уменьшение вероятности возникновения определенных ранее рисков.

Помимо ФТА используют так называемый анализ Маркова (еще называют блок-схема надежности или диаграмма зависимости).

В параграфе 1.1 были выделены 12 критериев надёжности сайта.

Построим дерево неисправностей для некоторого исследуемого сайта, изображенное на рисунке 1.4. Нарушение каждого критерия будем обозначать цифрой, означающий его порядковый номер в списке.

Для нарушения критерия 1 возможны ошибки в коде сайта (критерий 1.1) или недостаточная мощность оборудования (1.2). Это свидетельствует о том, что сайт недостаточно протестирован (обозначение 1.3).

Для нарушения критерия 2 возможны ошибки при назначении прав администраторам сайта (критерий 2.1) или ошибки административной панели (критерий 2.2). Это показывает, что нарушена работа административной панели сайта (обозначение 2.3).

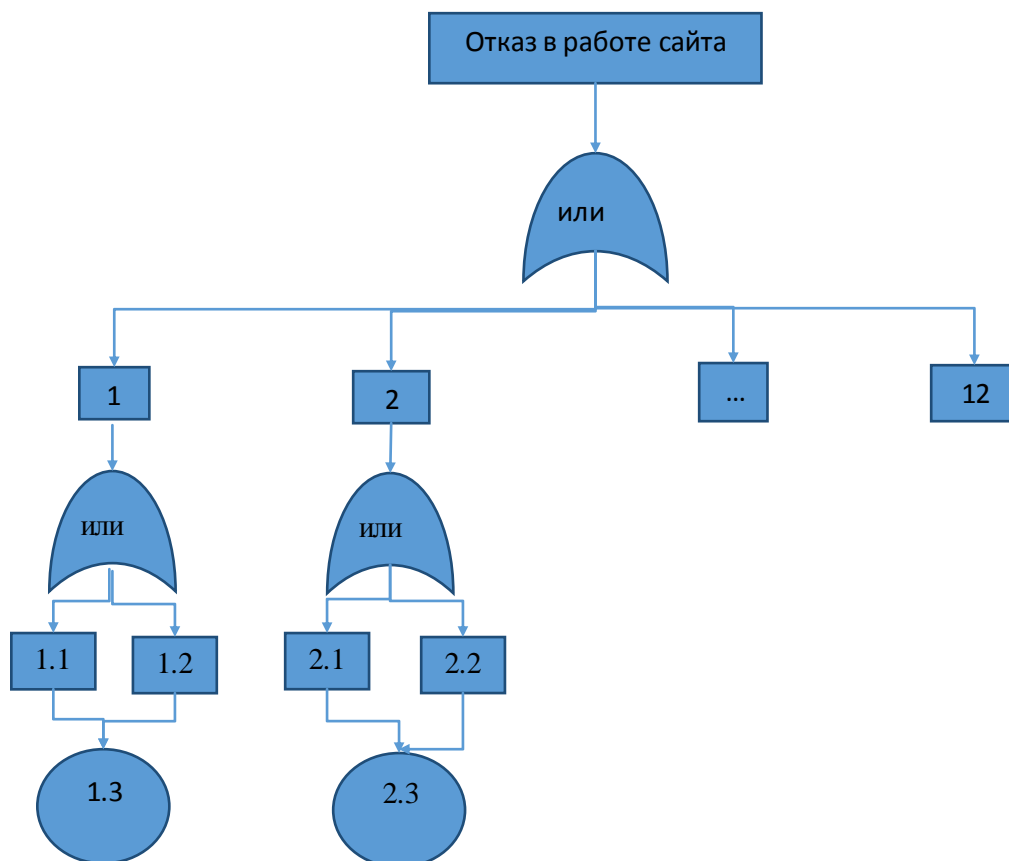


Рисунок 1.4 – Дерево неисправности для веб-сайта

После построения дерева неисправностей для сайта очевидно, что данный метод позволяет оценить надёжность сайта лишь на качественном уровне. Он

может быть использован для анализа причин уже выявленных неисправностей на действующем сайте, но на этапе проектирования он не может количественно вычислить вероятность безотказной работы сайта.

1.5 Анализ схем функциональной целостности (СФЦ)

СФЦ или схема функциональной целостности представляет собой универсальное графическое средство для структурного представления выбранных исследуемых качеств системных объектов. Описание аппарата данного метода (СФЦ) впервые было опубликовано в 1982 году Можяевым А.С. [35]. Использование так называемого аппарата функциональной целостности позволяет реализовать математическую логику в полном объеме в рамках «ИЛИ», «И» и «НЕ» логики. Применение такого аппарата позволяет правильно описывать разнообразные варианты структурных схем, начиная от базовых или традиционных (деревья отказов, блок-схемы и т.д.) и заканчивая некогерентными моделями структур различных качеств систем, над которыми производится анализ. Класс некогерентных систем является принципиально новым классом структурных схем.

Функциональная схема целостности содержит в своем составе следующие основные графические обозначение:

- два разных типа вершин (функциональная вершина и фиктивная вершина);
- два типа разнонаправленных ребер (два типа дуг – дизъюнктивная и конъюнктивная);
- два типа выхода дуги из вершин (инверсный выход (обратный) и прямой).

Ниже представлен перечень событий, являющиеся вершинами функций в схемах функциональной целостности:

- в период наработки технического средства на него не получают отказы в его работе;

- в период определенного для работы технического средства времени происходит отказ технического средства;
- на некоем этапе работы или управления системой принимают или не принимают определенное решение;
- на некоем этапе управления системой происходит корректное или некорректное выполнение работы оператором;
- попадание или непопадание при попытке поражения ударом соперника.

Построим схему функциональной целостности дерева неисправностей, построенном в параграфе 1.4.

Введем следующие обозначения:

- отказ по причине нарушения стабильности безотказной работы сразу после установки сайта, то есть при большом количестве посетителей, обозначим цифрой 1;
- отказ по причине нарушений в работе административной панели сайта обозначим цифрой 2;
- каждое событие отказа будем обозначать соответствующей цифрой его критерия.

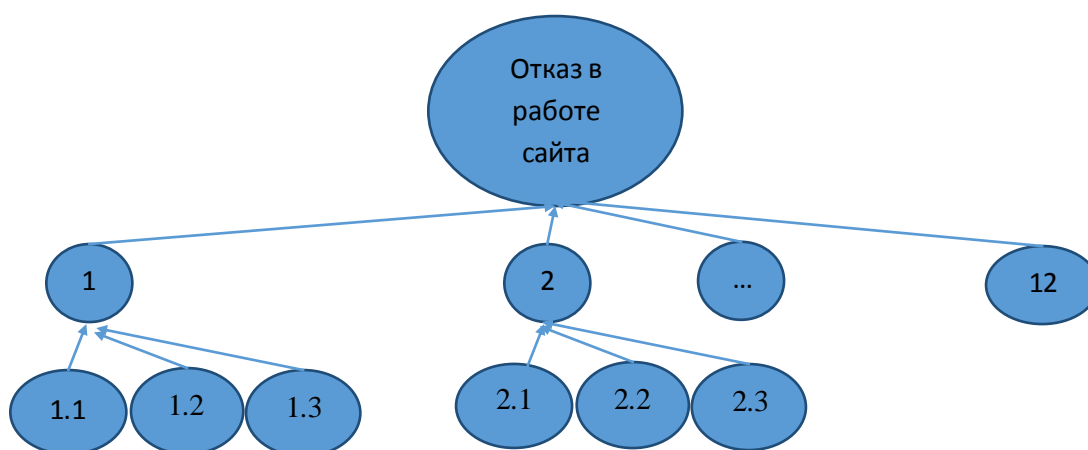


Рисунок 1.5 – СФЦ для дерева неисправностей

Как видно из схемы, изображенной на рисунке 1.5, метод анализа схем функциональной целостности достаточно полно показывает схему возникновения отказа и связи между элементами, от которых зависит возникновение отказа. Применение этого метода будет не лишним при проектировании сайта. Однако с помощью него невозможно количественно определить вероятность данного отказа, что является целью данной работы.

1.6 Вейбулл-анализ (анализ эмпирических данных из испытаний и эксплуатации)

Для моделирования данных, когда интенсивность отказов является величиной возрастающей, убывающей или постоянной, применяют так называемое распределение Вейбулла. Применение такого распределения практически неограниченно спектрами данных. Один из основных принципов такого анализа – фиксация непрерывных параметров, таких как наработка до отказов, пробег до отказа, число циклов до отказа и т.д. Такая фиксация применяется для абсолютно всех объектов. Преимущество такого анализа заключается в том, что несмотря на неполный отказ всех объектов представляется возможным выполнить моделирование распределения ресурса. Распределение Вейбулла в своих работах описывают источники [21, 22].

Рассмотрим универсальный закон (при определенных значениях параметров он трансформируется в привычные виды распределения, а именно нормальное, экспоненциальное и т.д.) Вейбулла-Гнеденко [34], который является двухпараметрическим распределением. Название закона, очевидно, сформировано из фамилий двух инженеров В. Вейбулла и Б.В. Гнеденко. В. Вейбулл применил такие распределения для анализа результатов усталостных испытаний. Он пытался произвести описание разбросов усталостных прочностей стали, полученных в ходе экспериментов. Применение закона Вейбулла-Гнеденко достаточно широко распространено. Его используют при анализе ресурса подшипников, а также при оценке надежности узлов и деталей машин (автомобилей), элементов радиоэлектронных устройств. Также данное

распределение полезно при оценке надежности механизмов машин в процессе их приработки.

Примем, что распределение случайной величины X задано плотностью $f_x(x)$, которая примет в таком случае следующий вид (1.11).

$$f_x(x) = \frac{k}{\lambda} \left(\frac{x}{\lambda}\right)^{k-1} e^{-\left(\frac{x}{\lambda}\right)^k}, x \geq 0, \quad (1.11)$$

$$0, x < 0.$$

где $\lambda > 0$ – коэффициент масштаба,

$k > 0$ – коэффициент формы.

В этом случае принимают, что X распределяется по Вейбуллу (1.12).

$$X \sim W(k, \lambda), \quad (1.12)$$

где $\lambda > 0$ – коэффициент масштаба,

$k > 0$ – коэффициент формы.

Чтобы получить распределение, в котором существует зависимость интенсивности отказа от времени, необходимо принять величину X за наработку до отказа. Тогда получим следующие значения коэффициента k , который является так называемым модулем Вейбулла:

- при $k < 1$, интенсивность отказов падает с течением временем;
- при $k = 1$, интенсивность отказов падает с течением временем;
- при $k > 1$, интенсивность отказов растет с течением временем.

Модуль Вейбулла оказывает существенное влияние на вид функции плотности распределения:

- при $0 < k < 1$, плотность стремится к бесконечности;
- при $k = 1$, плотность стремится к $\frac{1}{\lambda}$;
- при $k > 1$, плотность стремится к нулю;
- при $x=0$ и $0 < k < 1$, плотность стремится к нулю с бесконечно отрицательным угловым коэффициентом;
- при $x=0$ и $1 < k < 2$, плотность стремится к бесконечности с бесконечно положительным угловым коэффициентом;

- при $x=0$ и $k > 2$, плотность распределяется с нулевым угловым коэффициентом;
- при $k=2$, плотность стремится к бесконечности с конечным положительным угловым коэффициентом;
- $k \rightarrow \infty$ и $x = \lambda$, распределение Вейбулла сводится к так называемой дельта-функции.

Запишем вид функции Вейбулла для случая, когда $x \geq 0$, и $F(x; k; \lambda) = 0$ при $x < 0$ (1.13).

$$F(x, k, \lambda) = 1 - e^{-\frac{x^k}{\lambda}}, \quad (1.13)$$

где $\lambda > 0$ – коэффициент масштаба,

$k > 0$ – коэффициент формы.

Квантиль распределения Вейбулла при $0 \leq p < 1$ равен (1.14).

$$Q(p, k, \lambda) = \lambda * -\ln(1 - p)^{\frac{1}{k}}, \quad (1.14)$$

где $\lambda > 0$ – коэффициент масштаба,

$k > 0$ – коэффициент формы.

Интенсивность отказов h равна (1.15).

$$h(x, k, \lambda) = \frac{k}{\lambda} * \frac{x^{k-1}}{\lambda}, \quad (1.15)$$

где $\lambda > 0$ – коэффициент масштаба,

$k > 0$ – коэффициент формы.

Производящая функция моментов логарифма случайной величины, имеющей распределение Вейбулла имеет вид (1.16).

$$E e^{t \log X} = \lambda^t \Gamma\left(\frac{t}{k} + 1\right), \quad (1.16)$$

где Γ – это гамма-функция.

$\lambda > 0$ – коэффициент масштаба,

$k > 0$ – коэффициент формы.

Аналогично, характеристическая функция логарифма X задаётся формулой (1.17).

$$E e^{it \log X} = \lambda^{it} \Gamma\left(\frac{it}{k} + 1\right) \quad (1.17)$$

Моменты случайной величины X , имеющей распределение Вейбулла имеют вид (1.18).

$$E X^n = \lambda^n * \Gamma \left(1 + \frac{n}{k} \right), \quad (1.18)$$

где Γ — гамма-функция,
 $\lambda > 0$ — коэффициент масштаба,
 $k > 0$ — коэффициент формы.

Откуда

$$E X = \lambda * \Gamma \left(1 + \frac{1}{k} \right), \quad (1.19)$$

$$D X = \lambda^2 \Gamma \left(1 + \frac{2}{k} \right) - \Gamma^2 \left(1 + \frac{1}{k} \right), \quad (1.20)$$

Коэффициент асимметрии задаётся функцией (1.21).

$$\gamma_1 = \frac{\Gamma \left(1 + \frac{3}{k} \right) \lambda^3 - 3\mu\sigma^2 - \mu^3}{\sigma^3}, \quad (1.21)$$

где μ^3 — третий центральный момент,
 σ — стандартное отклонение E ,
 $\lambda > 0$ — коэффициент масштаба,
 $k > 0$ — коэффициент формы.

Коэффициент эксцесса вычисляется по (1.22).

$$\gamma_2 = \frac{-6\Gamma_1^4 + 12\Gamma_1^2\Gamma_2 - 3\Gamma_2^2 - 4\Gamma_1\Gamma_3 + \Gamma_4}{\Gamma_2 - \Gamma_1^2} \quad (1.22)$$

где $\Gamma_i = 1 + \frac{i}{k}$, так же может быть записан в виде (1.23).

$$\gamma_2 = \frac{\lambda^4 \Gamma \left(1 + \frac{4}{k} \right) - 4\gamma_1 \sigma^3 \mu - 6\mu^2 \sigma^2 - \mu^4}{\sigma^4} - 3 \quad (1.23)$$

Для 2-х параметрического распределения Вейбулла функция надёжности Вейбулла имеет вид (1.24) или (1.25).

$$R(t|T) = \frac{R(T+t)}{R(T)} = \frac{e^{-\frac{T+t}{\lambda}^k}}{e^{-\frac{T}{\lambda}^k}} \quad (1.24)$$

$$R(t|T) = e^{-\frac{T+t}{\lambda}^k + \frac{T}{\lambda}^k} \quad (1.25)$$

Для 3-х параметрического распределения данная функция имеет вид (1.26).

$$R(t|T) = \frac{R(T+t)}{R(T)} = \frac{e^{-\frac{T+t-\theta}{\lambda}^k}}{e^{-\frac{T-\theta}{\lambda}^k}} \quad (1.26)$$

Так как в этом случае функция определяет вероятность работы объекта еще t времени после того как она уже проработала T , то такая функция называется условной.

Далее можно представить распределения Вейбулла при помощи графика Вейбулла (типа Q-Q), изображенного на рисунке 1.6, который является графиком выборочной функции распределения с определенными ранее специальными осями: $-\ln(-\ln(1 - F(x)))$ и $\ln(x)$.

График Вейбулла представляет собой графический метод для определения того, относится ли набор данных к популяции, которая логически соответствовала бы двухпараметрическому распределению Вейбулла (предполагается, что начальная точка равна нулю).

График Вейбулла имеет специальные масштабы, которые разработаны таким образом, что, если данные действительно соответствуют распределению Вейбулла, точки будут линейными (или почти линейными).

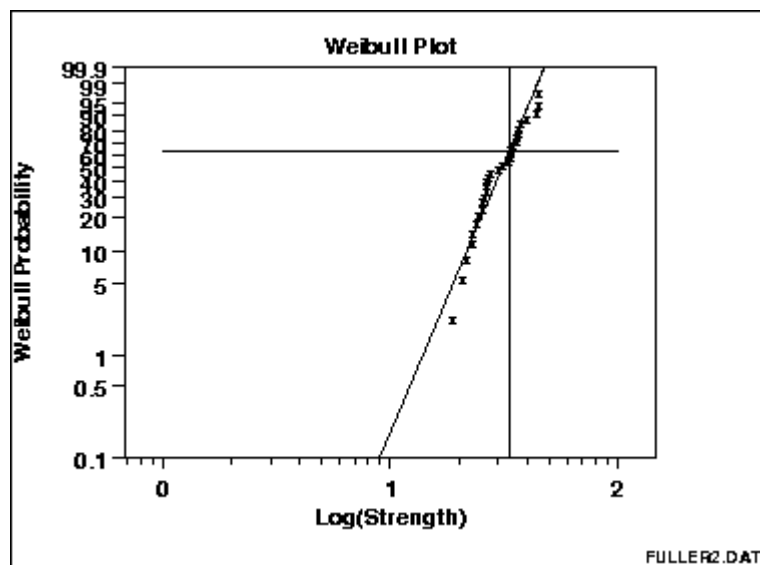


Рисунок 1.6 – Пример графика Вейбулла

Замена переменных в этом случае обусловлена желанием представить выборочную функцию распределения Вейбулла в виде линейной зависимости (1.27).

$$\begin{aligned}
 F(x) &= 1 - e^{-\frac{x^k}{\lambda}} \\
 -\ln(1 - F(x)) &= \frac{x^k}{\lambda} \\
 \ln(-\ln(1 - F(x))) &= k \ln x - k \ln \lambda
 \end{aligned} \tag{1.27}$$

y $\ln x$ c

Таким образом, при получении данных из такого распределения Вейбулла на графике отобразится прямая линия, так как зависимость линейная.

Существует несколько методик нахождения выборочной функции распределения на основании полученных данных. Например, существует метод, позволяющий с использованием таких величин как ранг данных точки «i» и количество общих точек «n», получать численное значение вертикальную координату любой точки: $F = \frac{i-0,3}{n+0,4}$.

Рассмотрим основные области применения функции Вейбулла:

- для анализа выживаемости объекта;
- в теоретических исследованиях надежности и анализе причин отказов;
- в электротехнике для моделирования величины напряжения, которое возникает в электрических цепях;
- в инженерной промышленности;
- в теоретической математике и информатике, приложенных к теории экспериментальных значений;
- в области ветроэнергетике для прогнозирования погоды, в частности распределения скорости ветра;
- в системах радиолокации при определении распределения уровня принимаемых сигналов;

- в коммуникациях беспроводных соединений при проблемах с замиранием сигнала;
- при формировании прогнозов грядущих технологических изменений;
- в области гидрологии при моделировании вероятности экстремальных событий;
- при размельчении, помоле и дроблении помогает описывать размеры получаемых частиц;
- в качестве альтернативы электронным таблицам со статистической информацией в случае, когда таблицы не дают получить требуемую точность построения графиков статистики.

Данный метод позволяет наиболее полно оценить надёжность системы. С помощью вейбулл-анализа представляется возможным вычисление значения вероятности отказа в работе сайта и проведение анализ надёжности сайта. Однако, поскольку он основан на экспериментальных данных, использование данного метода оказывается невозможным на этапе проектирования сайта.

В данной главе были проанализированы основные методы для оценки надёжности сайта. Каждый из этих методов можно применить для практической оценки надёжности веб-сайта, однако их нельзя применить для оценки надёжности сайта на этапе его проектирования и нахождения численного значения вероятности безотказной работы сайта, поскольку все они основываются на эмпирических данных.

Глава 2 МАТЕМАТИЧЕСКАЯ МОДЕЛЬ ОЦЕНКИ НАДЁЖНОСТИ ВЕБ-САЙТОВ

2.1 Постановка задачи для модели оценки надёжности веб-сайтов

На надёжность веб-сайта оказывает сильное влияние текущая конъюнктура рынка: активность конкурентов, качество и количество сайтов-конкурентов, спрос на конкретные товары и услуги, сезонность и т.п.

Вероятность безотказной работы является одним из основных показателей при расчетах на надёжность веб-сайта.

Под вероятностью безотказной работы будем понимать вероятность того, что данный объект будет сохранять свои параметры в течение некоторого определённого промежутка времени в заданных пределах и при определённых условиях функционирования.

Статистическую оценку вероятности безотказной работы $P(t)$ для массовых объектов можно получить, если обработать результаты испытаний на надёжность достаточно больших выборок. Метод нахождения оценки зависит от плана испытаний.

Под термином наработка на отказ подразумевают, средний промежуток времени между отказами, произошедшими последовательно, причем отказавший элемент был обязательно восстановлен. Расчет надёжности (безотказной работы) любого сайта на этапе его проектирования обычно учитывает высокий уровень активных пользователей на нем [8,16]. Также необходимо понимать, что происходящие на веб-сайте процессы являются случайными величинами. Следовательно, применение СМО (систем массового обслуживания) для анализа временных параметров веб-сайта возможно осуществить. По этой же причине возможно применение определенных методов моделирования.

Для проведения анализа и получения числового значения вероятности безотказной работы веб-сайта необходимо обладать такой информацией как максимально возможный поток его посетителей [9] при неком определенном времени задержки среднего пользователя на странице веб-сайта. Также

необходимо понимать какое число пользователей может одновременно посетить веб-сайт. Далее необходимо провести анализ зависимости между описанными выше величинами, т.е. какая между ними существует корреляция (функциональная зависимость).

Использование СМО с применением теории массового обслуживания [53, 55] открывает пути к проведению анализа разных моделей функционирования различных систем.

Одной из основных характеристик ТМО является простейший поток, который настолько же основополагающий в ней, насколько важен закон нормального распределения в теории вероятности. Его основной особенностью является то, что если сложить любое число несвязанных между собой простых потоков, то произойдет их сложение и появление нового суммарного потока, структура которого будет не сильно отличаться от структуры простейшего потока.

Каждое событие определяется моментом времени, когда это событие случается – t . Символом T обозначается интервал времени между двумя описанными выше моментами времени. Поток событий является независимо цепочкой моментов t .

Ниже перечислены основные показатели эффективности систем массового обслуживания с отказами:

A – абсолютную пропускную способность системы массового обслуживания, т.е. среднее число заявок, обслуживаемых за единицу времени,

Q – относительную пропускную способность, т.е. среднюю долю пришедших заявок, обслуживаемых системой,

$P_{\text{отк}}$ – вероятность отказа, т.е. вероятность того, что заявка покинет систему массового обслуживания необслуженной,

k – среднее число занятых каналов (только для многоканальной системы).

Любая страница веб-сайта, который проектируют, может быть представлена в виде математической модели СМО, которая представляет некую задержку (время присутствия пользователя на странице веб-сайта), а модель

всего веб-сайта можно представить в виде открытой сети массового обслуживания.

Постановка задачи для предлагаемой модели имеет вид:

Дано:

M – число страниц веб-сайта.

Параметры страницы:

Ts_i – среднее время пребывания пользователя на странице i , $i = 1 \dots M$;

λ_{0_i} – интенсивность входного потока заявок к странице (заходов на страницу извне, то есть не с других страниц сайта) i , $i = 1 \dots M$;

P_{ij} , $i = 0 \dots M$ – матрица вероятностей перехода между страницами сайта. Под узлом 0 понимается внешний источник заявок;

P_m – доля посетителей сайта, для которой вычисляется максимальное время задержки.

Найти:

Tq_0 – среднее время задержки пользователя на сайте;

Tq_{m_0} – максимальное время задержки пользователя на сайте, то есть время, превышение которого возможно только для доли посетителей сайта P_m (обычно в качестве значения P_m выбирают стандартное значение – 0,9 или 0,95);

Tq_i , $i = 1 \dots M$ – среднее время задержки пользователя на странице i ;

P_{li} , $i = 1 \dots M$ – вероятность отказа в обслуживании (невозможности захода на страницу i).

2.2 Составление математической модели оценки надёжности веб-сайтов

Переход на страницы сайта возможен тремя путями: набором адреса в адресной строке браузера, щелчком по гиперссылке на другом сайте, щелчком по гиперссылке на другой странице данного сайта [58][60]. Интенсивность λ_{0_i} определяет число входов на страницу в единицу времени первыми двумя

способами, а матрица вероятностей перехода P_{ij} определяет алгоритм перехода по страницам сайта третьим способом.

Современные методы моделирования позволяют [57] учесть возможность неэкспоненциального распределения входных потоков и времени задержки пользователя на страницах сайта. В этом случае в качестве входных параметров добавляются соответствующие квадраты коэффициентов вариации, а в качестве результатов появляется возможность получить стандартные отклонения времени посещения сайта в целом и каждой его страницы в отдельности.

Затем необходимо определить класс систем массового обслуживания (СМО), подходящий для адекватного представления процессов, имеющих место на страницах сайта [14]. Поскольку на каждой странице одновременно могут находиться несколько пользователей, рассмотрим класс многолинейных систем массового обслуживания с потерями (типа G/G/l/0).

Число каналов обслуживания l будет определять максимальное число одновременно находящихся на странице сайта пользователей.

Выбор данного класса СМО обусловлен следующим [63]:

- каждый сайт имеет ограничение по числу одновременно находящихся пользователей (это число определяется возможностями используемой СУБД и характеристиками веб-сервера и его следует максимизировать, но в любом случае оно конечно);
- если число пользователей превышает максимально возможное значение, возможен отказ;
- в посещении веб-страницы;
- очередь к веб-страницам отсутствует.

При проектировании сайта [7] возможно исследование его характеристик в широком диапазоне изменений параметров, в том числе параметра l . Это впоследствии поможет оценить технические характеристики оборудования, используемого в качестве веб-сервера [61], что, в свою очередь, влияет на стоимостные показатели надёжности.

Максимальное время пребывания в системе вычисляется с помощью решения трансцендентных уравнений (2.1), если квадрат коэффициента вариации потока заявок на выходе системы больше или равен 1, и (2.2), если квадрат коэффициента вариации потока заявок на выходе системы меньше 1.

$$a_1 * e^{-\mu_1 T_{qm}} + a_2 * e^{-\mu_2 T_{qm}} - 1 - P_m = 0, \quad (2.1)$$

$$\sum_{k=0}^{n_1-2} \frac{\lambda_2}{\lambda_2 - \lambda_1} \frac{\lambda_2^{n_1-k-1}}{\lambda_2 - \lambda_1} * \sum_{j=0}^k \frac{\lambda_2^{*j} T_{qm}^j}{j!} - 1 - P_m = 0, \quad (2.2)$$

$$a_1 = \frac{1}{2} \left(1 - \frac{C_{tq}^4 - 1}{C_{tq}^2 - 1} \right), \quad (2.3)$$

$$a_2 = 1 - a_1, \quad (2.4)$$

$$\mu_1 = \frac{2 * a_1}{T_q}, \quad (2.5)$$

$$\mu_2 = \frac{2 * (1 - a_1)}{T_q}, \quad (2.6)$$

$$\lambda_1 = \frac{n_1}{T_q} * \left(1 + \frac{1}{n_1 - 1 * C_{tq}^2 * n_1 - 1} \right)^{-1}, \quad (2.7)$$

$$\lambda_2 = \frac{n_1}{T_q} * \left(1 - \frac{(C_{tq}^2 * n_1 - 1)^{-1}}{(n_1 - 1)} \right), \quad (2.8)$$

$$n_1 = \frac{1}{C_{tq}^2}. \quad (2.9)$$

где C_{tq}^2 – квадрат коэффициента вариации потока заявок на выходе системы,

T_q – время задержки пользователя в СМО,

P_m – доля посетителей сайта, для которой вычисляется максимальное время задержки,

a_1, μ_1, a_2, μ_2 – параметры гиперэкспоненциального распределения,

$\lambda_1, \lambda_2, n_1$ – параметры распределения эрланговского типа.

Тогда вероятность отказа в обслуживании вычисляется по формуле:

$$P_1 = \frac{1}{C_{tq}^2} * I^{T_{qm}} * \lambda * (e^{-\lambda})^{\frac{1}{2}}, \quad (2.10)$$

где P_l – вероятность отказа в обслуживании (невозможности захода на страницу),

C_{tq}^2 – квадрат коэффициента вариации потока заявок на выходе системы,

l – максимально возможное число пользователей,

T_{qm} – максимальное время пребывания в системе,

λ – интенсивность входного потока заявок.

Далее с помощью данной модели рассчитаем вероятность безотказной работы сайта.

Глава 3 РЕАЛИЗАЦИЯ И ПРИМЕНЕНИЕ МОДЕЛИ ОЦЕНКИ НАДЁЖНОСТИ

3.1 Анализ программного обеспечения для реализации математической модели

Без применения специального программного обеспечения нельзя представить себе применение методов математического моделирования. Компьютерная алгебра (или символьная или компьютерная математика) представляет собой важную часть математического моделирования. Очевидно, что подобное ПО можно относить к инженерным программам для автоматизированного проектирования. Среди подобных программ выделяют три главных раздела:

- компьютерное проектирование (CAD),
- автоматизация производства (CAM),
- компьютерная инженерия (CAE).

Данные программы помогают пользователю не допустить большое количество ошибок, а также уменьшить временные затраты во время вычислений.

Нельзя не заметить, что эти системы решают множество задач в разных сферах:

- анализ и разработка различного рода алгоритмов,
- обработка и анализ данных,
- исследования в математике, которые требуют аналитических выкладок и вычислений,
- компьютерный эксперимент,
- инженерная и научная графика,
- математическое моделирование,
- визуализация данных,
- разработка приложений, связанных с расчетами или графикой.

Самым важным программным обеспечением в этой сфере выступают:

1. «Maple» является продуктом компании Waterloo Maple Software, Inc., который называют системой компьютерной алгебры или системой символьных вычислений. Данный продукт даёт возможность производить как численные, так и аналитические расчеты, в которых можно редактировать текст и формулы на рабочем листе. Поскольку в этой среде есть возможности для представления формул в полиграфическом формате, используется двух- и трехмерная графика хорошего качества, а также присутствует анимация, «Maple» одновременно является еще и мощным графическим редактором научного рода. Эффективным средством для создания новых алгоритмов его делают эффективный и в то же время простой язык-интерпретатор, возможность преобразования кода «Maple» в код C, а также открытая архитектура. Благодаря широкому функционалу, понятному интерфейсу и простым правилам работы этот пакет завоевал большую популярность также и у отечественных инженеров и математиков.

2. Система «Mathematica», разработанная компанией Wolfram Research, Inc., имеет чрезвычайно широкий набор средств, переводящих сложные математические алгоритмы в программы. Практически все содержащиеся в курсе технического вуза по высшей математике алгоритмы заложены в ядро инструментальной среды «Mathematica». В некоторых странах, таких как США, система высшего образования неразрывно связана с этим продуктом. Большое преимущество системы «Mathematica» заключается в том, что её способы записи алгоритмов и операторы естественны и просты. В этой системе имеется мощный графический пакет, который помогает строить даже очень сложные графики функций одной и двух переменных. Важным преимуществом среды «Mathematica», которое делает её одним из лидеров среди других систем высокого уровня, является то, что эта система столь широко распространена по всему миру, охватывая широкие области применения в инженерных и научных исследованиях, а также в образовательной сфере.

3. Инструмент «Macysma», созданный компанией Macysma, Inc. является одним из первых математических программных инструментов, которые оперируют символьной математикой. Сильной стороной этого пакета является достаточно развитый аппарат дифференциальных уравнений и линейной алгебры. Эта инструментальная сторона предназначена для прикладных расчетов и не ориентирована на теоретические исследования в математической области. Именно поэтому в программе отсутствуют или частично сокращены разделы, которые связаны с теоретическими методами, такими как теория групп, теория чисел и другими. Важным плюсом «Macysma», выделяющим это ПО перед другими, является то, что пользователю предоставляется возможность аналитически и численно проводить расчеты большого числа задач, таких как решение различных типов уравнений в частных производных. Данная инструментальная среда отличается от других тем, что она имеет возможность интегрироваться с Microsoft Word. В библиотечных файлах этой системы автоматически загружается большое количество команд. Также пользователи выделяют, что браузер этого ПО достаточно удобен для просмотра. Важной особенностью «Macysma» является то, что эта система генерирует код на таких языках программирования как FORTRAN и C, включая управляющие операторы. «Macysma» работает исключительно под OS Windows на платформе Intel.

4. Инструментальная среда «S-PLUS» является интерактивной компьютерной средой, которая обеспечивает пользователю полнофункциональный графический анализ данных. Также необходимо отметить, что она включает в себя необычный объектно-ориентированный язык. «S-PLUS» представляет собой довольно гибкую систему. Она используется учеными по всему миру для математических расчетов, исследовательского анализа, статистического анализа и достаточно удобного графического представления данных. Достоинством «S-PLUS» можно назвать несомненно мощные вычислительные возможности, простоту использования статистических методов современности, большую функциональность,

понятный интерфейс, и, что важно, расширяемый набор статистических методов.

«Maple», «Mathematica» и «MathCad» являются, пожалуй, самыми популярными представителями этой области. Далее приведем анализ этих средств для компьютерного моделирования.

3.2 Анализ инструментальной среды «Maple»

Система компьютерной математики «Maple» является одной из самых важных в своем роде и по настоящий момент показывает себя одним из лидеров среди универсальных систем компьютерной математики. Данная система, пользующаяся большой популярностью в профессиональном сообществе, включает в себя удобную интеллектуальную среду для математических исследований разного уровня. Нельзя оставить без внимания тот фактор, что символьный анализатор «Maple» принято считать самой сильной частью этой системы, именно по этой причине он и был позаимствован в несколько другие САЕ-пакеты, а именно в Matlab и MathCad, а также стал частью программного обеспечения для подготовки научных публикаций, таких как WorkPlace и Math Office for Word.

Первоначально система «Maple» была доступна только узкому кругу профессиональных математиков и была разработана Высшей технической школой (ETHZ, Цюрих, Швейцария) совместно с государственным университетом Ватерлоо (шт. Онтарио, Канада), именно поэтому она ориентирована в первую очередь на научных работников.

Данная инструментальная среда главным образом предназначается для проведения символьных (аналитических) вычислений и для этого в ней имеется один из самых мощных в своем классе набор специализированных функций и процедур, а именно более 3000. Для большинства пользователей, столкнувшихся с необходимостью проведения средних по сложности математических расчетов, такая комплектация является избыточной, и пользователи начинают в ней путаться. Некоторые из встроенных в «MathCad»

(до версии 13.1 включительно) возможностей основаны на подмножестве системы компьютерной алгебры «Maple» и были случаи, когда пользователи могли получить доступ по всем функциям и процедурам «Maple».

Как уже сказано выше, инструментальная среда «Maple» ориентирована только на профессиональных пользователей, это находит выражение в том, что решение задач в данной среде требует не только умения оперировать той или иной функцией, но также и знания методов решения, которые заложены в эту функцию: во многих встроенных функциях этого пакета фигурирует аргумент, который задает метод решения. Данный аспект работы этого ПО затрудняет его использования для людей, которые впервые столкнулись с этой инструментальной средой.

В среде «Maple» пользователь может с использованием других систем, а также языков программирования высокого уровня, создавать интегрированные среды. Когда пользователь закончил расчеты и ему необходимо проиллюстрировать публикацию, то он может использовать эту среду для подготовки иллюстраций и визуализации данных. Прямо в данном программном обеспечении можно подготовить статью, отчет или даже книгу (любой печатный материал), а после можно переходить к другому исследованию. Данная инструментальная среда предоставляет пользователю интерактивный интерфейс, когда он вводит команды и тут же в реальном времени ему предоставляются результаты выполнения этих команд.

Это является существенным плюсом этого пакета, поскольку в таком режиме достаточно удобно производить корректировку математической модели.

Но данное программное обеспечение не проявляет признаков сходства с привычными средами программирования, принуждающими пользователя строго формализовать все переменные и действия с ними. «Maple» автоматически подбирает тип данных и проверяет правильность выполнения операций с ними. Это означает, что в данной среде не требуется формализация записи и описание переменных

Это также будет достаточным преимуществом этого пакета: он помогает исключить «человеческий фактор» при построении модели, поскольку здесь автоматически проверяется корректность операций.

«Maple» включает в себя отлично оптимизированные процедуры на языке С (ядро), написанной на языке Maple библиотеки, а также достаточно современного интерфейса.

Библиотека включает в себя большое количество команд, которые представляют собой процедуры, выполняемые в интерпретационном режиме.

Концепция рабочего поля (или worksheet) является основополагающей для работы в данной среде. Это означает, что работа здесь проходит в документе, содержащем в себе строки ввода-вывода и даже графику.

Инструментальная среда «Maple» обеспечивает пользователю работу в виде интерпретатора. То есть, при задании команды в поле ввода, пользователь нажимает на клавишу ввода и системе сама выводит на экран результат выполнения команды (строки вывода или сообщение об ошибке). Далее появляется приглашение на ввод дальнейшей команды.

Данный пакет является достаточно мощным инструментом для реализации математической модели: в нём содержатся большие возможности визуализации, огромное количество функций, интерфейс пользователя довольно развит. Все эти перечисленные факторы оказывают значительное влияние на выбор «Maple» для реализации математической модели.

3.3 Анализ инструментальной среды «Mathematica»

Пакет «Mathematica» (текущая версия 11.3) является собственной разработкой Wolfram Research. Он получил широкое применение в научных исследованиях и является хорошо известным в научной и образовательной сферах. Бытует мнение, что в данном программном обеспечении существует некая функциональная избыточность, к примеру, там имеются возможности для синтеза звука.

Это очень мощная математическая система, которой возможно не смогут пользоваться для расчетов, к примеру, секретари или даже директора коммерческих фирм, потому что в ней содержится избыток функций, которые будут путать обычного пользователя. Является очевидным, что крупная научная лаборатория или профильная кафедра в вузе, если они заинтересованы в автоматизации расчетов, должны иметь в своем арсенале подобное программное обеспечение.

Однако, не обращая внимание на профессиональную направленность, система «Mathematica» достаточно проста в использовании и применяется даже неопытными пользователями, такими как студенты и учащиеся профильных классов в школе. Все категории пользователей могут неограниченно применять данное программное обеспечение для автоматизации своих расчетов.

Учитывая огромное количество функций этой среды, её интерфейс не перегружается, а вычисления происходят достаточно быстро. Численные расчеты и символьные преобразования в данной среде производятся с высокой скоростью. У всех программ присутствуют определённые плюсы и минусы, но пакет «Mathematica» из всех систем более универсальная и полная. И, что самое главное, у каждой программы присутствуют свои «фанаты», которые полностью уверены в превосходстве этой системы над другими. Однако люди, которые всерьез занимаются научной работой с использованием систем компьютерной алгебры, как правило используют сразу несколько систем, чтобы гарантировать высокий уровень достоверности вычислений.

Нельзя не выделить тот факт, что наряду с Wolfram Research в разработке разных версий системы участвовали различные фирмы и множество высококвалифицированных специалистов, включая, конечно же, программистов и математиков.

Данное программное обеспечение выделяют как одно из крупнейших в своей сфере. Система «Mathematica» реализует в себе самые эффективные алгоритмы для вычислений, такие как механизм контекстов, который исключает возникновение побочных эффектов в программах.

Инструментальная среда «Mathematica» на сегодняшний день является одним из мировых лидеров на ранке компьютерных систем компьютерной алгебры, используемым на ПК. Данные системы наряду с выполнением численных расчетов высокого уровня сложности с выводом результатов этих расчетов в удобном графическом виде обеспечивают проведение требующих больших затрат аналитических преобразований и вычислений.

Версии «Mathematica» для OS Windows отличаются современным интерфейсом пользователя, позволяющим работать в документах в форме записных книжек (Notebooks). Они соединяют в себе описания алгоритмов решения задач, исходные данные и результаты решения в различном виде, таком как числа, матрицы, векторы, графики, таблицы и любые математические формулы. Данные аспекты работы с этой системой необходимо учитывать при выборе программы для реализации математической модели.

Эта система изначально представляла собой ПО, которое способствует максимальной автоматизации труда математиков, аналитиков и научных работников. Именно по этой причине она заслуживает внимания как типичный представитель элитного программного обеспечения высшей степени сложности для высокоинтеллектуальных расчетов.

Однако эта система является также и гибким инструментарием математического типа, являющимся в добавок ко всему достаточно мощным. «Mathematica» в силу своей направленности оказывает огромную помощь многим научным работникам, преподавателям высших учебных заведений, инженерам, а также школьникам профильных классов и школ.

В данной инструментальной среде огромную роль играет графика, включая динамическую. Также здесь имеются возможности для синтеза звуков и динамической анимации (мультимедиа). Достаточно широким является набор функций, относящихся к графике, а также изменяющих их действие. С самого создания этой системы графика была сильным преимуществом «Mathematica» перед другими представителями программ символьной математики. Однако функции графики работа с мультимедиа не является важным критерием для

выбора системы для построения математической модели, поскольку для неё будет достаточно построения довольно простых графиков.

Данное программное обеспечение стремительно заняло ведущую роль на рынке программ подобного функционала. Исключительным преимуществом этой системы всегда являлось то, что она предоставляет пользователю обширные графические возможности, и конечно же реализация интерфейса вида записная книжка. В добавок ко всем преимуществам, это программное обеспечение обеспечивает динамическую связь между ячейками документов типа Excel (электронных таблиц) при решении символьных задач. Это свойство являлось отличительным для «Mathematica».

Главную роль в системе играет машинно-независимое ядро математических операций, дающее возможность для переноса системы на другую платформу. В системе «Mathematica» используется программный интерфейс процессов Front End, который позволяет переносить систему на другую платформу без потерь. Интерфейсные процессоры систем для других платформ обладают своими особенностями и именно они определяют, какой вид имеет пользовательский интерфейс. Для быстрого вызывания из ядра функций оно сделано достаточно компактным.

В данной системе присутствует библиотека (Library) и набор пакетов расширения (Add-on Packages) для увеличения количества функций. Пакеты расширений создаются на собственном языке программирования «Mathematica», также они представляют собой важнейшее средство для увеличения возможностей системы и их приспособлению к решению определенных типов задач пользователя. Кроме всего перечисленного, «Mathematica» имеет встроенную электронную справочную систему, которая называется Help и включает в себя книги с практическими примерами для работы. Такое обилие справочной информации несомненно пригодится при построении математической модели в этой системе, однако стоит заметить, что расширения, предлагаемые для «Mathematica» все платные и порой набор расширений может стоить гораздо больше самой системы.

То есть данный пакет, с одной стороны является типичной средой программирования, составленной на базе одного из мощнейших ориентированных на проблему языков функционального программирования высокого уровня. Однако эта же система представляет собой средство для решения большого числа математических задач в режиме диалога без привычного программирования.

Из проведенного анализа можно сделать вывод, что система «Mathematica», если рассматривать её как среду программирования, имеет массу возможностей для организации ввода-вывода, создания и разработки управляющих структур, работы с системными функциями и обслуживании разных периферийных устройств. Также, при использовании Add-ons (пакетов расширений) система подстраивается под запросы конкретного пользователя, начиная от типичного пользователя, которому не нужны мощные возможности, заканчивая профессионалами своей области, которым необходимы огромные возможности этой программы.

Для случаев, когда данная система используется для математического моделирования, она предоставляет пользователю весь свой обширный функционал. Однако существенным усложнением в ней является довольно необычный язык программирования. Большим плюсом является наличие дополнительных расширений, но как правило все они являются платными, что влечет за собой удорожание системы для пользователя.

3.4 Анализ инструментальной среды «MathCad»

Одним из пакетов для построения математической модели является пакет «MathCad» [5][6].

Он представляет собой систему символьной математики, которая ориентирована на создание интерактивных документов с визуальной составляющей и вычислениями, входящую в класс систем автоматизированного проектирования. Отличительной чертой «MathCad» является простота применения для совместной работы в коллективе.

Эта система была создана соучредителем компании Mathsoft, в настоящее время являющейся частью корпорации Parametric Technology Corporation (PTC). В настоящее время «MathCad» занимает лидирующее место в своем классе.

Эта инструментальная среда отличается интуитивно понятным интерфейсом, который является простым для использования. Что является очень удобным, для ввода данных есть возможность использовать не только клавиатуру, но и специальную панель инструментов.

До версии 13.1 (включительно) некоторые возможности этой системы основывались на подмножестве рассмотренной в параграфе 3.2 системы символьной математики «Maple». Далее (с версии 14) применяется символьное ядро MuPAD.

В отличие от текстовой записи на языках программирования, в «MathCad» работа проходит на рабочем листе, на котором автоматически показываются выражения и уравнения.

В данной программе используется принцип «что видишь, то и получаешь» (What You See Is What You Get), который сокращенно называется WYSIWYG. Он является очень удобным, когда необходимо построить компьютерную модель.

Инструментальная среда «MathCad» создана для непрофессиональных пользователей, не программистов. Но она же находит применение для визуализации результатов математического моделирования, используя распределённые вычисления и языки программирования в задачах, отличающихся высокой сложностью.

Именно поэтому данная система вполне применима к цели диссертации. В добавок, стоит отметить, что «Mathcad» может использоваться в крупных инженерных проектах, в которых важную роль играет трассируемость и соответствие стандартам.

Также «Mathcad» достаточно широко используется для обучения [3].

В данной системе присутствует поддержка технологий XML и .NET и в сочетании с открытой архитектурой она даёт возможность легко интегрироваться в разнообразные IT-структуры и приложения для инженеров.

«Mathcad» применяют более двух миллионов человек в разных странах. Это объясняет большое количество примеров применения данной системы широкую изученность проблем и ошибок в системе.

Данное программное обеспечение включает в себя множество встроенных функций и операторов, направленных на решение технических задач. «Mathcad» даёт пользователю возможность проводить символьные и численные расчеты, производить операции со скалярными величинами, матрицами и векторами, а также автоматически оперировать единицами измерения.

«Mathcad» имеет достаточно много возможностей, таких как:

- решение дифференциальных уравнений, в том числе и численными методами;
- построение двумерных и трёхмерных графиков функций (в разных системах координат, контурные, векторные и т. д.);
- использование греческого алфавита как в уравнениях, так и в тексте;
- выполнение вычислений в символьном режиме;
- выполнение операций с векторами и матрицами;
- символьное решение систем уравнений;
- аппроксимация кривых;
- выполнение подпрограмм;
- поиск корней многочленов и функций;
- проведение статистических расчётов и работа с распределением вероятностей;
- поиск собственных чисел и векторов;
- вычисления с единицами измерения;

- интеграция с САПР-системами, использование результатов вычислений в качестве управляющих параметров.

Используя эту систему, пользователь имеет возможность документировать каждый этап решения задачи в процессе её решения.

Важным преимуществом «Mathcad» перед программами такой же направленности является не текстовый режим ввода, а графический: для ввода данных есть возможность применять не только клавиатуру, но и кнопки на различных панелях инструментов.

И, что более важно, какой способ ввода не применял бы пользователь, данные будут иметь одинаковый привычный вид. Это означает, что пользователю не требуется особой подготовки для ввода данных. Вычисления с использованием введенных пользователем данных могут проводиться в трех видах, в зависимости от выбора самого пользователя:

- мгновенно,
- по команде,
- вместе с набором данных.

В данной среде формулы вычисляются подобно чтению текста: слева направо и сверху вниз. Результаты всех изменяемых данных, включая параметры, формулы и переменные можно видеть в режиме реального времени. Эта особенность является важной частью интерактивных документов для вычисления.

В программах подобного назначения, таких как «Maple», «Murad» и «Mathematica», присутствует режим программного интерпретатора при проведении вычислений. Это режим, который изменяет формулы, которые были введены в виде текста.

Для работы в среде «MathCad», в отличие от ориентированной на пользователей с навыками программирования «Maple», при решении простых задач вовсе не обязательно уметь программировать.

Систему «MathCad» принято относить к системам символьной математики. Это означает, что данная система является в некотором роде

средством для автоматизации математических расчетов. Среди ПО такого рода числится достаточно большое количество аналогов данной системы, различным по функциональности и принципам работы. Это такие системы как «Maple», анализируемые в параграфе 3.2, «Mathematica», проанализированная в параграфе 3.3. Очевидно, что у таких крупных систем существует много аналогов, таких как MuPAD, Scilab и Maxima и т.д. Однако, их объективное сравнение затрудняется тем, что у этих программ разное назначение, а также различная идеология использования.

Создатели «Mathcad» ориентировались на потребности пользователей, расширяя систему. Для этого назначены дополнительные библиотеки и пакеты расширения, которые можно приобрести отдельно и которые имеют дополнительные функции, встраиваемые в систему при установке, а также электронные книги с описанием методов решения специфических задач, с примерами действующих алгоритмов и документов, которые можно использовать непосредственно в собственных расчетах. Этот фактор нельзя не учесть при выборе программного обеспечения для реализации математической модели. Кроме этого, в случае необходимости и при условии наличия навыков программирования в С, есть возможность создания собственных функций и их прикрепления к ядру системы через механизм DLL.

В отличие от «Maple», «MathCad» изначально задумывался для численного решения математических задач. Данная система ориентирована на задачи не теоретической, а прикладной математики и используется, когда необходимо получить результат без углубления в математическую часть.

Однако, когда необходимы символьные вычисления, то для них и задумано интегрированное ядро «Maple» (с версии 14 – MuPAD). Наиболее полезно это бывает при необходимости создания документов для образовательного процесса, при котором нужно показать построение математической модели, основываясь на физической картине явления или процесса.

В сравнении с оригинальным ядром «Maple», в ядре «MathCad» доступно только около 300 функций, в нём присутствуют искусственные ограничения, однако этого количества как правило вполне достаточно для решения задач инженерного характера.

Более того, опытные пользователи «Mathcad» обнаружили, что в версиях до 13 включительно есть возможность не очень трудозатратным образом «включить» практически весь функциональный арсенал ядра «Maple» (так называемые «недокументированные возможности»), что приближает вычислительную мощь «Mathcad» к «Maple».

Mathcad задумывался как средство программирования без программирования, но, если возникает такая потребность — Mathcad имеет довольно простые для усвоения инструменты программирования, позволяющие, впрочем, строить весьма сложные алгоритмы, к чему прибегают, когда встроенных средств решения задачи не хватает, а также когда необходимо выполнять серийные расчёты [5]. Это также необходимо учитывать в случаях, когда речь идет о возможностях конкретных систем для построения математических моделей.

В среде Mathcad фактически нет графиков функций в математическом понимании термина, вместо них используется визуализация данных, находящихся в векторах и матрицах (то есть осуществляется построение как линий, так и поверхностей по точкам с интерполяцией), хотя пользователь может об этом и не знать, поскольку у него есть возможность использования непосредственно функций одной или двух переменных для построения графиков или поверхностей соответственно. Так или иначе, механизм визуализации «Mathcad» значительно хуже визуализации «Maple», где достаточно иметь только вид функции, чтобы построить график или поверхность любого уровня сложности. По сравнению с «Maple», графика «Mathcad» имеет такие недостатки, как: невозможность построения поверхностей, заданных параметрически, с прямоугольной областью определения двух параметров;

создание и форматирование графиков только через меню, что ограничивает возможности программного управления параметрами графики [5].

Однако следует помнить об основной области применения «Mathcad» — для задач инженерного характера и создания учебных интерактивных документов возможностей визуализации вполне достаточно.

Основным преимуществом «Mathcad» является естественный математический язык, на котором формируются решаемые задачи. Объединение текстового редактора с возможностью использования общепринятого математического языка позволяет пользователю получить готовый итоговый документ.

От других продуктов аналогичного назначения, например, Maple & Theorist (компании Waterloo Maple Software) и Mathematica (компании Wolf Research), «Mathcad» (компании Mathsoft) отличается ориентацией на создание высококачественных документов (докладов, отчетов, статей) в режиме WYSIWYG (What You See Is What You Get). Это означает, что, внося изменения, пользователь немедленно видит их результаты и в любой момент может распечатать документ. Это является важным преимуществом данного пакета, поскольку есть возможность отредактировать документ и тут же его распечатать. Это может быть очень удобно при корректировке математических моделей. Работа с пакетом за экраном компьютера практически совпадает с работой на бумаге с одной лишь разницей – она более эффективна. Преимущество «Mathcad» состоит в том, что он не только позволяет провести необходимые расчеты, но и оформить свою работу с помощью графиков, рисунков, таблиц и математических формул. А эта часть работы является наиболее рутинной и малотворческой, к тому же она и трудоёмкая и занимает достаточно много времени. Именно по этим причинам при выборе программного обеспечения для реализации математической модели оценки надёжности рассматривается именно этот пакет.

3.5 Сравнение средств для реализации математической модели

Далее проведём сравнение программных средств для реализации математической модели оценки надёжности веб-сайтов.

Сравнение будет проводиться по следующим параметрам:

- требуемые аппаратные ресурсы;
- стоимость лицензионной системы;
- производительность системы для решения задач невысокой сложности;
- возможности с учётом внешних библиотек и пакетов;
- графические возможности системы;
- интерфейс пользователя;
- возможности редактирования документов (с формулами);
- база данных для помощи пользователям (часто задаваемые вопросы, форумы пользователей).

Начнем с первого параметра – требуемые аппаратные ресурсы. В таблице указаны основные требования для работы в сравниваемых системах.

Таблица 3.1 – Сравнение системных требований пакетов «Maple», «Mathematica» и «MathCad»

Maple	<ul style="list-style-type: none">• Maple совместим с Windows начиная с Windows Vista;• DVD-ROM (для установки с DVD диска);• рекомендуется 16-битный монитор с разрешением 1024 на 768 (или выше);• должны быть включены внутренние соединения TCP / IP;• при работе ниже
-------	------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

	<p>рекомендуемой памяти может пострадать производительность системы;</p> <ul style="list-style-type: none"> • классический рабочий лист недоступен в 64-битной Windows.
Mathematica	<ul style="list-style-type: none"> • Процессор: Intel Pentium Dual-Core или эквивалентный; • Дисковое пространство: 14 GB; • Оперативная память (RAM): рекомендуется больше 2 GB; • доступ в интернет необходим для доступа к данным с инфобанка Wolfram Knowledgebase;
MathCad	<ul style="list-style-type: none"> • Mathcad совместим с Windows 8, Windows 7 (64 или 32-bit), Windows Vista (64 или 32-bit), Windows XP Professional (включая 64-bit), Windows XP Home, и Windows 2000; • процессор с тактовой частотой 700 MHz или выше; рекомендуется 2000 MHz или больше; • 512 MB RAM; • 1,75 GB свободного пространства на жестком диске: 350 Мб для самого Mathcad и

	<p>1,4 Гб для временных файлов, создаваемых во время установки;</p> <ul style="list-style-type: none"> • установленный Microsoft .NET Framework начиная с версии 4.0 и выше
--	--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

Из таблицы 3.1 можно сделать вывод, что системе «MathCad» требуется значительно меньше ресурсов, чем остальным рассматриваемым.

Следующим немаловажным параметром является стоимость лицензионной системы. Для сравнения возьмем студенческие лицензии, которые не предназначены для коммерческих исследований, и лицензии для индивидуального пользования. Далее приведём стоимость лицензий:

1. Лицензионный пакет «Maple» имеет цену в 239\$ для индивидуального пользования и 99\$ для студентов.
2. Лицензионный пакет «Mathematica» имеет цену в 320\$ для индивидуального пользования и 155\$ для студентов.
3. Лицензионный пакет «MathCad» имеет цену в 315\$ для индивидуального пользования и 97\$ для студентов.

После изучения стоимости лицензионных продуктов становится очевидным, что для коммерческих проектов «Maple» предоставляет самую низкую стоимость, а для студенческих – «MathCad».

Обе системы подходят для задач невысокой сложности, а «MathCad» и вовсе ориентирована на задачи невысокой сложности.

Пакеты «Mathematica» и «Maple» имеют большее количество библиотек и расширений, однако их стоимость может значительно превышать стоимость самой программы.

Считается, что в пакете «Mathematica» графические возможности значительно выше конкурентов.

Самый просто и понятный интерфейс у «MathCad», а возможности визуализации богаты. Все вычисления в «MathCad» осуществляются на уровне визуальной записи выражений в общепотребительной математической форме. Пакет имеет хорошие подсказки, подробную документацию и функцию обучения пользователей работе с программой.

В ходе сеанса работы с системой создаются документы, которые содержат текстовые сообщения, математические формулы, рисунки и таблицы. Их нужно постоянно редактировать с для исправления ошибок или придания документам наиболее удобного вида. Работа с этими документами лучше всего реализована в «MathCad» и «Mathematica». Основной недостаток системы «Maple» – сложность задания математических формул в их естественном виде, что давно реализовано в других системах.

Базы данных для пользователей у всех трех программ реализованы на одинаковом уровне. У «MathCad» она более обширная за счёт того, что данный пакет программ рассчитан на более широкий круг пользователей.

Из приведённого выше сравнения можно сделать вывод о целесообразности использования пакета «MathCad» для реализации математической модели оценки надёжности веб-сайтов, поскольку данный пакет имеет более простой интерфейс, низкие системные требования, лучшие возможности редактирования, что немаловажно для корректировки модели, и более низкую цену для студенческой версии.

3.6 Реализация математической модели оценки надёжности веб-сайтов

Рассмотрим одну страницу веб-сайта ($i=1$). В случае использования системы G/G/1/0 время $Tq_0 = Tq_{0i}$ и $Tq_{m_0} = Tq_{m_i}$, причём эти времена не зависят от λ_{0i} .

Пусть $Tq_1 = 3,5$ [мин] и максимально возможное число пользователей $I=47$. Исследуем поведение системы в диапазоне значений $\lambda_{0i} = [1 \div 100]$ [заявок/ мин] с шагом 1. Расчёты с использованием пакета «MathCad» [11] дают результаты, показанные в таблице 3.2.

Таблица 3.2 – Поведение системы в диапазоне значений $\lambda_{0_i} = [1 \div 100]$ [заявок/ мин] с шагом 1

λ_{0_i}	Tq_0	Tq_{m_0}	P_l
1,000	3,500	14,2200	$0,99264 * 10^{-30}$
2,000	3,500	14,2200	$0,14197 * 10^{-22}$
3,500	3,500	14,2200	$0,13386 * 10^{-17}$
4,000	3,500	14,2200	$0,84843 * 10^{-14}$
5,000	3,500	14,2200	$0,71922 * 10^{-11}$
6,000	3,500	14,2200	$0,12427 * 10^{-8}$
7,000	3,500	14,2200	$0,84931 * 10^{-7}$
8,000	3,500	14,2200	$0,24604 * 10^{-5}$
9,000	3,500	14,2200	$0,37543 * 10^{-4}$
10,000	3,500	14,2200	$0,31191 * 10^{-3}$
11,000	3,500	14,2200	$0,17752 * 10^{-2}$
12,000	3,500	14,2200	$0,61112 * 10^{-2}$
13,500	3,500	14,2200	$0,17362 * 10^{-1}$
14,000	3,500	14,2200	$0,34582 * 10^{-1}$
15,000	3,500	14,2200	$0,57721 * 10^{-1}$
16,000	3,500	14,2200	$0,86232 * 10^{-1}$

Как видно из расчетов, при данных характеристиках максимально возможный поток посетителей страниц – 16 чел./мин. Построим график зависимости вероятности отказа в обслуживании от интенсивности потока заявок к странице.

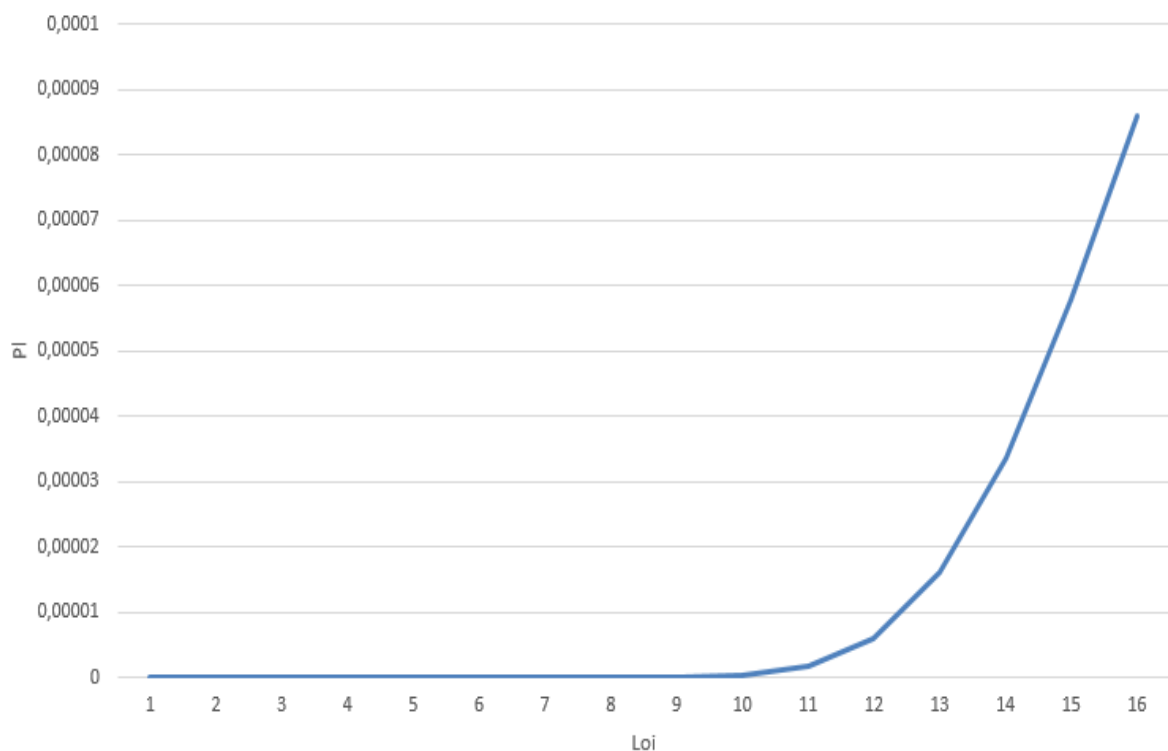


Рисунок 3.1 – Зависимость вероятности отказа в обслуживании от интенсивности потока заявок к странице

Как продемонстрировано на рисунке 3.1, вероятность отказа в доступе начинает увеличиваться, когда $\lambda_{0_i} > 10$ [заявок/мин].

Пусть $\lambda_{0_i} = 9$ [человек/мин], $Tq_1 = 3,5$ [мин]. Исследуем поведение системы в диапазоне изменения значений максимально возможного числа пользователей $l = [1 \div 49]$ с шагом 1. Расчет дает результаты, приведенные в таблице 3.3.

Таблица 3.3 – Поведение системы в диапазоне изменения значений максимально возможного числа пользователей $l = [1 \div 49]$ с шагом 1

Tq_0	Tq_{m_0}	P_l
3,500	14,2200	$0,51294 \cdot 10^{-4}$

3,500	11,9400	$0,39126*10^{-3}$
3,500	6,4373	$0,42216*10^{-3}$
3,500	5,8723	$0,68108*10^{-3}$
3,500	5,4189	$0,84661*10^{-3}$
3,500	5,0844	$0,11383*10^{-2}$
3,500	4,8378	$0,12883*10^{-2}$
3,500	4,6493	$0,12217*10^{-2}$
3,500	4,5068	$0,13936*10^{-2}$
3,500	4,3936	$0,15979*10^{-2}$
3,500	4,3017	$0,15753*10^{-2}$
3,500	4,2259	$0,16122*10^{-2}$
3,500	4,1623	$0,16773*10^{-2}$
3,500	4,1083	$0,18113*10^{-2}$
3,500	4,0618	$0,17513*10^{-2}$
3,500	4,0215	$0,18982*10^{-2}$
3,500	3,9861	$0,18169*10^{-2}$
3,500	3,9549	$0,18439*10^{-2}$
3,500	3,9271	$0,18468*10^{-2}$
3,500	3,9022	$0,1846*10^{-2}$
3,500	3,8797	$0,1678*10^{-2}$
3,500	3,8594	$0,19248*10^{-2}$
3,500	3,8410	$0,194*10^{-2}$
3,500	3,8241	$0,19536*10^{-2}$
3,500	3,8087	$0,19655*10^{-2}$
3,500	3,7945	$0,19767*10^{-2}$
3,500	3,7813	$0,19867*10^{-2}$
3,500	3,7692	$0,19957*10^{-2}$
3,500	3,7579	$0,20109*10^{-2}$

3,500	3,7474	$0,20114 \cdot 10^{-2}$
3,500	3,7376	$0,20181 \cdot 10^{-2}$
3,500	3,7284	$0,20254 \cdot 10^{-2}$
3,500	3,7198	$0,20289 \cdot 10^{-2}$
3,500	3,7117	$0,2035 \cdot 10^{-2}$
3,500	3,7041	$0,20396 \cdot 10^{-2}$
3,500	3,6970	$0,2044 \cdot 10^{-2}$
3,500	3,6902	$0,20479 \cdot 10^{-2}$
3,500	3,6838	$0,20517 \cdot 10^{-2}$
3,500	3,6777	$0,20535 \cdot 10^{-2}$
3,500	3,6720	$0,20578 \cdot 10^{-2}$
3,500	3,6665	$0,20605 \cdot 10^{-2}$
3,500	3,6613	$0,2064 \cdot 10^{-2}$
3,500	3,6564	$0,20656 \cdot 10^{-2}$
3,500	3,6516	$0,20678 \cdot 10^{-2}$
3,500	3,6471	$0,20691 \cdot 10^{-2}$
3,500	3,6428	$0,20711 \cdot 10^{-2}$
3,500	3,6387	$0,20728 \cdot 10^{-2}$
3,500	3,6348	$0,20741 \cdot 10^{-2}$
3,500	3,6310	$0,20754 \cdot 10^{-2}$

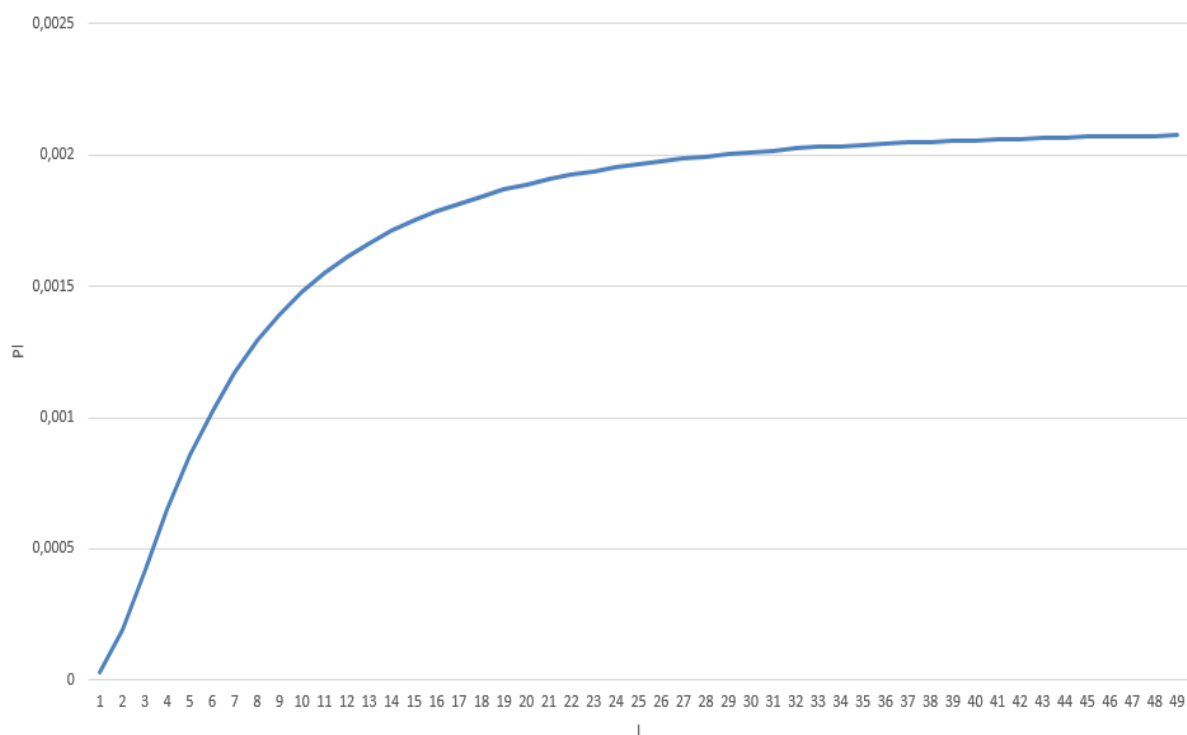


Рисунок 3.2 – Зависимость вероятности отказа в обслуживании от максимально возможного числа пользователей

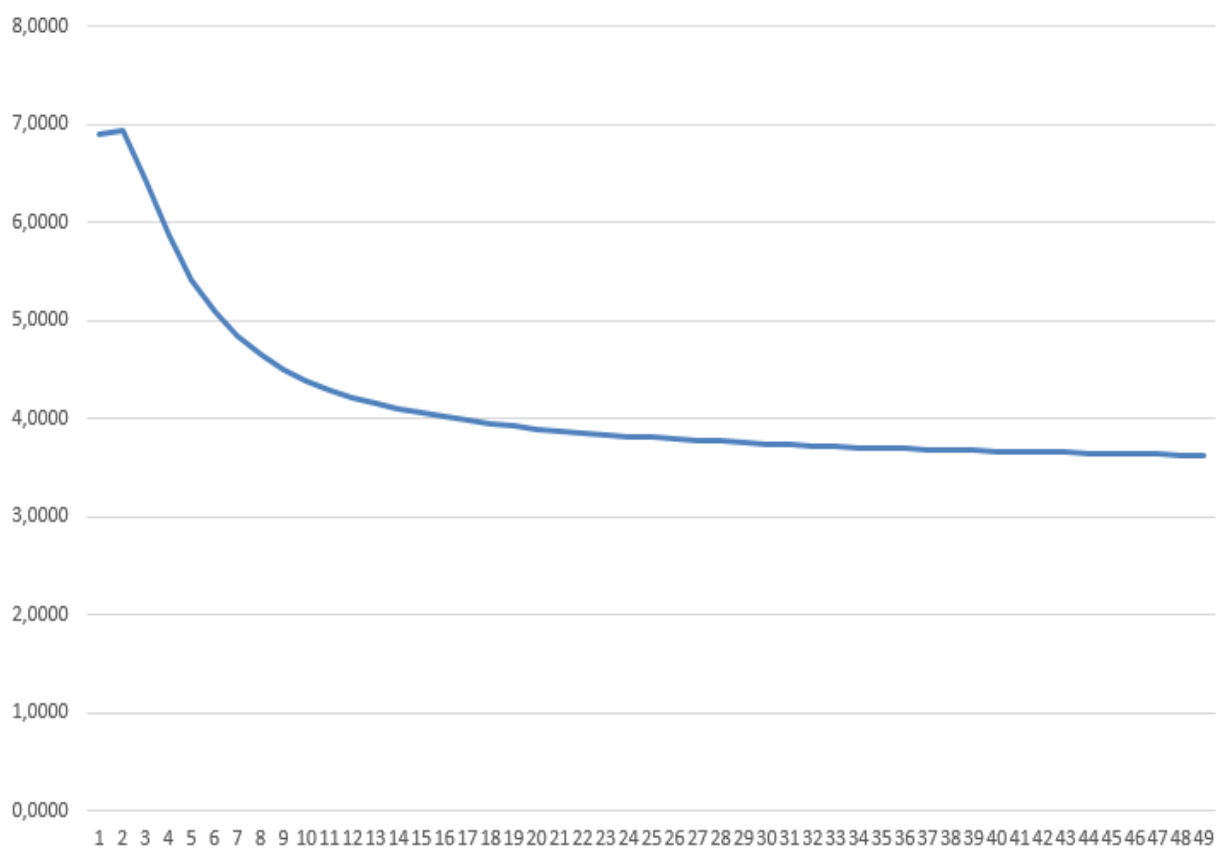


Рисунок 3.3 – Зависимость максимального времени задержки пользователя на сайте от максимально возможного числа пользователей

Как продемонстрировано на рисунках 3.2 и 3.3, можно увидеть, что при заданной интенсивности посещений сайта при увеличении максимально возможного число пользователей значение максимального времени задержки на странице падает, а вероятность потерь (отказа в доступе) растет.

Пусть максимально возможное значение $\lambda_{0_i} = 10$ [человек/мин] и $Tq_1 = 3$ [мин]. Исследуем поведение системы в диапазоне изменения значений максимально возможного числа пользователей $I=[1/49]$ [посетителей] с шагом 1, коррелируя при этом значение λ_{0_i} .

В таблице 3.4 обозначены результаты проведённого эксперимента.

Таблица 3.4 – Поведение системы в диапазоне изменения значений $I=[1/49]$ [посетителей] с шагом 1, коррелируя при этом значение λ_{0_i}

1	λ_{0_i}	Tq_0	Tq_{m_0}	P_l
1	0,20409	3,500	14,2200	0,34374
2	0,40816	3,500	14,2200	0,20782
3	0,61224	3,500	14,2200	0,14717
4	0,81633	3,500	14,2200	0,11233
5	1,0204	3,500	14,2200	$0,89488 \cdot 10^{-1}$
6	1,2245	3,500	14,2200	$0,73019 \cdot 10^{-1}$
7	1,4286	3,500	14,2200	$0,60676 \cdot 10^{-1}$
8	1,6827	3,500	14,2200	$0,51936 \cdot 10^{-1}$
9	1,8367	3,500	14,2200	$0,43315 \cdot 10^{-1}$
10	2,0405	3,500	14,2200	$0,37018 \cdot 10^{-1}$
11	2,2449	3,500	14,2200	$0,3181 \cdot 10^{-1}$
12	2,4490	3,500	14,2200	$0,27457 \cdot 10^{-1}$
13	2,6531	3,500	14,2200	$0,23789 \cdot 10^{-1}$
14	2,8571	3,500	14,2200	$0,20676 \cdot 10^{-1}$
15	3,0612	3,500	14,2200	$0,18019 \cdot 10^{-1}$
16	3,2653	3,500	14,2200	$0,15471 \cdot 10^{-1}$

17	3,4694	3,500	14,2200	$0,13779 \cdot 10^{-1}$
18	6,6735	3,500	14,2200	$0,12084 \cdot 10^{-1}$
19	3,8776	3,500	14,2200	$0,10615 \cdot 10^{-1}$
20	4,0816	3,500	14,2200	$0,93381 \cdot 10^{-2}$
21	4,2857	3,500	14,2200	$0,82258 \cdot 10^{-2}$
22	4,4898	3,500	14,2200	$0,72547 \cdot 10^{-2}$
23	4,6939	3,500	14,2200	$0,64053 \cdot 10^{-2}$
24	4,8980	3,500	14,2200	$0,5661 \cdot 10^{-2}$
25	5,1020	3,500	14,2200	$0,50078 \cdot 10^{-2}$
26	5,3061	3,500	14,2200	$0,44337 \cdot 10^{-2}$
27	5,5102	3,500	14,2200	$0,39285 \cdot 10^{-2}$
28	5,7143	3,500	14,2200	$0,34833 \cdot 10^{-2}$
29	5,9184	3,500	14,2200	$0,30907 \cdot 10^{-2}$
30	6,1224	3,500	14,2200	$0,21667 \cdot 10^{-2}$
31	6,3265	3,500	14,2200	$0,19288 \cdot 10^{-2}$
32	6,5308	3,500	14,2200	$0,17143 \cdot 10^{-2}$
33	6,7347	3,500	14,2200	$0,15259 \cdot 10^{-2}$
34	6,9388	3,500	14,2200	$0,13588 \cdot 10^{-2}$
35	7,1429	3,500	14,2200	$0,12015 \cdot 10^{-2}$
36	7,3469	3,500	14,2200	$0,10788 \cdot 10^{-2}$
37	7,5510	3,500	14,2200	$0,96175 \cdot 10^{-3}$
38	7,7551	3,500	14,2200	$0,85769 \cdot 10^{-3}$
39	7,9392	3,500	14,2200	$0,76513 \cdot 10^{-3}$
40	8,1633	3,500	14,2200	$0,68277 \cdot 10^{-3}$
41	8,3673	3,500	14,2200	$0,60946 \cdot 10^{-3}$
42	8,5714	3,500	14,2200	$0,54416 \cdot 10^{-3}$
43	8,7755	3,500	14,2200	$0,48599 \cdot 10^{-3}$
44	8,9796	3,500	14,2200	$0,43415 \cdot 10^{-3}$

45	9,1837	3,500	14,2200	$0,38793 \cdot 10^{-3}$
46	9,3578	3,500	14,2200	$0,34671 \cdot 10^{-3}$
47	9,5918	3,500	14,2200	$0,38793 \cdot 10^{-3}$
48	9,7959	3,500	14,2200	$0,34671 \cdot 10^{-3}$
49	10,000	3,500	14,2200	$0,31191 \cdot 10^{-3}$

Как видно из расчетов в таблице 3.4, значение Tq_{m_0} не зависит от числа одновременных посетителей и входного потока [14].

В данном параграфе была реализована математическая модель оценки надёжности веб-сайта и с помощью этой модели были вычислены значения вероятности безотказной работы сайта. Также было выявлено, что при заданной интенсивности посещений сайта при увеличении максимально возможного число пользователей значение максимального времени задержки на странице падает, а вероятность потерь (отказа в доступе) растет.

3.7 Доказательство гипотезы о возможности построения математической модели поиска численного значения вероятности отказа в работе сайта

Гипотезой диссертации является предположение о том, что может быть построена математическая модель, с помощью которой возможно получить численное значение вероятности отказа в работе сайта на этапе его проектирования, что является главным критерием надёжности.

Основными способами доказательства гипотез являются: дедуктивное обоснование выраженного в гипотезе предположения; непосредственное обнаружение предположенных в гипотезе предметов; логическое доказательство гипотезы.

Для доказательства данной гипотезы выберем такой метод, как непосредственное обнаружение искомых предметов, поскольку гипотеза состоит в предположении о возможно построения математической модели.

Частные гипотезы в науке и версии в судебном исследовании нередко ставят своей задачей выявление факта существования в определенное время и в определенном месте конкретных предметов и явлений либо отвечают на вопрос о свойствах и качествах таких предметов. Наиболее убедительным способом превращения такого предположения в достоверное знание является непосредственное обнаружение в предположенное время или в предположенном месте искомых предметов либо непосредственное восприятие предположенных свойств.

Проведём эксперимент, который поможет доказать гипотезу о возможности построения математической модели, которая поможет получить численное значение вероятности отказа в работе сайта.

Рассмотрим одну страницу веб-сайта ($i=1$). В случае использования системы G/G/1/0 время $Tq_0 = Tq_{0_i}$ и $Tq_{m_0} = Tq_{m_i}$, причём эти времена не зависят от λ_{0_i} . Пусть $Tq_1 = 3,5$ [мин] и максимально возможное число пользователей $l=47$. Результаты вычислений с использованием искомой модели представлены в таблице 1.

Далее, пусть $\lambda_{0_i} = 9$ [человек/мин], $Tq_1 = 3,5$ [мин]. Исследуем поведение системы в диапазоне изменения значений максимально возможного числа пользователей $l = [1 \div 49]$ с шагом 1. Расчет, проведённый с помощью данной математической модели дает результаты, приведенные в таблице 2.

Проведённые выше эксперименты показывают определение численного значения вероятности в отказе в работе сайта с помощью математической модели, построенной в главе 2.

Поскольку в качестве доказательной базы был выбран такой метод, как эксперимент, а с помощью построенной математической модели были найдены численные значения вероятности отказа в обслуживании, то гипотеза диссертации может считаться доказанной.

В данной главе был проведён анализ современного программного обеспечения, подходящего для реализации математической модели оценки надёжности веб-сайтов. Далее с помощью сравнения этих средств по ряду

показателей для реализации модели была выбрана инструментальная среда «MathCad». Затем, с помощью «MathCad» была реализована математическая модель оценки надёжности веб-сайтов и с использованием этой модели был проведён эксперимент, доказывающий, что можно найти количественное значение вероятности отказа в обслуживании сайта на этапе его проектирования, если построена математическая модель.

ЗАКЛЮЧЕНИЕ

В данной работе было проведено исследование оценки надёжности веб-сайта. В ней была составлена модель, которая позволяет при помощи систем массового обслуживания оценивать среднее и, что более важно, максимальное время задержки пользователей на сайте, что, в свою очередь, помогает принимать необходимые управленческие решения при развитии веб-сайта или его проектировании.

В данной диссертации был проведён анализ методов оценки надёжности веб-сайтов, с помощью которого было выявлено, что на данный момент задача оценки надёжности сайта при его проектировании недостаточно проработана.

Для решения этой задачи была разработана математическая модель оценки надёжности веб-сайта на этапе его проектирования, которая использует аппарат систем массового обслуживания.

Также было вычислено максимальное время пребывания пользователя на сайте с помощью решения трансцендентных уравнений.

В работе были выявлены следующие зависимости:

- зависимость вероятности отказа в обслуживании от интенсивности потока заявок к странице - при определённом уровне интенсивности входного потока заявок к странице вероятность отказа в доступе начинает расти;
- зависимость вероятности отказа в обслуживании от максимально возможного числа пользователей – при увеличении максимально возможного числа посетителей вероятность отказа в доступе увеличивается;
- зависимость максимального времени задержки пользователя на сайте от максимально возможного числа пользователей – при увеличении максимально возможного числа пользователей максимальное время задержки пользователя на странице падает.

Из данных зависимостей можно сделать вывод, что при увеличении максимального числа пользователей, вероятность отказа будет увеличиваться, а

максимальное время задержки пользователя на странице будет только падать, что отрицательно скажется на позициях самого сайта.

Также в магистерской диссертации была доказана гипотеза о том, что может быть построена математическая модель, с помощью которой возможно получить численное значение вероятности отказа в работе сайта на этапе его проектирования, что является главным критерием надёжности.

СПИСОК ИСПОЛЬЗУЕМОЙ ЛИТЕРАТУРЫ

Нормативно-правовые акты

1. ГОСТ 27.002—89 «Надежность в технике. Основные понятия. Термины и определения» 3.Дефекты, повреждения, отказы
2. ГОСТ 27.002—89 «Надежность в технике. Основные понятия. Термины и определения» 4.Временные понятия
3. ГОСТ 27.002—89 «Надёжность в технике. Основные понятия. Термины и определения» 1.Общие понятия
4. ГОСТ Р 50779.27—2017 «Статистические методы. Распределение Вейбулла. Анализ данных»

Научная и методическая литература

5. Абрамов Петр Борисович, Леньшин Андрей Валентинович Оценка параметров систем массового обслуживания при аппроксимации дисциплины обслуживания потоками Эрланга // Вестник ВИ МВД России. 2012. №2.
6. Александрова У.А., Сенашов С.И. Анализ статистики посещаемости сайта типичного Красноярского кинотеатра // Актуальные проблемы авиации и космонавтики. 2014. №10.
7. Александровская Л. и др. Безопасность и надежность технических систем. – Litres, 2017.
8. Александровская, Л. Н. Статистические методы анализа безопасности сложных технических систем: учебник / Л.Н. Александровская, И.З. Аронов, А.И. Елизаров [и др.]; под ред. В.П. Соколова. – М.: Логос, 2001. – 232 с.
9. Афанасьев Т. А., Линдигрин А. Н., Яковлев Б. С. Применения отказоустойчивых систем и резервного копирования на полиграфических предприятиях //Известия Тульского государственного университета. Технические науки. – 2017. – №. 9-1.

10. Ашихмин Н. А. Разработка сервиса для обеспечения автоматизированного тестирования облачных приложений : дис. – Южно-Уральский государственный университет, 2016.
11. Богомолова Е. П., Очков В. Ф., Мати Х. Решатели или великолепная семерка Mathcad //Открытое образование. – 2015. – №. 3.
12. Бойченко О. В., Тупота Е. С. РЕШЕНИЕ ПРОБЛЕМ ПРЕДОТВРАЩЕНИЯ УЯЗВИМОСТЕЙ САЙТОВ //Международный научно-исследовательский журнал. – 2016. – №. 4-2. – С. 42-44.
13. Брумштейн Ю. М., Бондарев А. А. Информационная безопасность сайтов высших учебных заведений: проблемы и решения //Экономическая безопасность и качество. – 2014. – №. 1 (14).
14. Васильев Д. А., Проколова Н. С. Анализ подходов к повышению безопасности интернет-сайтов, развернутых с использованием систем наполнения контентом //Auditorium. – 2017. – №. 2 (14).
15. Власова Н. С., Чернякова Т. В., Прокубовская А. О. Подходы к проектированию интерфейсов информационно-образовательных сред //Сибирский педагогический журнал. – 2013. – №. 2.
16. Ворожцов А.С., Тутова И.В., Тутов А.В. Методика оптимального распределения виртуальных серверов в центрах обработки данных. //Т-Comm Телекоммуникации и транспорт, 2015, № 7. -С.3-5.
17. Горбунков А.Л. Марковские модели посещаемости web-сайтов. //http://hdl.handle.net/10995/1334 (дата обращения ноябрь 2017).
18. Горнаков С. Осваиваем популярные системы управления сайтом (CMS). – Litres, 2017.
19. Гуров С. В., Уткин Л. В. Надежность восстанавливаемых резервированных систем с последствием отказов //Автоматика и телемеханика. – 2017. – №. 1. – С. 137-151.
20. Гусейнов Р. В., Султанова Л. М. Определение показателей надежности автомобилей //Вестник Дагестанского государственного технического университета. Технические науки. – 2015. – Т. 38. – №. 3.

21. Дормидонтова Т. В., Кирьяков В. В. Применение методов теории надёжности на практике //Интернет-журнал Науковедение. – 2015. – Т. 7. – №. 1 (26).
22. Дуплякин В.М., Княжева Ю.В. Выбор закона распределения входного потока заявок при моделировании системы массового обслуживания торгового предприятия. // Вестник Самарского государственного аэрокосмического университета им. академика С.П. Королёва (национального исследовательского университета). – 2012. - № 6(37)
23. Егорова И. Н., Гладкая А. А. Исследование возможностей резервного копирования веб-сайтов, созданных на основе WordPress. – 2017.
24. Елисеева, Т. А. Корректность экспертных оценок при анализе надёжности технических систем на стадии проектирования / Т.А. Елисеева, Е.В. Плахотникова // Научно-практ. конф. преподавателей, сотрудников и аспирантов: сб. на-уч. тр. – Вып.4. – ВСГУТУ. – Улан-Уде. – 2015. – С. 35-41.
25. Ефстафьев, И. Н. Тотальный риск-менеджмент / И.Н. Ефстафьев. – М.: Эксмо, 2008. – 208 с.
26. Зиннатулин Ф. Ф. ОБЛАЧНЫЙ И КЛАССИЧЕСКИЙ ХОСТИНГ В ВЕБ-РАЗРАБОТКЕ //НОВЫЕ ИНФОРМАЦИОННЫЕ ТЕХНОЛОГИИ В НАУКЕ НОВОГО ВРЕМЕНИ. – 2017. – С. 40-43.
27. Зиннатулин Ф. Ф. Основные направления развития веб-дизайна в проектировании веб-сайтов в 2016 году //Перспективы развития информационных технологий. – 2016. – №. 31. – С. 126-130.
28. Зюзин Р. Г. и др. ПЕРСПЕКТИВЫ РАЗВИТИЯ ОНЛАЙН-ПЛАТФОРМ С ПОМОЩЬЮ СИСТЕМ ВЕБ-АНАЛИТИКИ //Международный научно-исследовательский журнал. – 2017. – №. 05 (59) Часть 3. – С. 47-49.
29. Исмагилова А. Х. Оценка качества электронных библиотек. – 2015.
30. Канн С. К. Библиотечный сайт: статистика отказов и отказ от базовых ценностей //Вестник научных конференций. – ООО Консалтинговая компания Юком, 2017. – №. 3-4. – С. 31-34.

31. Карманов Ф. И., Острейковский В. А. Статистические методы обработки экспериментальных данных. Лабораторный практикум с использованием пакета MathCad. Учебное пособие. – 2012.
32. Каштанов В., Медведев А. Теория надежности сложных систем. – Litres, 2017.
33. Кирпичников А. П., Фадхкал З. Прикладная теория массового обслуживания //Теория и практика современной науки. – 2014. – С. 15-19.
34. Кирпичников А.П., Титовцев А.С., Фадхкал З. Некоторые особенности числовых характеристик многоканальных систем массового обслуживания открытого типа. // Вестник Казанского технологического университета. – 2015.
35. Можаяев А.С. Логико-вероятностный подход к оценке надежности автоматизированных систем управления. СПб.: ВМА им. Гречко А.А. Депонирована п/я А-1420 № Д047550, 1982. - 24 С.
36. Мокров Е.В., Самуйлов К.Е. Модель системы облачных вычислений в виде системы массового обслуживания с несколькими очередями и с групповым поступлением заявок. // Т-Сотт - Телекоммуникации и Транспорт – 2013. - №11.
37. Осипов Г.С. Моделирование систем массового обслуживания с отказами. // Бюллетень науки и практики. – 2016. - № 11(12).
38. Пятков А. Г., Золотарёв В. В. Методы анализа надёжности космических аппаратов //Актуальные проблемы авиации и космонавтики. – 2015. – Т. 1. – №. 11.
39. Розенберг В. Я., Прохоров А. И. Что такое теория массового обслуживания. – Рипол Классик, 2013.
40. Сербиновский Б. Ю., Сербиновская А. А., Белоус М. А. Развитие коммуникаций и сбыта продукции и услуг ресторанной сети с использованием IT-технологий (опыт моделирования и проектирования многофункционального веб-сайта). Часть 1 //Инженерный вестник Дона. – 2013. – Т. 25. – №. 2 (25).

41. Смирнов А. А., Коваленко А. В. Методы качественного анализа и количественной оценки рисков разработки программного обеспечения //Системи обробки інформації. – 2016. – №. 5. – С. 153-157.

42. Сыдыкова Ж. Н., Мусиралиева Ш. Ж. Метод обеспечения безопасности веб-сайтов //Инновации в науке. – 2016. – №. 55-1. – С. 58-61.

43. Ташков П. А., Ташков П. А. Веб-мастеринг: HTML, CSS, JavaScript, PHP, CMS, графика, раскрутка: Html, Css, Javascript, Php, Cms, Grafika, Raskrutka. – Издательский дом" Питер", 2008.

44. Шевченко Д. А., Локтюшина Ю. В. Эффективность веб-сайтов высших учебных заведений //Методика оценки конкурентоспособности сайта вуза в Интернете. М.: ННОУ "МИПК. – 2014.

45. Шейн Е. А. Методологический подход к разработке официального сайта как действенного инструмента маркетинговой деятельности вуза в сети Интернет //Вестник Белгородского университета кооперации, экономики и права. – 2012. – №. 3. – С. 43.

46. Шушерин, В. В. Средства и методы управления качеством: учебное пособие / В.В. Шушерин, С.В. Кортов, А.С. Зеткин. – Екатеринбург: ГОУ ВПО УГТУ-УПИ, 2006. – 202 с.

Электронные ресурсы

47. <https://metrika.yandex.ru/> (дата обращения ноябрь 2017).

48. <https://www.google.com/analytics/> (дата обращения ноябрь 2017).

49. Фонталин Е. Оценка эффективности корпоративного сайта, 2014. - // <http://www.cossa.ru/155/60369/> (дата обращения апрель 2016).

50. Яндекс. Вебмастер. <https://yandex.ru/support/webmaster/protecting-sites/intro.html>

Литература на иностранном языке

51. Alam M. I., Pandey M., Rautaray S. S. A comprehensive survey on cloud computing //International Journal of Information Technology and Computer Science. – 2015. – Т. 2. – С. 68-79.

52. Benker H. Practical Use of Mathcad®: Solving Mathematical Problems with a Computer Algebra System. – Springer Science & Business Media, 2012.
53. Hyun K. C. et al. Risk analysis using fault-tree analysis (FTA) and analytic hierarchy process (AHP) applicable to shield TBM tunnels //Tunnelling and Underground Space Technology. – 2015. – T. 49. – C. 121-129.
54. Ikram M. A., Hussain F. K. Software as a Service (SaaS) Service Selection Based on Measuring the Shortest Distance to the Consumer's Preferences //International Conference on Emerging Internetworking, Data & Web Technologies. – Springer, Cham, 2018. – C. 403-415.
55. ITU-T Focus Group on Cloud Computing Technical Report Version 1.0 Part 3: Requirements and framework architecture of cloud infrastructure 2012, 51p.
56. Jarno A. D. B. et al. Cloud Testing: Requirements, Tools and Challenges //JOURNAL OF APPLIED TECHNOLOGY AND INNOVATION. – 2017. – T. 1. – №. 2. – C. 79-93.
57. Kaur M. A., Kumari M. P., Singh M. P. A TAXONOMY ON TOOLS FOR SCIENTIFIC WORKFLOW MANAGEMENT SYSTEM.
58. Khalid I. et al. BUILDING A WEB-STORE USING PRESTASHOP PLATFORM. – 2017.
59. Khinchin A. Y., Andrews D. M., Quenouille M. H. Mathematical methods in the theory of queuing. – Courier Corporation, 2013.
60. Ochkov V. F., Bogomolova E. P. Teaching Mathematics with Mathematical Software //Journal of Humanistic Mathematics. – 2015. – T. 5. – №. 1. – C. 265-285.
61. Robertazzi T. G. Computer networks and systems: queueing theory and performance evaluation. – Springer Science & Business Media, 2012.
62. Sekar K. R., Sethuraman J. Optimal component selection for rich internet applications in web engineering //Networks & Advances in Computational Technologies (NetACT), 2017 International Conference on. – IEEE, 2017. – C. 420-425.

63. TAUFIQ F. M. VULNERABILITY ANALISIS UNTUK PENINGKATAN SISTEM KEAMANAN WEBSITE (STUDI KASUS DI LINGKUNGAN UMM) : дис. – University of Muhammadiyah Malang, 2017.