

ЧАСТЬ 3. ОЦЕНКА НАДЕЖНОСТИ ПРОГРАММНЫХ СИСТЕМ: ТЕОРИЯ ДИНАМИКИ И ЕЕ ПРИЛОЖЕНИЯ

Для оценки надежности программных систем (ПС) широко используются методы математического моделирования. Целью моделирования показателей надежности ПС является оценка количества оставшихся в системе дефектов, а также прогнозирование времени и динамики их выявления. Эта задача не является новой, но методы ее решения не могут считаться хорошо изученными. Так, например, в [1] авторы отмечают: «Необходимо подчеркнуть, что к настоящему времени теорию надежности программных средств нельзя рассматривать как сложившуюся науку ... Поэтому (как ни банально это звучит) можно констатировать наличие существенного разрыва между теорией (математическими моделями и методами) и практикой». Действительно, на сегодняшний день известно более двадцати (а с учетом разновидностей – более тридцати) различных моделей надежности программного обеспечения (МНПО). Наиболее полное систематизированное описание этих моделей в отечественной литературе приведено в [1], а в иностранной – в [2]. Все известные МНПО основаны на едином концептуальном базисе – теории вероятностей и рассматривают выявление ошибок как вероятностный процесс с определенным законом распределения. Каждая модель характерна собственным набором допущений, ограничивающих область ее применения. Как показывает опыт, результаты моделирования существенно зависят от исходных данных – конкретных программных продуктов, что еще раз подтверждает отмеченный выше разрыв между теорией и практикой. Принимая во внимание, что большинство моделей имеет пересекающиеся множества допущений, возникает проблема их выбора для использования в том или ином конкретном случае.

Представляется, что такое обилие разных МНПО не может являться доказательством хорошей изученности этого вопроса, а свидетельствует, скорее, о нашем непонимании сути процессов, приводящих к проявлению ошибок в программных системах. В.А. Эткин в работе «К единой теории реальных процессов» [3] отмечает, что «ученые перестали тяготиться тем, что теории не проясняют реальности, и уже не ставят задачей понимание причинно-следственных связей в проявлениях тех или иных законов. Объяснение явлений перестало быть основной функцией науки». В связи со сказанным несомненный интерес представляет создание единого теоретического и методологического базиса, объединяющего различные подходы и объясняющего процессы выявления и исправления ошибок ПС.

В данном разделе изложены основы теории динамики программных систем (ДПС), которая, как показали исследования, и может служить этим недостающим теоретическим и методологическим базисом. Как показано ниже, из этой теории непосредственно следуют все наблюдаемые закономерности выявления ошибок, а сама ДПС может использоваться для моделирования и прогнозирования показателей надежности программных систем.

3.1. Основы теории динамики программных систем

3.1.1. Дефекты и ошибки. Понятие динамики программных систем

В начале дадим определение терминам «ошибка» и «дефект». В ГОСТ 15467-79 [4] термин «дефект» определяется как каждое отдельное несоответствие продукции установленным требованиям. Если единица продукции имеет дефект, то это значит, что по меньшей мере один из показателей ее качества или параметров вышел за предельное значение или не выполняется одно из требований нормативной документации к признакам продукции.

В случае программных систем требования нормативной документации лишь в идеале могут быть критерием для идентификации дефекта. Дело в том, что большинство ПС претерпевают постоянные изменения вслед за изменениями своей предметной области (ПрО). В идеале каждое изменение ПрО должно повлечь изменение документации и уже потом эти изменения должны стать основанием для изменений ПС. Однако, изменение документации это длительный процесс, предусматривающий долгий путь согласований и утверждений, а при изменениях ПрО, соответствующая программная система должна учитывать эти изменения «уже сегодня». В цепочке «изменение ПрО» – «изменение документации» – «изменение ПС» второе звено является самым инерционным. Вследствие этого запаздывания, на основании сравнения результатов работы ПС с документацией, не всегда можно судить о возникновении дефекта.

В случае систем с неизменными ПрО неизменной остается и их документация. Если документация разработана тщательно и полностью соответствует требованиям ПрО, становится безразличным, с чем сравнивать результат работы ПС – с требованиями документации или требованиями ПрО. Это дает основание в общем случае называть *дефектами* любые несоответствия ПС своей ПрО. В процессе эксплуатации эти несоответствия приведут к отклонениям результата

функционирования ПС от требований ПрО, то есть к проявлению дефектов. Такие проявленные дефекты будем называть *ошибками* ПС.

Можно сказать, что дефект и ошибка соотносятся между собой как причина и следствие. Любая ошибка является результатом проявления дефекта во время работы ПС.

Объектом рассмотрения теории динамики программных систем являются присущие им дефекты. Предметом рассмотрения являются динамические процессы и законы, которые обуславливают возникновение, выявление и ликвидацию дефектов.

3.1.2. Основные предположения динамики программных систем

Теория ДПС рассматривает программную систему как открытую неравновесную систему, взаимодействующую с окружающей средой. В качестве окружающей среды выступает ПрО такой системы. Неравновесные системы – это системы, в которых имеют место градиенты определенных свойств, например, концентрации, температуры и так далее, и характеризуются наличием диссипативных (от лат. *dissipatio* – «рассеиваю, разрушаю») структур [5]. В работе [6] акцентируется внимание на широком распространении диссипативных структур: «Мы начинаем понимать, сколь велико разнообразие структур, которое мы наблюдаем в Природе. Ныне понятие диссипативной структуры и самоорганизации встречаются в широком диапазоне научного познания, от астрофизики к гуманитарных наукам и экономике».

Диссипативные структуры возникают и существуют благодаря потокам энергии, вещества, энтропии, которые создаются и поддерживаются внешними силами, а также градиентами определенных свойств системы. В общей теории неравновесных процессов физическая природа этих свойств, которые получили название «термодинамических потенциалов» [6], не суть важна. Важно лишь то, что в системе существуют их градиенты, которые исполняют роль термодинамических сил. Под воздействием этих сил возникают потоки, которые призваны привести систему к состоянию равновесия со своей окружающей средой. Динамика такой системы определяется пространственно-временным распределением этих потоков, имеющих свои значения в каждой ее физической точке.

Для программной системы понятие пространства возможно лишь в смысле «в пределах» или «за пределами» системы, а понятие физической точки вообще не может быть использовано. Поэтому касательно ПС можно говорить лишь о закономерностях распределения потоков во времени. Открытость ПС определяется степенью и характером их

взаимосвязи с внешним миром, в качестве которого выступает ПрО системы, а уровень неравновесности определяется количеством дефектов, содержащихся в ней. При этом, как указывалось, сама ПрО принимается за эталон, то есть в ней по определению не может быть дефектов.

Количество дефектов, содержащихся ПС в момент времени t будем обозначать как f или $f(t)$.

Теория ДПС основывается на сформулированных ниже основных предположениях.

Предположение 1. ПС является открытой неравновесной системой, которая взаимодействует со своей ПрО по законам неравновесных процессов.

Предположение 2. ПС может характеризоваться специальной функцией состояния - количеством находящихся в ней дефектов.

Предположение 3. Исчезновение и появление дефектов в ПС является результатом действия прямого (выходного) и обратного (входного) потоков.

Предположение 4. Интенсивность (скорость изменения потока во времени) пропорциональна количеству дефектов, которые образуют поток.

Предположение 5. (Принцип равноправности). Все дефекты равноправны и участвуют в образовании потока одинаковым образом, независимо от причины возникновения, места расположения, типа дефекта и возможных последствий его проявления.

Предположение 6. (Принцип непрерывности). Существует производная $\frac{df}{dt}$ как предел:

$$\frac{df}{dt} = \lim_{\Delta t \rightarrow 0} \frac{f(t) - f(t + \Delta t)}{\Delta t}.$$

Основной концепцией ДПС является концепция потоков программных дефектов. При этом вне поля зрения остаются причины возникновения, последствия проявления и обстоятельства исправления каждого конкретного дефекта. Каждый дефект рассматривается как составная часть общего потока, который подчиняется уже не законам теории вероятностей, а законам возникновения и изменения во времени потоков в неравновесных системах.

3.2. Поток первичных и вторичных дефектов программных систем

Традиционно считается, что причиной возникновения дефектов в ПС являются ошибки, допущенные разработчиками во время их создания, отладки и сопровождения. Имеющиеся в ПС дефекты можно условно разделить на первичные и вторичные. *Первичными* называются дефекты, которые присутствовали в системе на момент начала исследований (при $t=0$). В результате их исправления в ПС могут быть внесены новые дефекты. Такие дефекты, которые отсутствовали в системе при $t=0$ и появились в ней в более позднее время, называются *вторичными*.

Однако, что не отмечено в литературе, данное ранее определение допускает существование еще одного рода дефектов. Рассмотрим случай (возможно гипотетический) системы, не содержащей дефектов и полностью соответствующей своей ПрО. Допустим, что в какой-то момент времени ПрО этой системы изменилась. С этого момента ПС функционирует не верно с точки зрения изменившейся ПрО. А это означает, что в ПС, согласно данному определению, возникли дефекты. Движение, как известно, относительно. Поэтому, рассматривая исправление дефектов как приближение ПС к своей ПрО, мы должны учесть и возможное обратное движение – движение ПрО от соответствующей ПС.

Таким образом, возникновение дефектов возможно и в ПС, в которую вообще не вносилось никаких изменений. Более того, эти дефекты возникли именно из-за того, что в систему не вносились изменения. С точки зрения теории ДПС, подобные дефекты можно отнести к числу вторичных. Действительно, ведь возникли эти дефекты при $t > 0$, а в силу принципа равноправности причина их возникновения не имеет значения.

Схема возникновения и взаимодействия потоков представлена на рис.3.1.

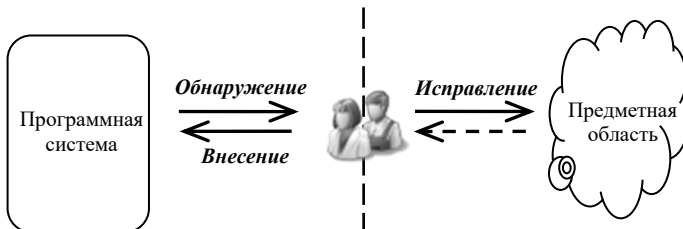


Рис. 3.1. Образование потоков дефектов в программной системе

Поясним причины возникновения потоков. При обнаружении дефекта он устраняется, исчезая из ПС. Такое исчезновение можно трактовать как *вынесение дефекта* за пределы ПС, то есть, как возникновение прямого *выходного потока* дефектов, направленного из ПС в окружающую среду. Конкретный механизм такого вынесения, конкретный процесс выявления дефекта и его устранения, равно как и конкретные исполнители, нас в данный момент не интересуют. Важен сам факт наличия дефекта в системе и его исчезновения через определенное время.

При исправлении дефекта возможно внесение в ПС новых вторичных дефектов. Они ранее не существовали в системе, а были *внесены* в нее. Это дает основание говорить о возникновении обратного *входного потока* дефектов, направленного из окружающей среды в ПС.

Сложность восприятия процесса выявления и исправления отдельных ошибок как целостного явления обусловлена тем, что человек (будем называть его «наблюдатель»), который собственно обнаруживает и исправляет дефекты, сам является составной частью данной системы. Именно целеустремленные действия наблюдателя и являются той причиной, благодаря которой дефекты ПС исправляются, то есть сам наблюдатель является носителем потока, а его действия – причиной возникновения потоков дефектов. С одной стороны наблюдатель концентрирует внимание на выявлении и исправлении каждой конкретной ошибки. С другой стороны, процессы выявления и исправления достаточно длительны во времени (для сложных систем это месяцы или даже годы). Все это способствует тому, что наблюдатель, концентрируясь на каждом отдельном дефекте не замечает существования потока.

В работе [3] показано, что для каждого независимого процесса, которым в частности является процесс выявления программных дефектов, существует особая, специфическая для него функция состояния – координата, отвечающая этому процессу. В случае процесса выявления и исправления программных дефектов в качестве такой координаты можно рассматривать количество дефектов f_1 или $f_1(t)$.

В неравновесной динамике под потоком какой-либо скалярной величины понимают произведение скорости изменения этой величины на площадь поверхности контакта. Обменные процессы между системой и внешней средой будут идти тем быстрее, чем больше площадь поверхности контакта. В случае программной системы роль поверхности контакта играет сам наблюдатель, а интенсивность обменных процессов определяется скоростью его работы по выявлению и исправлению

дефектов. Понятие площади в этом случае не имеет смысла, поэтому под потоком величины будем понимать просто скорость ее изменения во времени. Все формулы неравновесной динамики при этом сохраняют смысл, а потоки, которые входят в них, могут считаться удельными потоками, проходящимися на единицу площади.

Поток выявленных дефектов в этом случае численно может характеризоваться величиной скорости (интенсивности) потока $\frac{df_1}{dt}$. Эта

характеристика тождественна интенсивности отказов – основному показателю надежности программных систем. Сама система при этом может рассматриваться как имеющая только одну степень свободы, а процессы в ней описывается дифференциальным уравнением первого порядка.

В случае учета встречного потока – внесения в ПС вторичных дефектов, его координатой может быть текущее количество внесенных дефектов, которое обозначим как f_2 или $f_2(t)$. При совместном рассмотрении обоих потоков получим две координаты, характеризующие процессы выявления дефектов в ПС – f_1 и f_2 , а сама система в этом случае должна рассматриваться как система с двумя степенями свободы и описываться дифференциальными уравнениями второго порядка.

Порядок дифференциального уравнения, описывающего динамику выявления дефектов, дает возможность ввести новую характеристику программных систем – «порядок системы» и классифицировать их по этой характеристике.

Определение. Порядком программной системы будем называть порядок дифференциального уравнения, которое описывает закон изменения количества выявленных дефектов в этой системе.

При учете только одного выходного потока первичных дефектов программная система является системой первого порядка. При учете встречного потока вторичных дефектов программная система становится системой второго порядка.

Рассмотрим отдельно моделирование потоков дефектов и последствий их взаимодействия в системах первого и второго порядков.

3.3. Динамика программных систем первого порядка

Будем полагать, что в программной системе действует только выходной поток дефектов. Положение о действии только одного этого потока, то есть наличия в системе только одной степени свободы,

эквивалентно следующим двум предположениям большинства известных МНПО :

- при выявлении ошибки она исправляется до выявления следующей;
- новые ошибки при исправлениях не вносятся.

Действительно, первое предположение фактически эквивалентно наличию потока дефектов (дефекты исправляются, то есть перестают существовать, выносятся из системы), а второе говорит об отсутствии потока вторичных дефектов.

Согласно [6], поток любой скалярной величины в неравновесной системе возникает только за счет действия движущих сил, которые отвечают этому потоку. В качестве такой движущей силы в непрерывных системах выступает градиент потенциала соответствующей величины, а в прерывных – разность потенциалов на границе контакта. Программную систему, как это показано на рис. 3.1, следует рассматривать именно как прерывную – внутри системы дефекты есть, а во внешней среде они отсутствуют. Из этого следует, что потенциал на границе контакта изменяется скачкообразно. Принимая во внимание отсутствие дефектов во внешней среде, можем принять потенциал дефектов этой среды равным нулю. Тогда разность потенциалов программной системы и внешней среды будет равняться потенциалу дефектов системы, а выходной поток, согласно [6], может быть представлен как:

$$\frac{df_1}{dt} = -G_1 \cdot \varphi_1, \quad (3.1)$$

где G_1 - коэффициент пропорциональности, а φ_1 - потенциал дефектов программной системы. Знак « минус » в формуле (3.1) говорит о том, что поток $\frac{df_1}{dt}$ направлен в сторону убывания потенциала, то есть из программной системы во внешнюю среду. При этом количество дефектов в системе будет уменьшаться, а производная $\frac{df_1}{dt}$ будет отрицательной.

Между потенциалом дефектов φ_1 и соответствующей ему величиной количества дефектов f_1 существует соотношение $f_1 = C_1 \cdot \varphi_1$, где коэффициент C_1 называется емкостью системы по отношению к величине f_1 . Поэтому, с учетом понятия емкости, уравнение (3.1) можно переписать как

$$\frac{df_1}{dt} = -\frac{G_1}{C_1} \cdot f_1. \quad (3.2)$$

Уравнения, по структуре сходные с (3.1) или (3.2) имеют место для любой открытой неравновесной системы и фактически описывают процессы переноса в этих системах. Попробуем разобраться с физическим смыслом коэффициентов G_1 и C_1 в случае программной системы. В теории неравновесных процессов коэффициент G_1 называется проводимостью системы по отношению к величине f_1 . Из уравнения (3.1) следует, что при постоянной величине потенциала ϕ_1 скорость выявления ошибок $\frac{df_1}{dt}$ прямо пропорциональна величине проводимости системы G_1 .

В программной системе скорость выявления ошибок будет прямо пропорциональна частоте обращений наблюдателя к системе. Здесь имеется в виду «идеальный» наблюдатель, который обязательно обнаружит дефект именно в момент его проявления. Поэтому проводимость G_1 может трактоваться как частота обращений наблюдателя к системе. Это следует также из анализа размерностей в формуле (3.1). Параметр процесса – f_1 (количество дефектов в системе) является безразмерной величиной. Производная $\frac{df_1}{dt}$ имеет размерность *время*⁻¹, поэтому коэффициент G_1 должен иметь такую же размерность, то есть размерность частоты.

В реальном случае наблюдатель может не заметить ошибку и она будет выявлена при дальнейших обращениях. Однако это эквивалентно простому снижению частоты обращений. Таким образом, в качестве проводимости G_1 в ПС выступает частота обращений наблюдателя к системе. Потенциал ϕ_1 в силу этого, должен быть безразмерным.

Емкость C_1 показывает, как должно измениться количество дефектов в системе, чтобы их потенциал ϕ_1 изменился на единицу потенциала. С учетом того, что в программных системах потенциал является безразмерным, безразмерной должна быть и емкость системы.

Обозначим $A_1 = \frac{G_1}{C_1}$. Тогда уравнение (3.2) можно переписать как:

$$\frac{df_1}{dt} + A_1 \cdot f_1 = 0. \quad (3.3)$$

Коэффициент A_1 в этом уравнении показывает влияние количества дефектов в ПС на интенсивность выходного потока. Поэтому будем называть его коэффициентом влияния.

Производная $\frac{df_1}{dt}$ в (3.3) по своему физическому смыслу является интенсивностью отказов, на основании которой можно рассчитать остальные показатели надежности и прогнозировать ее изменения во время эксплуатации системы:

$$\lambda_1(t) = \left| \frac{df_1}{dt} \right|. \quad (3.4)$$

Здесь производная взята по модулю, так как интенсивность отказов всегда является положительной величиной, а количество дефектов в ПС уменьшается и производная будет отрицательной.

Выражение (3.3) – линейное однородное дифференциальное уравнение первого порядка. Его решением с учетом начального условия $f_1(0) = F_0$ является функция

$$f_1(t) = F_0 \cdot e^{-A_1 t}, \quad (3.5)$$

где F_0 – количество дефектов в ПС на момент начала исследования.

По этой формуле можно определить количество дефектов, оставшихся в системе на момент времени t от начала исследования.

Интенсивность отказов $\lambda_1(t)$ для ПС первого порядка на основании (3.4) и (3.5) можно вычислить как

$$\lambda_1(t) = A_1 \cdot F_0 \cdot e^{-A_1 t}. \quad (3.6)$$

Полученные на основании теории ДПС результаты (3.5) и (3.6) полностью совпадают с математическими выражениями многих МНПО, таких, например, как экспоненциальные модели Джелинского-Моранды [7], неоднородного Пуассоновского процесса (модель Гела-Окумото) [8], Шнайдервинда [9], Мусы [10] и других. Интересно отметить, что некоторые модели, которые не являются экспоненциальными, тоже имеют подобные математические выражения. Сюда относится модель распределения Вейбулла [11], S-образная модель [12], основанная на гамма-распределении, логарифмическая модель Мусы-Окумото [13]. В основу этих моделей положены разные математические предположения, в основном из теории вероятностей.

В теории ДПС не только получен тот же результат, но и на основании общепринятых физических представлений этот результат объяснен. Теперь, базируясь на ДПС, можно не только знать, что происходит с дефектами в программной системе, но и понимать, почему это происходит.

Однако, все существующие модели надежности основаны на том предположении, что во время исправления дефекта новые дефекты не вносятся. Теория ДПС позволяет снять это ограничение за счет введения встречного потока дефектов, направленного из внешней среды в систему. Для этого перейдем к рассмотрению динамики программных систем второго порядка.

3.4. Динамика программных систем второго порядка

В случае учета потока вторичных ошибок $\frac{df_2}{dt}$ программная система имеет уже две степени свободы и характеризуется двумя координатами – количеством дефектов f_1 , которые образуют выходной поток и будут вынесены из системы, и количеством внесенных в систему вторичных дефектов f_2 . На основании динамики неравновесных процессов связь между потоками $\frac{df_1}{dt}$ и $\frac{df_2}{dt}$ опишем системой уравнений:

$$\begin{cases} \frac{df_1}{dt} = -G_{11} \cdot \varphi_1 - G_{12} \cdot \varphi_2 \\ \frac{df_2}{dt} = -G_{21} \cdot \varphi_1 - G_{22} \cdot \varphi_2 \end{cases} \quad (3.7)$$

В этой системе φ_1 – потенциал дефектов, выносимых из системы, а φ_2 – потенциал вносимых вторичных дефектов. Коэффициенты G_{11} и G_{22} характеризуют влияние потенциалов φ_1 и φ_2 на собственные, связанные с ими потоки $\frac{df_1}{dt}$ и $\frac{df_2}{dt}$. По аналогии с изложенным в п. 3.3, эти коэффициенты играют роль проводимости и характеризуют частоту обращений к одной и той же системе. Поэтому можно утверждать, что $G_{11} = G_{22}$. Коэффициенты G_{12} и G_{21} характеризуют перекрестные влияния потенциалов на не связанные с ними потоки. Согласно принципу симметрии Онзагера [14], эти перекрестные влияния одинаковы, из чего следует равенство коэффициентов G_{12} и G_{21} .

Для перехода от потенциалов φ_1 и φ_2 к более физически прозрачным значениям f_1 и f_2 , как и в случае системы первого порядка, надо обратиться к понятию емкости программной системы относительно выносимых и вносимых дефектов. Оба вида дефектов одновременно существуют в одной и той же программной системе, из чего следует, что ее емкость относительно обоих видов дефектов должна быть одинаковой. Поэтому систему (3.7) можно переписать таким образом:

$$\begin{cases} \frac{df_1}{dt} = -A_1 \cdot f_1 - A_2 \cdot f_2 \\ \frac{df_2}{dt} = -A_2 \cdot f_1 - A_1 \cdot f_2 \end{cases}, \quad (3.8)$$

где $A_1 = \frac{G_{11}}{C} = \frac{G_{22}}{C}$, $A_2 = \frac{G_{12}}{C} = \frac{G_{21}}{C}$.

В этой системе коэффициент A_1 имеет ту же величину, что и соответствующий ему коэффициент влияния A_1 в уравнении (3.3), которое описывает динамику потока дефектов, выносимых из системы при отсутствии входного потока. Физический смысл коэффициента A_1 в уравнениях (3.3) и (3.8) одинаков – это коэффициент пропорциональности между количеством дефектов, которые образуют выходной или входной поток и интенсивностью этих же потоков. Поэтому коэффициент A_1 , в случае системы второго порядка, назовем собственным коэффициентом влияния. Коэффициент A_2 показывает перекрестное влияние выходного потока дефектов на входной поток, и наоборот, входного потока на выходной. Понятно, что такое взаимное влияние действительно существует в реальных системах. При увеличении интенсивности выходного потока дефектов растет вероятность того, что при их исправлении в ПС будут внесены дополнительные вторичные дефекты, т.е. к увеличению интенсивности входного потока. В этом заключается влияние выходного потока на входной. Одновременно, внесение в ПС вторичных дефектов увеличит общее количество дефектов в системе, что приведет к увеличению интенсивности выходного потока. Коэффициент A_2 в (3.8) назовем взаимным коэффициентом влияния.

Рассмотрим решение уравнений (3.8) относительно выходного и входного потоков дефектов в отдельности.

3.4.1. Динамика изменения выходного потока дефектов

Для нахождения динамики изменения выходного потока решим систему (3.8) относительно переменной f_1 . Для этого продифференцируем первое уравнение системы (3.8), а значение производной $\frac{df_2}{dt}$ возьмем из второго уравнения. Получим:

$$\frac{d^2 f_1}{dt^2} = -A_1 \cdot \frac{df_1}{dt} + A_2^2 \cdot f_1 + A_1 A_2 \cdot f_2.$$

Подставив в качестве f_2 его значение из первого уравнения, получим искомое уравнение:

$$\frac{d^2 f_1}{dt^2} + 2A_1 \cdot \frac{df_1}{dt} + (A_1^2 - A_2^2) \cdot f_1 = 0. \quad (3.9)$$

Его решение будем искать в виде:

$$f_1 = B_1 \cdot e^{p_1 t} + B_2 \cdot e^{p_2 t}, \quad (3.10)$$

где p_1 и p_2 - корни характеристического уравнения, а B_1 и B_2 - неизвестные пока постоянные интегрирования.

Характеристическое уравнение, соответствующее (3.9) имеет вид:

$$p^2 + 2A_1 \cdot p + (A_1^2 - A_2^2) = 0,$$

а его корни равны:

$$p_1 = -A_1 + A_2, \quad p_2 = -A_1 - A_2.$$

Для определения постоянных B_1 и B_2 воспользуемся начальными условиями исследуемого процесса:

$$\begin{cases} B_1 + B_2 = f_1(0) \\ (A_2 - A_1) \cdot B_1 - (A_2 + A_1) \cdot B_2 = \left. \frac{df_1}{dt} \right|_{t=0} \end{cases}. \quad (3.11)$$

Значение $f_1(0) = F_0$ - начальное количество дефектов в системе. Учитывая то, что $f_2(0) = 0$ (до начала работы с системой вторичные дефекты не могли быть внесены), из первого уравнения (3.8) получаем:

$$\left. \frac{df_1}{dt} \right|_{t=0} = -A_1 \cdot F_0.$$

Решая систему (3.11) при этих значениях свободных членов, получаем:

$$B_1 = B_2 = \frac{F_{10}}{2}.$$

С учетом этого

$$f_1 = \frac{F_0}{2} \cdot e^{A_2 t} \cdot e^{-A_1 t} + \frac{F_0}{2} \cdot e^{-A_2 t} \cdot e^{-A_1 t}.$$

После преобразований получим окончательно:

$$f_1 = F_0 \cdot e^{-A_1 t} \cdot ch(A_2 t). \quad (3.12)$$

Интенсивность выходного потока отказов $\lambda_1(t)$ можно выразить как:

$$\lambda_1(t) = A_1 \cdot F_0 \cdot e^{-A_1 t} \cdot ch(A_2 \cdot t) + A_2 \cdot F_0 \cdot e^{-A_1 t} \cdot sh(A_2 \cdot t). \quad (3.13)$$

Для анализа полученных результатов на рис. 3.2 показан график зависимости количества дефектов, находящихся в системе от времени при различных соотношениях $k = A_2/A_1$. Эти кривые построены для некоторой гипотетической программной системы с такими параметрами: начальное количество дефектов $F_0 = 100$, значение коэффициента $A_1 = 0,01 \text{ сут}^{-1}$, коэффициент k изменяется от 0 до 1,1. При этом $k = 0$ отвечает полному отсутствию потока вторичных дефектов, а значение $k = 1$ – случаю, когда исправление каждого выявленного дефекта сопровождается внесением одного нового вторичного. При значениях $k > 1$ количество вносимых вторичных дефектов превышает количество исправленных.

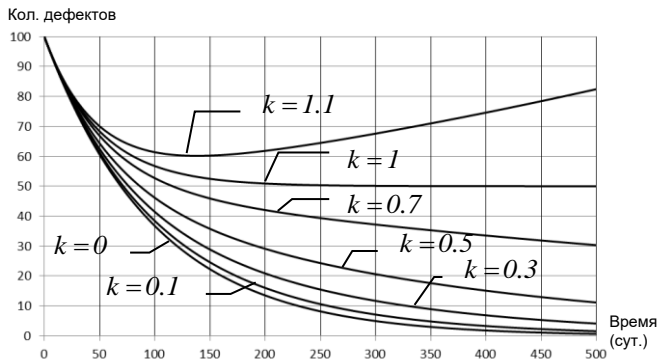


Рис. 3.2. Количество дефектов выходного потока в зависимости от $k = A_2/A_1$

Из анализа рис. 3.2 можно сделать вывод, что при увеличении коэффициента k количество дефектов, имеющих в системе, увеличивается. Обратим внимание, что при $k = 1$ это количество

стабилизируется на половинном значении от первоначального количества дефектов F_0 . При этом надо учитывать, что f_1 - это количество дефектов, которые будут **вынесены** из системы. Общее количество дефектов, содержащихся в системе определяется не только этими дефектами, но еще и дефектами, которые были **внесены** в нее обратным входным потоком. Поэтому для общего представления о процессе выявления дефектов надо исследовать динамику изменения во времени входного потока.

3.4.2. Динамика изменения входного потока дефектов

Для изучения динамики выявления дефектов входного потока решим систему (3.8) относительно переменной f_2 . По аналогии с преобразованиями п. 3.4.1, получим:

$$\frac{d^2 f_2}{dt^2} + 2A_1 \cdot \frac{df_2}{dt} + (A_1^2 - A_2^2) \cdot f_2 = 0. \quad (3.14)$$

Можно показать, что его решением будет функция:

$$f_2 = -F_0 \cdot e^{-A_1 t} \cdot sh(A_2 t). \quad (3.15)$$

Знак «минус» перед значением f_2 связан с обратным направлением входного потока относительно выходного. Действительно, дефектам, которые выносятся из программной системы традиционно приписывается знак плюс, а дефекты, которые были внесены, при этом должны учитываться со знаком минус. Однако, учитывая, что количество дефектов может быть только положительным, в дальнейшем при практических расчетах будем вычислять количество внесенных дефектов по модулю, то есть как

$$f_2 = F_0 \cdot e^{-A_1 t} \cdot sh(A_2 t). \quad (3.16)$$

Для интенсивности входного потока отказов $\lambda_2(t)$ получим:

$$\lambda_2(t) = A_1 \cdot F_0 \cdot e^{-A_1 t} \cdot sh(A_2 \cdot t) + A_2 \cdot F_0 \cdot e^{-A_1 t} \cdot ch(A_2 \cdot t). \quad (3.17)$$

Для интерпретации и анализа полученных результатов, на рис. 3.3 показан график зависимости количества дефектов, внесенных в ПС от времени при разных значениях $k = A_2/A_1$. Эти кривые построены для той же программной системы, что и показанные 3.2 с теми же параметрами: начальное количество дефектов $F_0 = 100$, значение коэффициента $A_1 = 0,01 \text{ сут}^{-1}$, коэффициент k изменяется от 0 до 1,1.

Анализируя график на рис. 3.3 можно сделать следующие выводы:

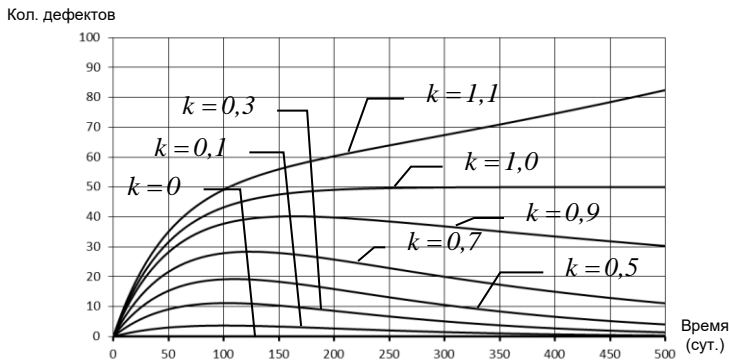


Рис. 3.3. Количество дефектов входного потока в зависимости от $k = A_1/A_2$

– При малых значениях коэффициента k интенсивность входного потока вторичных дефектов является незначительной в сравнении с интенсивностью выходного потока.

– Зависимость количества вторичных дефектов от времени имеет четко выраженный максимум, особенно при больших значениях k - от 0,5 до 0,7. Этот, не известный ранее факт, следует учитывать при планировании процесса тестирования ПС.

– При $k = 1$ поток вторичных дефектов стабилизируется на уровне $\frac{F_0}{2}$, то есть на половинном количестве дефектов, имеющихся в системе на начало исследования. Обратим внимание, что выходной поток, как видно из рис. 3.2, при этом значении k тоже достигает уровня $\frac{F_0}{2}$.

– При $k > 1$ количество вторичных дефектов растет во времени, что хорошо согласуется с общими представлениями о динамике выявления дефектов в этом случае.

В п. 3.4.1 и 3.4.2 получены зависимости для дефектов, образующих выходной поток и выносимых из системы, а также дефектов, вносимых входным потоком. На их основании можно построить аналогичные зависимости для общего количества дефектов, содержащихся в ПС.

3.4.3. Динамика изменения общего количества дефектов в ПС

Для получения выражения для количества дефектов, оставшихся в ПС в произвольный момент времени t , достаточно сложить количество

дефектов выходного и входного потоков. Действительно, формула (3.12) определяет количество дефектов, которые образуют выходной поток, но при этом *еще остаются* в системе. Формула (3.16) показывает количество вторичных дефектов, которые были внесены в систему и уже находятся в ней. Поэтому для количества дефектов, существующих в системе при одновременном действии обоих потоков (f_{12}), получим:

$$f_{12} = F_0 \cdot e^{-A_1 t} \cdot ch(A_2 t) + F_0 \cdot e^{-A_1 t} \cdot sh(A_2 t)$$

Учитывая, что $sh(A_2 t) + ch(A_2 t) = \frac{1}{2} \cdot e^{A_2 t}$ получаем окончательно:

$$f_{12} = F_0 \cdot e^{(A_2 - A_1) t}. \quad (3.18)$$

Графики зависимости f_{12} от времени, построенные при разных значениях k приведены на рис. 3.4.

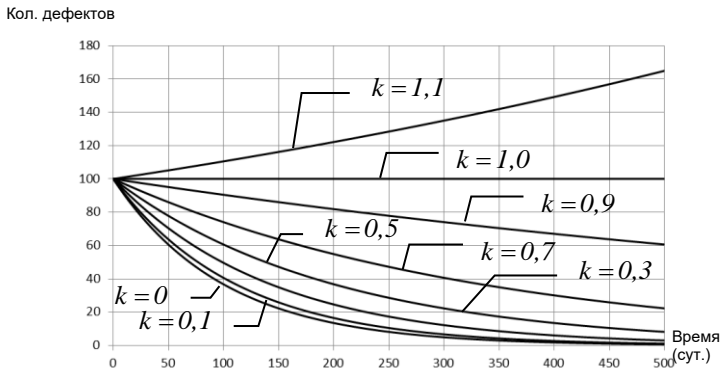


Рис. 3.4. Количество оставшихся в ПС дефектов в зависимости от $k = A_1/A_2$

Как видно из рис. 3.4, при $A_2 < A_1$ (интенсивность входного потока меньше интенсивности выходного) общее количество дефектов в ПС уменьшается. При обратном соотношении их общее количество, наоборот, увеличивается. Эти выводы теории ДПС хорошо согласуются с прогнозируемыми результатами.

Интересно, что при $A_2 = A_1$ из теории следует, что количество дефектов в системе остается постоянным и не зависит от времени. Этот факт тоже совпадает с прогнозируемым результатом. Действительно, при $A_2 = A_1$ на каждый выявленный и ликвидированный дефект приходится

один внесенный. То есть общее количество дефектов в системе, составляющее на начало исследований значение F_0 , должно сохраняться неизменным и в дальнейшем. Именно поэтому, как обращалось внимание ранее, выходной и входной потоки стабилизировались на половинном значении от первоначального количества дефектов.

Таким образом, прогнозируемое на основании ДПС поведение дефектов в программных системах совпадает с практикой, что подтверждает справедливость теории.

3.5. Кумулятивные тренды дефектов

Как отмечается в [1], несмотря на то, что дефекты образуют потоки с четко определенными закономерностями, процесс их выявления имеет значительную неопределенность. Поэтому случайный учет или не учет каждого отдельного дефекта снижает точность моделирования за счет имеющего место значительного «шума». В связи с этим МНПО, входными данными которых являются моменты выявления каждого отдельного дефекта, не могут обеспечить достаточную точность моделирования так как неточными уже являются их входные данные. Временной ряд, образующий кумулятивное количество дефектов (кумулятивный профиль отказов), является более точным за счет того, что случайный учет или не учет каждого конкретного дефекта не может отразиться на общей тенденции изменения этого ряда. Фактически кумулятивный профиль отказов уже освобожден от случайной составляющей и является трендом.

Обозначим кумулятивное количество дефектов, обнаруженное на момент времени t как $\mu(t)$. Его можно рассчитать как:

$$\mu(t) = -\int_0^t \lambda(t) \cdot dt.$$

Знак «минус» в этой формуле обусловлен тем, что производная $\lambda(t) = \frac{df}{dt}$ является отрицательной (количество дефектов со временем уменьшается).

Рассмотрим отдельно выражения для кумулятивного количества дефектов для случаев системы первого и второго порядка.

В системе первого порядка $\lambda_1(t) = A_1 \cdot F_0 \cdot e^{-A_1 t}$. Поэтому для кумулятивного количества дефектов получаем:

$$\mu(t) = -\int_0^t \lambda_1(t) \cdot dt = A_1 \cdot F_0 \cdot \int_0^t e^{-A_1 t} \cdot dt = -F_0 \cdot e^{-A_1 t} \Big|_0^t.$$

После преобразований получим окончательно:

$$\mu(t) = F_0 - F_0 \cdot e^{-A_1 t}. \quad (3.19)$$

Выражение (3.19) полностью совпадает с результатами, приведенными в [2] для всех экспоненциальных моделей надежности.

Для определения кумулятивного количества дефектов, выносимых выходным потоком из системы второго порядка, из выражения (3.13), раскрывая гиперболические функции, получаем:

$$\begin{aligned} \lambda_1 &= A_1 \cdot F_0 \cdot e^{-A_1 t} \cdot ch(A_2 \cdot t) + A_2 \cdot F_0 \cdot e^{-A_2 t} \cdot sh(A_2 \cdot t) = \\ &= \frac{F_0}{2} \left[(A_1 + A_2) \cdot e^{(A_2 - A_1)t} + (A_1 - A_2) \cdot e^{-(A_2 + A_1)t} \right]. \end{aligned} \quad (3.20)$$

Не трудно видеть, что при $A_1 = A_2$ первообразная от λ_1 не существует, поскольку разность $A_2 - A_1$ будет равняться нулю. Поэтому нахождение кумулятивного количества дефектов рассмотрим отдельно для двух случаев.

Случай 1. $A_1 \neq A_2$.

В этом случае первообразная от (3.20) существует всегда, поэтому имеем:

$$\mu_1(t) = \frac{F_0}{2} \cdot (A_1 + A_2) \cdot \int_0^t e^{(A_2 - A_1)t} \cdot dt + \frac{F_0}{2} \cdot (A_1 - A_2) \cdot \int_0^t e^{-(A_2 + A_1)t} \cdot dt$$

После интегрирования и подстановки пределов получим:

$$\mu_1(t) = \frac{F_0}{2} \cdot \left[\left(\frac{A_2 + A_1}{A_2 - A_1} \cdot e^{A_2 t} + \frac{A_2 - A_1}{A_2 + A_1} \cdot e^{-A_2 t} \right) \cdot e^{-A_1 t} - 2 \cdot \frac{A_1^2 + A_2^2}{A_2^2 - A_1^2} \right]. \quad (3.21)$$

Случай 2. $A_1 = A_2$.

В этом случае перед нахождением первообразной преобразуем выражение (3.20), учитывая, что $A_1 = A_2$. Получим: $\lambda_1 = \frac{F_{10}}{2} \cdot 2 \cdot A_1$, откуда:

$$\mu_1(t) = F_{10} \cdot A_1 \cdot \int_0^t dt.$$

После интегрирования получаем:

$$\mu_1(t) = F_{10} A_1 \cdot t. \quad (3.22)$$

При $A_1 = A_2$ на каждый дефект, который выносится из программной системы приходится один вторичный дефект, который вносится в нее. Поэтому общее количество дефектов в системе остается неизменным, что приводит, в силу предположения 4 (см. п. 3.1.2), к неизменной скорости выноса дефектов. Поэтому, предсказываемая теорией ДПС линейная зависимость кумулятивного тренда дефектов от времени при $A_1 = A_2$ полностью совпадает с наблюдаемой экспериментально.

Для кумулятивного количества дефектов, вносимых в систему второго порядка входным потоком, с учетом выражения (3.17), получаем аналогично:

$$\mu_2(t) = \frac{F_0}{2} \cdot \left[\left(\frac{A_2 + A_1}{A_2 - A_1} \cdot e^{A_2 t} - \frac{A_2 - A_1}{A_2 + A_1} \cdot e^{-A_2 t} \right) \cdot e^{-A_1 t} - 4 \cdot \frac{A_1 \cdot A_2}{A_2^2 - A_1^2} \right] \quad (3.23)$$

при $A_1 \neq A_2$ и

$$\mu_2(t) = F_0 A_1 \cdot t \quad (3.24)$$

при $A_1 = A_2$.

Выражение (3.24), полученное для входного потока дефектов, полностью совпадает с аналогичным выражением (3.22), полученным для выходного потока. Этот результат тоже является ожидаемым и полностью отвечает физическому смыслу рассматриваемых процессов.

3.6. Технология и примеры оценки надежности программных систем

На основе результатов, полученных в теории ДПС, разработана модель надежности программного обеспечения, свободная от недостатков, присущим известным моделям. Во-первых, эта модель органично учитывает влияние вторичных дефектов, а во-вторых, как будет показано далее, обеспечивает высокую точность моделирования независимо от конкретной ПС.

Для моделирования используется полученное ранее выражение для кумулятивного профиля выявленных дефектов (3.21) или (3.22).

Входными данными модели является экспериментально полученный кумулятивный профиль дефектов, выявленных на известном временном промежутке. Определению подлежат параметры модели – коэффициенты влияния A_1 и A_2 , а также начальное количество дефектов в системе F_0 .

Технология определения параметров предусматривает два этапа. На первом выполняется упрощенная предварительная оценка параметров, а на втором – их уточнение.

На этапе предварительной оценки будем считать, что входной поток вторичных дефектов отсутствует, поэтому коэффициент влияния $A_2 = 0$. В этом случае кумулятивная зависимость выявленных дефектов представляет собой экспоненту (3.19):

$$\mu(t) = F_0 - F_0 e^{-A_1 t}.$$

Из теории переходных процессов [15] известно, что в этой зависимости величина, обратная коэффициенту A_1 , является постоянной времени процесса $\tau = \frac{1}{|A_1|}$ и может быть графически выражена как подкасательная к экспоненте (рис. 3.5).

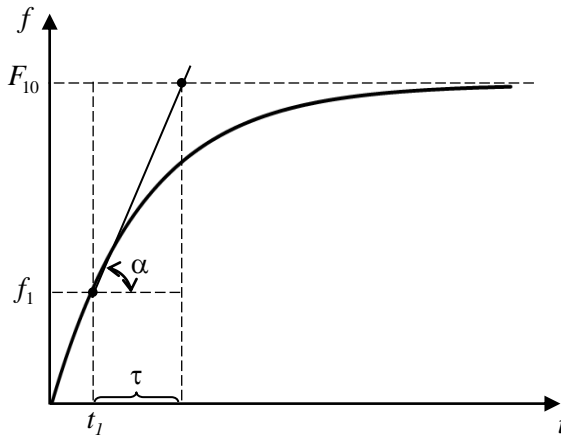


Рис. 3.5. К определению постоянной времени процесса

Из рис. 3.5 следует, что величина подкасательной может быть определена как

$$\tau = \frac{F_0 - f_1}{tg\alpha},$$

где угловой коэффициент $tg\alpha$ определяется из формулы $tg\alpha \approx \frac{f_2 - f_1}{t_2 - t_1}$.

Здесь f_2 – наблюдаемое значение кумулятивного профиля в момент времени t_2 , непосредственно следующий после предыдущего – t_1 .

Для повышения точности расчетов следует определять угловой коэффициент $tg\alpha$ для каждой двух последовательных точек экспериментального кумулятивного профиля. В качестве неизвестной величины F_0 можно использовать последнюю точку кумулятивного профиля f_n . Определенные таким образом для каждой последовательной пары точек значения τ следует усреднить. Из найденного среднего значения τ_{cp} находится приближение для коэффициента влияния A_1

$$A_1 = \frac{1}{\tau_{cp}}. \quad (3.25)$$

Приближенное значение для F_0 может быть получено для каждой точки кумулятивного профиля из (3.19) как :

$$F_0 = \frac{f_i}{1 - e^{-A_1 t_i}}. \quad (3.26)$$

Для повышения точности найденные таким образом значения F_0 усредняем по всем точкам экспериментального кумулятивного профиля.

Следует учесть, что, полагая коэффициент $A_2 = 0$, то есть исключая влияние входного потока, мы получаем завышенные оценки для F_0 и A_1 .

На втором этапе, после определения приближений для F_0 и A_1 производится итерационное уточнение этих значений, а также определение коэффициента A_2 . При этом необходимо учитывать зависимость $\mu_1(t)$ (3.21) от коэффициентов A_1 и A_2 . Эта зависимость показана на рис. 3.6.

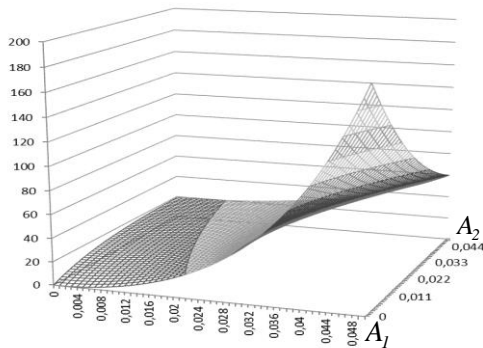


Рис. 3.6. Зависимость $\mu_1(t)$ от коэффициентов влияния

Из рис. 3.6 следует, что кумулятивный профиль $\mu_1(t)$ более чувствителен к изменению коэффициента A_1 , нежели A_2 . Поэтому уточнение параметров модели надо начать именно с коэффициента A_2 : допущенные здесь ошибки незначительно скажутся на точности вычислений.

Для оценки точности определения параметров используем критерий среднего квадратичного отклонения (СКО):

$$СКО = \frac{\sum_{i=1}^n (f_{io} - f_{ic})^2}{n}, \quad (3.27)$$

где n – количество точек в экспериментальной кумулятивной кривой; f_{io} – наблюдаемое (observed) значение i -й точки кривой; f_{ic} – вычисленное (calculated) с данными параметрами значение.

Для уточнения параметров модели предлагается следующий алгоритм:

Шаг 1. Определить начальное значение $СКО_b$ при полученных по формулам (3.25) и (3.26) приближениях для A_1 и F_0 . При этом положить $A_2 = 0$.

Шаг 2. Изменяя в формуле (3.21) коэффициент A_2 с шагом $\Delta A_2 = \frac{1.5A_1}{10}$, пока текущее значение $СКО_x$ меньше, чем $СКО_b$, получить уточненное значение для A_2 . Положить $СКО_b = СКО_x$.

Шаг 3. Изменяя в формуле (3.21) значение F_0 в диапазоне от 0,5 до 1,5 от полученного в шаге 1 приближения, пока текущее значение $СКО_x$ меньше, чем $СКО_b$, найти уточненное значение для F_0 . Положить $СКО_b = СКО_x$.

Шаг 4. Изменяя коэффициент в (3.21) коэффициент A_1 в диапазоне от 0,5 до 1,5 от полученного в шаге 1 приближения, пока текущее значение $СКО_x$ меньше, чем $СКО_b$, найти уточненное значение для A_1 . Положить $СКО_b = СКО_x$.

Шаг 5. Повторить шаги 2 – 4 для следующего этапа уточнения. Процесс уточнения считается законченным, если на очередном этапе достигнуто значение $СКО_b \leq \varepsilon$, где ε – заданная точность расчета.

Для оценки точности моделирования по разработанной МНПО и сравнения ее с существующими моделями были использованы данные по выявленным дефектам в двадцати различных программных системах [2, 16]. Для увеличения точности моделирования данные по выявлению дефектов в каждом кумулятивном профиле разбивались на интервалы, в течение которых закон их изменения во времени оставался неизменным. Всего обработано 123 временных ряда, и выполнено 1230 оценок показателей надежности. Для сравнения точности моделирования, наряду с МНПО на основе ДПС, были взяты основные известные модели надежности, охватывающие все существующие классы моделей. В их основе лежат разные законы распределения вероятностей, что дает возможность объективно оценить точность модели ДПС. В качестве исследуемых взяты модели Желинского-Моранды, неоднородного пуассоновского процесса, Шнайдервинда, Мусы, распределения Вейбулла, S-образная модель, модель Дюэна [17], геометрическая модель Моранды [18] и логарифмическая модель Мусы-Окумото. Результаты моделирования показаны на диаграмме (рис. 3.7).

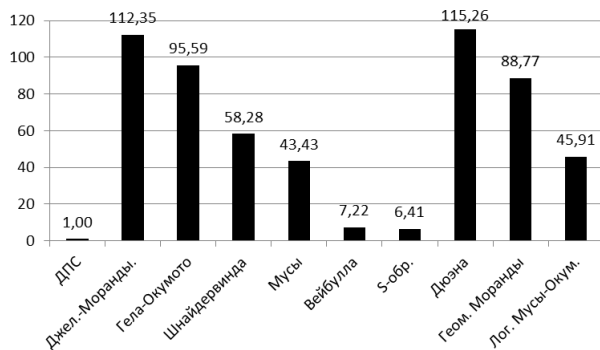


Рис. 3.7. Результаты сравнения различных МНПО

На диаграмме по оси ординат приведены значения СКО, полученные в результате проверки точности моделирования различными моделями. Для удобства СКО модели ДПС принято равным единице. Из диаграммы видно, что для всех исследованных программных систем модель ДПС показала результат, по точности более чем в шесть раз превосходящий наилучший результат S-образной модели. Эти результаты наглядно свидетельствуют о том, что теория ДПС может служить единым теоретическим и методологическим базисом оценки и моделирования надежности программных систем.

Выводы и вопросы для самоконтроля

В разделе описан новый концептуальный подход к проблеме моделирования надежности программных систем. Этот подход отличается от известных тем, что рассматривает выявление и исправление дефектов как неравновесный процесс, возникающий при взаимодействии ПС с внешней средой. Модели надежности программных систем, основанные на традиционном вероятностном подходе, моделируют только выявление первичных дефектов и сталкиваются с большими трудностями при попытке моделирования вторичных. Разработанная теория динамики программных систем позволила органично объединить и объяснить закономерности возникновения как первичных, так и вторичных дефектов. Созданная на базе этой теории модель надежности прошла экспериментальную проверку и показала значительно более высокую точность в сравнении с существующими. Существенным достоинством разработанной модели является то, что она показывает стабильно высокие результаты для всех исследованных программных систем.

Это дает основание утверждать, что теория динамики программных систем ликвидирует тот существенный разрыв между теорией и практикой, о котором говорилось в начале раздела.

В данном разделе изложены материалы, позволяющие получить ответы на следующие вопросы:

1. Назовите концептуальную основу построения известных моделей надежности ПС.
2. Каковы недостатки существующих моделей надежности?
3. В чем состоит основная идея теории ДПС?
4. Что такое поток дефектов?
5. В чем состоит различие между выходным и входным потоком дефектов?
6. Как в теории ДПС учитываются вторичные дефекты?
7. Что такое коэффициенты влияния и каков их физический смысл?
8. Запишите и объясните основные уравнения ДПС.
9. Почему члены в правой части основных уравнений ДПС входят со знаком «минус»?
10. Запишите и объясните решение основных уравнений для выходного потока.
11. Запишите и объясните решение основных уравнений для входного потока.

12. Какой физический смысл имеет знак «минус» в решении для выходного потока?
13. Как физически можно объяснить поведение выходного и входного потоков при равенстве коэффициентов влияния?
14. Что такое кумулятивный профиль отказов?
15. В чем заключается моделирование надежности на основе теории ДПС?

Литература

1. Харченко В.С. Методы моделирования и оценки качества и надежности программного обеспечения / В.С. Харченко, В.В. Скляр, О.М. Тарасюк. Учеб. пособие. — Харьков: Нац. аэрокосм. ун-т ХАИ, 2004. — 159 с.
2. Lyu M.R. Handbook of Software Reliability Engineering, McGraw-Hill Company, 1996. — 805 p.
3. Эткин В.А. К единой теории реальных процессов [Электронный ресурс]. / В. А. Эткин. — Режим доступа к журналу: http://zhurnal.lib.ru/e/etkin_w_a/kedinoyteoriirealnychprozessov.shtm
4. Управление качеством продукции. Основные понятия. Термины и определения (СТ СЭВ 3519-81) : ГОСТ 15467-79. — [Чинний від 1979-07-01]. — М. : Госком СССР по стандартам. — 42 с. — (Міждержавний стандарт).
5. Николис Г. Самоорганизация в неравновесных системах. От диссипативных структур к упорядоченности через флуктуации. / Г. Николис, И. Пригожин. // М.: Мир, 1979 — 512 с
6. Пригожин И. Современная термодинамика. От тепловых двигателей до диссипативных структур. / И. Пригожин, Д. Конденуди. Пер. с англ. Под ред. проф. Е.П. Агеева. // М.: Мир, 2002. — 461 с.
7. Moranda P.B. Final Report of Software Reliability Study. — / P.B. Moranda, J. Jelinski // McDonnell Douglas Astronautics Company. MDC Report № 63921, dec. — 1972. — 51 с.
8. Goel, A.L., Time-Dependent Error-Detection Rate Model for Software and Other Performance Measures / A.L. Goel, K. Okumoto // IEEE Transactions on Reliability, v. R-28, № 5, August. — 1979. — P. 206 — 211.
9. Schneidewind, N.F. Software Reliability Model with Optimal Selection of Failure Data / N.F. Schneidewind //IEEE Transactions on

- Software Engineering. — Vol. 19. — No. 11. Nov. — 1993. P. 1095–1104.
10. Musa J.D. Validity of Execution time theory of software reliability // IEEE Trans. on reliability. — 1979. — № 3. — P.199–205.
 11. Quadri, S. M. K. Software Reliability Growth Modeling with New Modified Weibull Testing—effort and Optimal Release Policy / S. M. K. Quadri, N. Ahmad // International Journal of Computer Applications. — Vol. 6. — 2010. —№ 12. — С. 1–10.
 12. Yamada, S. S-Shaped Reliability Growth Modeling for Software Error Detection / S. Yamada, M. Ohba, S. Osaki //IEEE Transactions on Reliability. Vol. R-32. No. 5, Dec. —1983. — P. 475—478.
 13. Musa J.D. Okumoto K. A Logarithmic Poisson Time Model for Software Reliability Measurement /Proc. Sevent International Conference on Software Engineering. — Orlando, Florida: — 1984. — P. 230—238.
 14. Гроот С. Неравновесная термодинамика / С. Гроот, П. Мазур // М.: Наука, 1964. — 456 с.
 15. Маєвський Д. А. Електричне моделювання процесу виявлення помилок в програмному забезпеченні інформаційних систем. /Д. А. Маєвський // Сб. Електромашиностроение и электрооборудование. —2010. — Вып. 75, — С. 113—116.
 16. Android – An Open Handset Alliance Project [Электронный ресурс]. Режим доступа:<http://code.google.com/p/android/issues/list>
 17. Duan J.T. Learning Curve Approach to Reliability Monitoring // IEEE Trans. on Aero-space. — 1964. — Vol. 2. — P. 563—566.
 18. Moranda, P.B. Event-Altered Rate Models for General Reliability Analysis / P.B. Moranda // IEEE Transactions on Reliability. Vol.R-28. — No. 5. — 1979. — С.376—381