

СТРАТЕГИИ ФУНКЦИОНАЛЬНОЙ ПЕРЕСТРОЙКИ ОТКАЗОУСТОЙЧИВЫХ ИНФОРМАЦИОННЫХ СИСТЕМ ПРИ РАЗЛИЧНЫХ ВИДАХ ДЕГРАДАЦИИ

На различных этапах своего жизненного цикла информационные системы (ИС) подвергаются воздействию неблагоприятных факторов. Из-за этого либо могут ухудшаться характеристики самой системы, либо может нарушаться логика процесса решения возложенных на нее задач, что приводит к деградации ИС [1]. Деградацию будем подразделять на эволюционную и эксплуатационную.

Под **эволюционной деградацией** понимается рассогласование между характеристиками ИС, заложенными при ее проектировании, и постоянно растущими требованиями к этим характеристикам, что стимулируется появлением новых задач. Такое рассогласование определяет моральное старение системы. Основным путем противодействия эволюционной деградации ИС — использование при проектировании максимально прогрессивных архитектурных и схемотехнических решений, а также модернизация системы в процессе эксплуатации.

Под **эксплуатационной деградацией** понимается либо ухудшение характеристик самой системы, либо уменьшение количества решаемых задач, либо снижение качества их решения, проявляемые в процессе ее функционирования. Эксплуатационная деградация (далее — просто деградация) вызывается воздействием на ИС различных неблагоприятных факторов.

Основные понятия

Введем ряд следующих понятий.

Модуль системы — совокупность аппаратно-программных средств, способных полностью выполнять одну или несколько функций, возложенных на ИС.

Функциональные возможности модуля — количество выполняемых модулем функций (функциональные возможности модулей определяют функциональные возможности системы).

Универсальный модуль — модуль, способный выполнять более двух функций.

Специализированный модуль — модуль, способный выполнять единственную функцию.

Полный функциональный отказ модуля — потеря модулем способности выполнять все возложенные на него функции.

Частичный функциональный отказ модуля — потеря модулем способности выполнять часть возложенных на него функций.

Функциональная перестройка ИС — процесс перераспределения возложенных на систему функций между работоспособными модулями.

Допустимый вариант распределения функций — вариант распределения функций между модулями системы, при котором все они выполняются одновременно.

Работоспособное функциональное состояние ИС — состояние системы, для которого существует допустимый вариант распределения функций.

Виды деградации ИС

Будем различать структурную, функциональную и структурно-функциональную деградацию, обозначив их соответственно как S -, F - и SF -деградация.

Структурной деградацией ИС называется увеличение числа полностью отказавших модулей системы без изменения ее функциональных возможностей.

Структурная деградация присуща системам, содержащим как только специализированные (S -системы), так и специализированные и универсальные модули одновременно (US -системы). Рассмотрим ситуации, когда она имеет место.



В *VS*-системах структурная деградация возникает тогда, когда какой-либо специализированный модуль после возникновении функционального отказа в системе, находящейся в произвольном состоянии s_v , теряет способность выполнения функции, закрепленной за ним, но при этом в системе, перешедшей в состояние s_{v+1} , существует резервный модуль, способный выполнять эту функцию.

В *US*-системах структурная деградация имеет место тогда, когда возникает полный функциональный отказ либо резервного специализированного модуля, либо основного, но существует некоторый модуль, способный выполнить его функцию (специализированный резервный, либо универсальный резервный, либо универсальный основной). Если этот специализированный модуль резервный, то система выполняет возложенные на нее функции при помощи основных модулей, а если он основной, то происходит передача функции данного модуля, либо резервному, либо основному универсальному в соответствии с выбранным вариантом допустимого распределения функций в системе.

Кроме того, в *US*-системах структурная деградация возможна тогда, когда возникает полный функциональный отказ резервного универсального либо основного универсального модуля, но при этом существует некоторый модуль, способный выполнить его функцию (специализированный резервный, либо универсальный резервный, либо универсальный основной).

В отношении возникновения полных функциональных отказов универсальных модулей необходимо сделать следующее замечание. Универсальный модуль способен выполнять не менее двух функций из множества функций, возложенных на систему. Предположим, что после некоторой совокупности функциональных отказов универсальный модуль обладает способностью выполнять две функции из множества изначально возможных. Если далее возникает очередной отказ, вследствие которого этот модуль теряет способность выполнения одной из двух оставшихся функций, то он уже представляет собой специализированный модуль. В связи с этим полностью потерять способность выполнения функций после возникновения некоторой совокупности функциональных отказов универсальный модуль может лишь тогда, когда перед последним отказом из этой совокупности он обладает способностью выполнения какой-либо одной функции, т. е. когда он, по сути, является специализированным.

Функциональной деградацией ИС называется снижение ее функциональных возможностей без увеличения числа полностью отказавших модулей системы.

Функциональная деградация имеет место либо в резервированных системах, состоящих только из универсальных модулей (*U*-системы), либо в резервированных *US*-системах. Она может быть скрытой и явной.

Скрытая функциональная деградация заключается в снижении функциональных возможностей какого-либо модуля без изменения функциональных возможностей системы.

Скрытая *F*-деградация вызывается возникновением функционального отказа в системе, находящейся в произвольном состоянии s_v , в результате которого она переходит в состояние s_{v+1} с потерей способности выполнения модулем некоторой функции, способностью выполнения которой система обладала в состоянии s_v .

Явная функциональная деградация заключается в снижении функциональных возможностей какого-либо модуля, вызывающем снижение функциональных возможностей всей системы. Это происходит тогда, когда в ИС, находящейся в произвольном состоянии s_v , возникает функциональный отказ и в результате этого она переходит в состояние s_{v+1} . При этом в состоянии s_{v+1} не существует ни одного допустимого варианта распределения функций между модулями.

Явная функциональная деградация выражается в возникновении следующих событий или их различных комбинаций:



- сокращение числа выполняемых системой функций;
- снижение качества выполнения некоторых (всех) функций без изменения их числа (упрощение алгоритмов выполнения функций, снижение точности решения задач, уменьшение количества обрабатываемых параметров и т. п.);
- увеличение времени выполнения функций.

В соответствии с ними явную функциональную деградацию будем подразделять на следующие виды: F_n -деградацию; F_q -деградацию; F_l -деградацию; F_{nq} -деградацию; F_{nl} -деградацию; F_{qt} -деградацию; F_{nqt} -деградацию.

Структурно-функциональной деградацией ИС называется увеличение числа полностью отказавших модулей системы с одновременным снижением ее функциональных возможностей.

Структурно-функциональная деградация возникает в нерезервированных S - и US -системах.

Данный вид деградации в нерезервированных S -системах возникает в ситуации, когда в состоянии s_v происходит функциональный отказ произвольного модуля.

В US -системах структурно-функциональная деградация возникает в ситуации, когда в состоянии s_v происходит функциональный отказ специализированного модуля.

Структурно-функциональная деградация выражается в возникновении событий, соответствующих явной функциональной деградации.

В связи с этим структурно-функциональную деградацию, так же как и явную функциональную деградацию, будем подразделять на следующие виды: SF_n -деградацию; SF_q -деградацию; SF_l -деградацию; SF_{nq} -деградацию; SF_{nl} -деградацию; SF_{qt} -деградацию; SF_{nqt} -деградацию.

Схема классификации различных видов деградации ИС приведена на рис. 1.



Рис. 1. Виды деградации информационных систем

Стратегии функциональной перестройки ИС

Эффективным противодействием деградации является функциональная перестройка ИС. Она предполагает формирование и реализацию стратегий, соответствующих определенным видам деградации: β_n^- , β_q^- , β_t^- , β_{nq^-} , β_{nt^-} , β_{qt^-} , β_{nqt^-} -стратегий. При этом проводятся анализ ситуации, определение спектра сохраняемых в системе функций в условиях воздействия неблагоприятных факторов, возможная корректировка состава функций и алгоритмов их реализации.

При формировании стратегий функциональной перестройки должны быть решены следующие основные задачи [2]: определение множества состояний $S = \{s_v\}$, в которых необходимо осуществлять перестройку и формирование собственно вариантов функциональной перестройки в состояниях множества S . Кроме того, дополнительно для каждого состояния множества S необходимо решить задачу выработки решения относительно функций, возложенных на ИС: «отбрасывать» ли некоторые функции, упрощать алгоритмы их реализации или увеличивать время их выполнения.

Для описания решений, которые принимаются в различных состояниях системы по отношению к функциям, возложенным на нее, воспользуемся следующим подходом.

Пусть $M_r^{(v)}$ — множество модулей, сохранивших способность в состоянии s_v выполнять закрепленные за ними в состоянии s_{v-1} функции, а $M_f^{(v)}$ — множество модулей, потерявших такую возможность. Пусть также функции, связанные с множеством $M_r^{(v)}$, образуют множество $F_r^{(v-1)}$, а функции, связанные с множеством $M_f^{(v)}$, образуют множество $F_f^{(v-1)}$.

Тогда при переходе системы из состояния s_{v-1} в состояние s_v по отношению к функциям множеств $F_r^{(v-1)}$ и $F_f^{(v-1)}$ могут быть приняты следующие решения:

- $a_r(a_f)$ — все функции множества $F_r^{(v-1)}$ ($F_f^{(v-1)}$) сохраняются в системе без изменений;
- $b_r(b_f)$ — все функции множества $F_r^{(v-1)}$ ($F_f^{(v-1)}$) сохраняются в системе, но алгоритмы выполнения некоторых из них (всех) упрощаются;
- $c_r(c_f)$ — все функции множества $F_r^{(v-1)}$ ($F_f^{(v-1)}$) отбрасываются;
- $d_r(d_f)$ — часть функций множества $F_r^{(v-1)}$ ($F_f^{(v-1)}$) отбрасывается, остальные сохраняются без изменений;
- $e_r(e_f)$ — часть функций множества $F_r^{(v-1)}$ ($F_f^{(v-1)}$) отбрасывается, остальные сохраняются, причем алгоритмы выполнения некоторых из них (всех) упрощаются.

Подходы к формированию стратегий определяются множеством Π всех возможных комбинаций указанных выше решений для функций множеств $F_r^{(v-1)}$ и $F_f^{(v-1)}$.

Решения $\pi_i \in \Pi$, где $i \in \{1, 2, \dots, 25\}$, являются основой для определения отображения:

$$\xi : F^{(v-1)} \rightarrow F^{(v)}, \quad (1)$$

где $F^{(v)}$ и $F^{(v-1)}$ — множества функций, выполняемых системой в состояниях s_v и s_{v-1} соответственно.

Реализация отображения (1), с учетом сформированного множества S , позволяет определить вариант функциональной перестройки ИС. Рассмотрим особенности его формирования. Стратегии функциональной перестройки основаны на реализации всего спектра решений $\pi_i \in \Pi$. Назовем β_n^- , β_q^- , β_t^- -стратегии чистыми, а β_{nq^-} , β_{nt^-} , β_{qt^-} , β_{nqt^-} -стратегии смешанными.

Формирование β_n^- -стратегий

Формирование β_n^- -стратегий основано на реализации решений π_3 , π_4 , π_{11} , π_{14} , π_{16} , π_{18} , π_{19} , любое из которых предполагает «отбрасывание» некоторых функций (без упрощения алгоритмов выполнения «сохраняемых» функций) в силу невозможности их выполнения из-за снижения ресурсной мощности системы.



Формирование β_n -стратегий осуществляется в два этапа:

1. Определение состава функций, «сохраняемых» в произвольном состоянии s_v ;
2. Определение варианта функциональной перестройки (составление плана распределения «сохраняемых» функций между модулями системы).

Определение состава функций, «сохраняемых» в произвольном состоянии s_v , имеет свои особенности для различных классов систем и видов функциональных отказов.

Для US -систем возможны два случая: либо происходит потеря способности выполнения какой-либо функции всеми модулями системы, либо происходит потеря способности выполнения всех функций каким-либо модулем. Решение, какая именно функция исключается из системы, для первого случая очевидно, а во втором случае определяется возможностями выполнения функций модулями системы: исключается та функция, способностью выполнения которой обладает наименьшее количество модулей.

Для U -систем также рассмотрим два случая. В первом случае предположим, что в системе происходят частичные функциональные отказы ее модулей, а для второго случая сделаем предположение о том, что в системе происходят только полные функциональные отказы модулей.

Для случая, когда в системе происходят частичные функциональные отказы, может возникнуть ситуация, при которой система из работоспособного функционального состояния s_{v-1} переходит в работоспособное функциональное состояние s_v , причем ни один из ее модулей полностью не отказал (но нет ни одного допустимого варианта распределения функций). В новом состоянии s_v дальнейшее функционирование системы без деградации невозможно. При этом необходимо снижать порядок системы — уменьшать на единицу число ее модулей. Решение, какая именно функция исключается из системы, может быть принято на основании ранжирования функций по степени их важности. Таким образом, формирование β_n -стратегий в этом случае заключается в следующем. Если все функции имеют одинаковый ранг, то в состоянии s_v исключается та, которая связана с последним частичным функциональным отказом, а остальные продолжают выполняться в соответствии с распределением, принятым в состоянии s_{v-1} . Если же функции имеют различные ранги, то в состоянии s_v исключается функция, обладающая наименьшим рангом, а для остальных при необходимости формируется новое распределение.

В случае, когда возникают только полные функциональные отказы модулей, определение состава функций, «сохраняемых» в произвольном состоянии s_v , происходит следующим образом.

Все функции ранжируются по степени важности, т. е. им присваиваются определенные «веса» w_i : функции $f_1 - w_1, f_2 - w_2$ и т. д., функции $f_{N_{v-1}} - w_{N_{v-1}}$, где N_{v-1} — количество функций, выполняемых системой в состоянии $s_{v-1}^{(i)}$. Физическая интерпретация «веса» функции может быть различной. Например, в качестве «веса» может выступать погрешность выполнения функции, количество обрабатываемых параметров и т. д. Далее из этих функций формируется множество K_i всевозможных их комбинаций, причем количество функций, входящих в эти комбинации, варьируется от одной до $N_{v-1} - 1$.

Каждая комбинация $K_f^{(i)}$ из множества K_i также обладает определенным «весом» $W^{(i)}$, равным композиции «весов» входящих в эту комбинацию функций. Если при определении состава функций, «сохраняемых» в системе в произвольном состоянии s_v , не выдвинуто требование сохранить какие-либо конкретные функции, то из множества K_i выбирается комбинация с максимальным «весом» и временем выполнения функций, не превышающим максимально допустимое. При необходимости сохранения конкретных функций в системе выбирается комбинация с максимальным «весом», включающая эти функции. При этом учитываются требования, предъявляемые ко времени выполнения функций.



Распределение функций выбранной комбинации между модулями системы (выбор варианта функциональной перестройки) представляет собой задачу теории расписаний. Задача составления оптимального расписания должна быть решена для всех состояний $s_v \in S$. Совокупность найденных таким образом вариантов распределения функций представляет собой искомую β_n -стратегию.

Формирование β_q -стратегий

Формирование β_q -стратегий основано на реализации решений $\pi_2, \pi_5, \pi_6, \pi_7, \pi_8, \pi_9, \pi_{10}, \pi_{12}, \pi_{15}, \pi_{17}, \pi_{20}, \pi_{21}, \pi_{22}, \pi_{23}, \pi_{24}, \pi_{25}$ [3]. Данные решения предполагают упрощение алгоритмов выполнения функций в силу невозможности реализации «полных» алгоритмов их выполнения из-за снижения ресурсной мощности системы. Формирование β_q -стратегий, так же как и в случае β_n -стратегий, осуществляется в два этапа:

1. Определение состава упрощенных алгоритмов выполнения функций и степени их упрощения.

2. Формирование варианта функциональной перестройки (составление плана распределения функций с упрощенными алгоритмами выполнения между модулями системы).

При определении состава упрощенных алгоритмов выполнения функций необходимо учитывать следующее обстоятельство. Выполнение ряда функций (особенно это характерно для управляющих систем) производится при помощи так называемых нониусных схем. При реализации таких схем предполагается, что функция выполняется в за несколько этапов, каждый из которых направлен на улучшение результата предыдущего, в частности его точности, причем выполнение функции может быть завершено, в принципе, на любом из них (все зависит от требований, предъявляемых к точности выполнения).

Учитывая наличие возможности многоэтапного выполнения ряда функций, под упрощением алгоритмов их выполнения будем понимать отказ от некоторых этапов выполнения какой-либо функции (функций). При этом так как этапы выполнения осуществляются строго последовательно, то их «отбрасывание» в случае необходимости для конкретной функции реализуется также строго определено: это происходит, начиная с последнего этапа.

Для каждой функции процедура упрощения алгоритма ее выполнения имеет свой конкретный физический смысл: в одном случае это снижение точности решения задачи, в другом — уменьшение количества обрабатываемых параметров и т. д. Например, если выполнение функции производится в несколько этапов, каждый из которых направлен на повышение точности предыдущего, то реализация функции может быть закончена, в принципе, на любом из них. В этом случае под упрощением алгоритма понимается невыполнение конечных этапов реализации функции.

В ряде случаев под упрощением алгоритмов выполнения функций понимается переход на другие алгоритмы, менее точные, но и менее трудоемкие в смысле вычислительных затрат. Например, это может иметь место при решении задач расчета параметров движения различных подвижных объектов, когда из-за снижения ресурсной мощности системы приходится увеличивать шаг интегрирования при определении местоположения объекта по измерениям его скорости или ускорения и т. п.

Будем полагать, что функциональные возможности ИС определяются значениями времени $T^{(v)}$ и погрешности выполнения $\sigma^{(v)}$ функций.

Время $T^{(v)}$ определяется сложностью алгоритмов выполнения функций множества F^v и производительностью системы в состоянии s_v . Номинальные значения $T_{ном}$, $\sigma_{ном}$ и предельно допустимые значения $T_{доп}$, $\sigma_{доп}$ этих величин определяют область работоспособных функциональных состояний как множество таких состояний, для которых справедливы соотношения:

$$T_{ном} \leq T^{(v)} \leq T_{доп}, \sigma_{ном} \leq \sigma^{(v)} \leq \sigma_{доп}.$$



Функционирование ИС в этой области может быть осуществлено путем выбора определенного набора функций с упрощенными алгоритмами их выполнения и надлежащей организации функциональной перестройки для любого из состояний $s_v \in S$. При этом она основывается на выборе подходящей альтернативы для каждого состояния $s_v \in S$ среди допустимых вариантов функциональной перестройки, представляющих собой множество $U_{s_v}^{(don)} = \{ {}^{(i)}U_{s_v}^{(don)} \}$. Это множество формируется следующим образом.

Пусть задан вектор внешних характеристик системы $\Psi = \{\psi_1, \dots, \psi_L\}$, т. е. характеристик выполнения функций с требуемым качеством в условиях функциональных отказов. Такими характеристики в данном случае являются погрешность $\sigma^{(v)}$ и время $T^{(v)}$ выполнения функций в произвольном состоянии $s_v \in S$. Внешние характеристики находятся в функциональной зависимости от возможных вариантов функциональной перестройки, образуя собой множество $U_{s_v}^{(воз)} = \{ {}^{(i)}U_{s_v}^{(воз)} \}$.

Элементы множества $U_{s_v}^{(воз)}$ формируются на основе различных комбинаций алгоритмов выполнения функций с учетом различной степени их упрощения. При определении степени упрощения алгоритмов в произвольном состоянии s_v каждой из функций ставится в соответствие некоторое множество алгоритмов ее выполнения:

$$f_i \Leftrightarrow A_i = \{a_i^{(1)}, a_i^{(2)}, \dots, a_i^{(m_i)}\}; i \in \{1, \dots, N_{v-1}\},$$

где N_{v-1} — количество функций, выполняемых системой в состоянии s_{v-1} ;

m_i — число степеней упрощения алгоритма выполнения функции f_i .

Все множества A_i входят подмножествами в множество A алгоритмов выполнения функций, возложенных на систему, с различными степенями упрощения. Каждому множеству A_i соответствует множество $\Sigma_i = \{\sigma_i^{(1)}, \sigma_i^{(2)}, \dots, \sigma_i^{(m_i)}\}$, элементы которого характеризуют погрешность выполнения функции f_i . Из элементов множеств A_i , $i \in \{1, \dots, N_{v-1}\}$ формируется множество $K^{(v)}$ всевозможных их комбинаций. Ему соответствует аналогично формируемое из элементов множеств Σ_i множество $K^{(v)}$.

Система отношений, выражающая зависимость между внешними характеристиками и возможными вариантами распределения, является математической моделью системы в условиях деградации. Любой параметр $\psi_i = ({}^{(i)}U_{s_v}^{(воз)})$, заданный этой моделью, т. е. $\sigma^{(v)}$ или $T^{(v)}$, полностью определяется соответствующим элементом ${}^{(i)}U_{s_v}^{(воз)}$ множества $U_{s_v}^{(воз)}$.

Если наложить ограничения на время выполнение функций, то элементы множества $U_{s_v}^{(воз)}$, удовлетворяющие этому ограничению, образуют множество допустимых вариантов распределения функций $U_{s_v}^{(дон)} = \{ {}^{(i)}U_{s_v}^{(дон)} \}$. При формировании множества искомых вариантов распределения функций $U_{s_v}^{(дон)}$ из множества $U_{s_v}^{(дон)}$ для каждого состояния $s_v \in S$ выбираются элементы ${}^{(i)}U_{s_v}^{(дон)}$, которым соответствуют минимальные значения $\sigma^{(v)}$, т. е.

$$U_{s_v} = \underset{s_v \in S}{\text{Argmin}} \quad \sigma^{(v)} \{ {}^{(i)}U_{s_v}^{(дон)} \}, i \in \{1, 2, \dots, L_a\}. \quad (2)$$

Выражение (2) представляет собой формальную постановку следующей задачи: найти множество вариантов распределения функций U_{s_v} , элементы которого представляют собой варианты перераспределения функций, обеспечивающие минимальные значения погрешности выполнения функций $\sigma^{(v)}$ для любого состояния $s_v \in S$ при соблюдении ограничений на времена выполнения функций. Общая схема ее решения включает три этапа.

На **первом этапе**, исходя из потребной производительности и погрешности выполнения функций, определяется необходимое количество модулей в первоначальной конфигурации и формируется множество состояний системы $S = \{s_v\}$, в которых необходимо осуществлять перераспределение функций.



Второй этап заключается в формировании множества $U_{s_v}^{(дон)} = \{ {}^{(i)}U_{s_v}^{(дон)} \}$ допустимых вариантов распределения функций путем реализации следующих процедур:

- формирование множества $K_a^{(v)}$ для состояний $s_v \in S$;
- составление планов распределения элементов множества $K_a^{(v)}$ по работоспособным в состоянии $s_v \in S$ модулям;
- оценка времени $T_i^{(v)}$ выполнения i -го плана, $i \in \{1, 2, \dots, |K_a^{(v)}|\}$;
- проверка выполнения ограничения $T_i^{(v)} \leq T^{(дон)}$.

Варианты распределения функций, удовлетворяющие ограничению $T_i^{(v)} \leq T^{(дон)}$, составляют множество $U_{s_v}^{(дон)}$.

На **третьем этапе** решения задачи происходит выделение из множества $U_{s_v}^{(дон)}$ подмножества U_{s_v} искомых вариантов перераспределения функций. При этом реализуются следующие процедуры:

- оценка погрешности $\sigma_i^{(v)}$ выполнения функций множества $F_v^{(i)}$, варианты распределения которых — элементы множества $U_{s_v}^{(дон)}$;
- выбор из множества $U_{s_v}^{(дон)}$ таких вариантов перераспределения функций, для которых

$$\sigma_i^{(v)} = \min_{s_v \in U_{s_v}^{(дон)}} \{ \sigma_1^{(v)}, \sigma_2^{(v)}, \dots, \sigma_{L_a}^{(v)} \}.$$

Выбранные таким образом варианты, составляющие множество U_{s_v} , представляют искомую β_q -стратегию.

Формирование β_l -стратегий

Формирование β_l -стратегий основано на реализации решения π_l применительно к U - и US -системам. Для анализа ситуаций, в которых необходимо формировать β_l -стратегии, будем использовать матрицу состояния системы $B(s_v) = [b_{ij}^{(v)}]$, где $b_{ij}^{(v)} \in \{0, 1\}$, причем $b_{ij}^{(v)} = 1$, если i -й модуль способен выполнять j -ю функцию, $b_{ij}^{(v)} = 0$ в противном случае. Распределение функций между модулями в произвольном состоянии $s_v \in S$ будем задавать матрицей распределения Y_k^v , представляющей собой матрицу подстановки, в которой $y_{ij}^{(k)} = 1$, если $\sigma_k(i) = j$, и $y_{ij}^{(k)} = 0$ в противном случае (где $\sigma_k(i)$ — i -й элемент k -й подстановки на множестве $\{1, 2, \dots, n\}$, $k \in \{1, 2, \dots, n\}$).

Реализация решения π_l предполагает ослабление требований ко времени выполнения функций при неизменных требованиях к погрешности их выполнения. При этом следует выделить два случая [4]:

1. Возникновение активного частичного функционального отказа.
2. Возникновение полного функционального отказа.

В **первом случае** производится анализ матрицы состояния системы $B(s_v)$ и матрицы распределения $Y_k^{(v-1)}$, в соответствии с которой осуществлялось закрепление функций за модулями системы в состоянии s_{v-1} .

В матрице состояния системы $B(s_v)$ определяется элемент, который вследствие осуществления деструктивного воздействия стал нулевым. Допустим, что таким элементом является элемент $b_{kl}^{(v)}$. В матрице $Y_k^{(v-1)}$ ему соответствует элемент $y_{kl}^{(v-1)}$. Далее в l -м столбце матрицы $B(s_v)$ отыскивается элемент, значение которого равно единице. При этом если таких элементов несколько, то выбирается любой из них. Пусть таким элементом будет элемент $b_{ql}^{(v)}$. Затем в матрице $Y_k^{(v-1)}$ элементу $y_{kl}^{(v-1)}$ присваивается нулевое значение, а элементу $y_{ql}^{(v-1)}$ — единичное. Таким образом, матрица $Y_k^{(v-1)}$ преобразуется в матрицу $Y_k^{(v)}$, в соответствии с которой происходит распределение функций по модулям системы. При этом один из модулей (в данном случае это q -й модуль) выполняет две функции и общее время выполнения функций увеличивается на время выполнения функции, передаваемой q -му модулю.



В дальнейшем при возникновении частичного функционального отказа и переходе системы в состояние s_{v+1} матрица $Y_k^{(v+1)}$ формируется таким образом, чтобы элемент, которому присваивается единичное значение, находился в строке с наименьшим количеством единиц.

Совокупность матриц $Y_k^{(v)}$, $v \in \{0, \dots, |S|\}$, представляет искомую β_t -стратегию для случая возникновения частичного функционального отказа.

Примером, иллюстрирующим процедуру формирования матриц $Y_k^{(v)}$, составляющих β_t -стратегию для данного случая, служит следующий. Предположим, что система, состоящая из трех модулей и выполняющая три функции, находилась в некотором состоянии s_v и произошел частичный функциональный отказ, в результате которого она перешла в состояние s_{v+1} . Этой ситуации соответствуют следующие матрицы состояния:

$$B(s_v) = \begin{bmatrix} 0 & 1 & 0 \\ 1 & 1 & 1 \\ 1 & 1 & 0 \end{bmatrix}, \quad B(s_{v+1}) = \begin{bmatrix} 0 & 1 & 0 \\ 1 & 1 & 1 \\ 0 & 1 & 0 \end{bmatrix}.$$

При этом допустимым распределением для состояния s_v было распределение, задаваемое следующей матрицей распределения:

$$Y_k^{(v)} = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 1 & 0 & 0 \end{bmatrix}.$$

В результате частичного функционального отказа элемент $b_{31}^{(v)}$ матрицы $B(s_v)$ принимает в матрице $B(s_{v+1})$ нулевое значение. Ему соответствует элемент $y_{31}^{(v)} = 1$ матрицы $Y_k^{(v)}$. Этому элементу присваивается нулевое значение и производится анализ первого столбца матрицы $B(s_{v+1})$ с целью поиска в нем ненулевых элементов. Таким ненулевым элементом в матрице $B(s_{v+1})$ оказывается элемент $b_{21}^{(v+1)}$, которому в матрице $Y_k^{(v)}$ соответствует элемент $y_{21}^{(v)}$. Элементу $y_{21}^{(v)}$ присваивается единичное значение, а элементу $y_{31}^{(v)}$ — нулевое и матрица $Y_k^{(v)}$ преобразуется в матрицу $Y_k^{(v+1)}$ следующего вида:

$$Y_k^{(v)} = \begin{bmatrix} 0 & 1 & 0 \\ 1 & 0 & 1 \\ 0 & 0 & 0 \end{bmatrix},$$

в соответствии с которой первый модуль системы выполняет вторую функцию, а второй модуль — последовательно первую и третью.

В случае полной потери модулем способности выполнения возложенных на него функций при возникновении функционального отказа формирование β_t -стратегий осуществляется следующим образом.

Предположим, что в систему входят только модули (речь здесь идет о процессорных модулях), имеющие одинаковое быстродействие, и в процессе функционирования произошел функциональный отказ одного из них. Вследствие этого отказа система перешла из состояния s_v в состояние s_{v+1} . Тогда если все модули системы в состоянии s_v выполняли по одной функции, то функция, реализуемая модулем, в котором возник полный функциональный отказ, передается в состоянии s_{v+1} на любой из оставшихся модулей. Если же система содержит модули, выполнявшие не одну функцию, то функция, реализуемая модулем, полностью потерявшим способность выполнения функций, закрепляется в состоянии s_{v+1} за модулем, выполнявшим в состоянии s_v наименьшее количество функций.

Когда система содержит модули, имеющие различное быстродействие, формирование β_t -стратегий осуществляется следующим образом.



Пусть известны значения сложности каждого алгоритма выполнения функции, представляющие, как и ранее, количества команд в программах, необходимых для реализации этих алгоритмов, а также быстродействие каждого модуля.

Время $T_i^{(j)}$ выполнения i -й функции j -м модулем определяется количеством команд N_{f_i} в программе, реализующей алгоритм ее выполнения, и быстродействием B_j j -го модуля следующим образом:

$$T_i^{(j)} = \frac{N_{f_i}}{B_j}. \quad (3)$$

Если j -й модуль выполняет несколько функций, то выражение (3) имеет следующий вид:

$$T_{F_j}^{(j)} = \frac{\sum_{f_i \in B_j} N_{f_i}}{B_j},$$

где запись $f_i \in B_j$ означает, что суммирование происходит для функций, выполняемых j -м модулем.

Предположим, что произошел полный функциональный отказ q -го модуля, выполняющего функцию f_q , и система перешла в некоторое состояние s_v . Для определения модуля, которому необходимо передать данную функцию в состоянии s_v , происходит анализ значений следующего выражения:

$$T_{F_i}^{(i)} = T_{F_i}^{(i)} + T_q^{(i)}, \quad i \in \{1, \dots, N\}, \quad i \neq q.$$

По результатам такого анализа выбирается тот модуль, которому соответствует наименьшее значение $T_{F_i}^{(i)}$, и за ним закрепляется функция f_i .

Время выполнения функций при таком подходе к составлению плана их распределения между модулями системы, строго говоря, не является минимальным, хотя и близко к нему. С целью получения его минимального значения используется схема динамического программирования, как в случае формирования β_n -стратегий.

Формирование смешанных β -стратегий основано на использовании различных комбинаций подходов к формированию соответствующих чистых β -стратегий.

СПИСОК ЛИТЕРАТУРЫ:

1. Авиженис А., Лапри Ж.-К. Гарантоспособные вычисления: от идей до реализации в проектах // ТИИЭР. Пер. с англ. 1989. Т. 74. № 5. С. 8–21.
2. Тарасов А. А. Методы функциональной реконфигурации отказоустойчивых систем // Надежность. 2002. № 2. С. 29–35.
3. Тарасов А. А. Управление функциональной реконфигурацией отказоустойчивых систем // Информационные технологии. 2000. № 8. С. 11–15.
4. Тарасов А. А. Функциональная отказоустойчивость систем обработки информации: монография. М.: МИНИТ, 2009. — 184 с.