

ОЦЕНКА ЭФФЕКТИВНОСТИ ОДНОГО ИЗ ВАРИАНТОВ МНОГОВЕРСИОННОГО РЕЗЕРВИРОВАНИЯ ВЗАИМОСВЯЗАННЫХ ПАРАЛЛЕЛЬНЫХ ЗАДАЧ ДЛЯ УПРАВЛЯЮЩИХ ПАРАЛЛЕЛЬНЫХ ВЫЧИСЛИТЕЛЬНЫХ СИСТЕМ¹

В.В. Игнатущенко

Институт проблем управления им. В.А. Трапезникова РАН
Россия, 117997, Москва, ул. Профсоюзная, 65

М.Л. Милков, А.В. Сидоров

Институт проблем управления им. В.А. Трапезникова РАН
Россия, 117997, Москва, ул. Профсоюзная, 65
E-mail: max-ml@yandex.ru

На основе формализованных описаний и математических моделей вычислительных процессов со случайными временами их реализации, аналитически определяется оценка отказоустойчивости управляющей параллельной вычислительной системы как вероятность успешного завершения произвольного набора взаимозависимых параллельных задач за заданное директивное время при *многоверсионном резервировании* программных модулей с их асинхронным и синхронным запуском, – как при нормальном, штатном функционировании системы, так и при случайной одиночной неисправности одного из ее процессоров. Для оценки отказоустойчивости здесь не задаются и не используются традиционные параметры надежности технических средств (интенсивности или вероятности сбоев или отказов, средняя наработка на отказ и т.п.). Проводится оценка эффективности подхода.

EFFICIENCY RATING FOR ONE OF MULTIVERSION REDUNDANTIZATION VARIANTS OF THE INTERDEPENDENT PARALLEL TASKS FOR THE PARALLEL CONTROL COMPUTING SYSTEMS WITH ASYNCHRONOUS AND SYNCHRONOUS TASKS (WORKS) LAUNCHINGS

/ V.V. Ignatushchenko, M.L. Milkov, A.V. Sidorov (Trapeznikov Institute of Control Sciences, Russian Academy of Sciences, 65, Profsoyuznaya St., Moscow, 117997, Russia). With the use of formalized descriptions and mathematical models of the computing processes having random times of realization, fault-tolerance of the parallel computing control system is estimated analytically as the probability of successful completion of an arbitrary set of interdependent parallel tasks in a user-defined deadline under multiversion redundantization of program modules with their asynchronous and synchronous launchings both at standard operation of the system and a single random fault of one processor. The traditional parameters of hardware reliability such as rates or probabilities of faults and failures, mean time before failure, and so on are not defined or used. Efficiency of the approach is estimated.

¹ Работа частично поддержана грантом Российского фонда фундаментальных исследований (проект № 09-08-00372-а)

1. Введение. Постановка задач исследования

Рассматриваемое исследование является составной частью разработки новой компьютерной технологии [1–4], создаваемой в Институте проблем управления РАН. Эта технология направлена на обеспечение *надежного* выполнения сложных наборов задач, со *случайными* временами их реализации, в *параллельных* управляющих вычислительных системах (ВС), функционирующих в контурах управления реального времени. Под надежным выполнением [1] конкретного набора задач в параллельной ВС понимается выполнение его за время, не превышающее заданное (пользователем) директивное время T_{\max} , с *требуемой* (удовлетворяющей пользователя) *вероятностью*. **Отказ** управляющей ВС в принятом здесь понятии *надежного* выполнения задач трактуется не только как нарушение или потеря ее физической работоспособности, но и как **НЕ**выполнение требуемых задач за время T_{\max} [1–4]. Такой отказ может иметь место при нормальном, безотказном функционировании аппаратуры и программного обеспечения ВС.

Применительно к управляющим *параллельным* ВС проблема исследования возможностей успешного завершения заданного пользователем набора задач на ВС за априорно предписанное директивное время T_{\max} получило название *математического прогнозирования* времени выполнения наборов задач [1–4].

Наборы задач (в общем случае – *программных модулей*) рассматриваются в наиболее трудоемких версиях – с произвольными информационными и логическими связями между задачами и/или их фрагментами – и потому названы в [1–4] комплексами взаимосвязанных работ (КВР) – комплексами взаимосвязанных программных модулей со *случайными* временами их выполнения.

В качестве управляющих параллельных ВС в технологии [1–4] рассматриваются многопроцессорные ВС со многими потоками команд и данных (класс МКМД, включая мультимикропроцессорные ВС с распределенной структурой), которые предназначены для использования в сложных контурах управления реального времени.

Формально под математическим (статическим) прогнозированием времени выполнения конкретного, заданного пользователем КВР по [1–4] понимается определение в статике (т.е. до реализации задач в ВС) стохастических оценок времени T реализации КВР и наиболее важной из них – функции $F(t)$ распределения времени T . По функции $F(t)$ определяется вероятность P завершения КВР (или его фрагментов) за время, не превышающее заданное T_{\max} , на параллельной ВС с заданной или предполагаемой конфигурацией и производительностью ее вычислительных ресурсов – процессоров ВС.

В настоящее время технология [1 – 4], базирующаяся на аппарате обрывающихся марковских процессов (ОМП), матстатистики и теории массового обслуживания (ТМО), развивается в направлении исследований различных методов и средств *резервирования программных модулей*, соответствующих работам КВР, и в первую очередь – нетрадиционных методов их асинхронного [5 – 7] и синхронного [8 – 10] резервирования (АР и СР соответственно) на основе математического прогнозирования надежного выполнения произвольного КВР (заданного либо сформированного по формальным правилам [11, 12]).

Исследуемое в данной работе *многоверсионное резервирование* (МР) вычислительных процессов (программных модулей) использует возможности асинхронного и синхронного резервирования. Принципы МР проистекают из принципов многоверсионного (многовариантного) программирования [13, 14], разработанных еще для традиционных, последовательных ВС в качестве методов анализа, верификации и обеспечения отказоустойчивого выполнения сложных программных комплексов.

Добавим, что само понятие «многоверсионное резервирование взаимосвязанных программных модулей» было предложено в ИПУ РАН, а затем и применено в литературе сравнительно недавно [15], и потому особенности этого варианта асинхронного резервирования, его свойства, потенциальные возможности и эффекты (как положительные, так и отрицательные) до сих пор практически не исследованы.

В отличие от СР и АР, при *многоверсионном* резервировании вместо идентичных копий программных модулей используются различные версии работ КВР, соответствующие различным алгоритмам реализации одной и той же работы. В настоящее время наиболее перспективными вариантами многоверсионного резервирования взаимозависимых параллельно-последовательных программных модулей представляются следующие [16]:

1. Резервные работы реализуют те же функции, что и соответствующие им работы-оригиналы (в терминологии [6]) исходного КВР, с теми же логическими и информационными связями между работами; таким образом, структура резервного КВР идентична структуре исходного КВР (как, например, при АР по [7]), но при этом резервные программные модули реализуют другие, более простые алгоритмы вычислений. Например, каждая резервная работа (названная работой-версией) является вычислительным аналогом работы-оригинала исходного КВР, но с меньшей точностью расчетов и, соответственно, с заведомо меньшим временем выполнения.

2. В резервном КВР полностью сохраняется структура связей исходного КВР, но исходный и резервный программные модули, выполняющие одну и ту же функцию, реализуют существенно различные вычислительные алгоритмы. К этому же варианту МР можно отнести случай, когда работа-оригинал и работа-версия написаны на различных языках программирования и/или разными группами пользователей. В этих случаях длительности выполнения резервных работ могут оказаться как большими, так и меньшими по отношению к временам реализации работ-оригиналов. Очевидно, такой вариант МР соответствует интерпретации асинхронного резервирования (АР), отличной от его трактовки по [5 – 7].

3. Резервный КВР имеет параметры, существенно отличающиеся от параметров исходного КВР – и по числу работ, и по длительностям их выполнения, и по топологии связей между работами; обычно это имеет место при разработке различных версий крупного, сложного программного комплекса (например, разработка одной и той же проблемы поручена двум конкурирующим отделам, организациям и т.п.).

В настоящей работе авторы ограничиваются исследованием только варианта 1 (из упомянутых выше), который близок к традиционной трактовке асинхронного резервирования по [5 – 7]; в этом варианте основное отличие МР от АР заключается в том, что при МР результаты работы-оригинала и соответствующей работы-версии могут не совпадать – с некоторым допустимым разбросом их численных значений, – при котором эти результаты полагаются все еще одинаковыми.

Цель и смысл подхода к оценке отказоустойчивости управляющих ВС, использующих программную избыточность (в терминологии [13, 14]) заключается в том, что на основе формализованных описаний и математических моделей вычислительных процессов аналитически определяются вероятности успешного завершения каждого конкретного КВР за заданное директивное время T_{max} при *многоверсионном* резервировании программных модулей, – как при нормальном, штатном функционировании ВС, так и при случайной *одиночной* неисправности – сбое или отказе одного (любого) – из процессоров ВС при реализации им любого из программных модулей (работ) КВР.

Для достижения этой цели, применительно к рассматриваемому варианту многоверсионного резервирования (в понятиях и терминологии [12,13]) в [16] была:

- формально обоснована, синтезирована и логически описана новая структура программного блока диагностирования (БД) для определения «координаты» одиночной неисправности (номера неисправного процессора ВС), ее типа (сбой или отказ) и идентификации работы КВР (программного модуля) с искаженными результатами ее выполнения;

- на основе аппарата ТМО и марковских процессов, разработана математическая модель для прогнозирования времени выполнения КВР с многоверсионным резервированием программных модулей как для штатного режима функционирования ВС, так и в условиях одиночных сбоев или отказов ее вычислительных ресурсов.

Целью настоящей работы является оценка эффективности разработанных в [16] формализованных описаний и математических моделей для выше указанного варианта 1 МР с асинхронным и синхронным запусками работ (программных модулей).

2. Эксперимент и численные результаты (асинхронный запуск работ)

Расчет разработанной в [15] математической модели для всех рассмотренных версий КВР (исходного, базового и *преобразованных* КВР) при многоверсионном резервировании работ КВР (так же, как и для АР, и для СР) может быть выполнен по методологии из [1 – 4]: формируется граф переходов состояний ОМП и соответствующая ему матрица интенсивностей переходов, а по методу [17] восстанавливаются соответствующие функции распределения для определения вероятности успешного завершения КВР за заданное директивное время T_{max} , – с одной стороны, а с другой – для определения вероятности отказа ВС в оговоренном ранее смысле (см. раздел 1). Заметим, однако, что получаемые по методу [17] аналитические оценки вероятности успешного завершения сформированных *преобразованных* КВР за время T_{max} являются нижними, наиболее «пессимистическими», т.к. эти КВР могут включать максимально возможное (и зачастую недостижимое) количество Q_j работ «отката» [6,12], которые необходимо перезапускать для восстановления вычислительного процесса.

Более точные, «эластичные» оценки могут быть получены с помощью программного комплекса аналитико-имитационного моделирования процессов выполнения любого из *преобразованных* КВР, включая КВР с МР, на основе математической модели, описанной в [16], т.к. при таком моделировании фиксируется *состояние* ОМП в момент обнаружения типа неисправности ОП, а в конечном счете – номера тех работ «отката», которые действительно необходимо перезапускать на повторное выполнение. Комплекс разработан и постоянно развивается в ИПУ РАН.

Для оценки эффективности варианта 1 многоверсионного резервирования программных модулей в данной работе рассматриваются три исходных КВР: КВР₁, КВР₂, КВР₃ (рис. 1,а,б,в). Один из них, – КВР₂ (рис. 1,б) совпадает с примером КВР по [16]. Два других являются его модификациями: с меньшим – КВР₁ (рис. 1,а) и большим – КВР₃ (рис.1,в) коэффициентами параллелизма² [18]. При этом количество работ и их временные параметры остаются неизменными. Изменяются лишь информационно-логические связи между работами КВР.

На рис. 2,а,б,в представлены графики функций $F(t)$ распределения времен выполнения трех базовых и шести *преобразованных* КВР с *асинхронным запуском работ*, полученные с помощью упомянутого комплекса.

Графики пронумерованы следующим образом:

- 1 – штатное выполнение базового КВР₁;
- 2 – реализация *преобразованного* КВР₁, сформированного для события *сбоя* при выполнении одной из тройки контрольных работ (в нашем примере - работ с номером 5): работы-оригинала a_5 (рис. 2,а), работы-версии a_5' (рис. 2,б), работы сравнения b_5 (рис. 2,в);
- 3 – реализация *преобразованного* КВР₁, сформированного для события *отказа* при выполнении одной из тройки тех же контрольных работ соответственно;
- 4 – штатное выполнение базового КВР₂;
- 5 – реализация *преобразованного* КВР₂, сформированного для события *сбоя* при выполнении одной из тройки контрольных работ: работы-оригинала a_5 (рис. 2,а), работы-версии a_5' (рис. 2,б), работы сравнения b_5 (рис. 2,в);

² На содержательном уровне коэффициент параллелизма равен наибольшему числу работ КВР, которые могут выполняться одновременно (параллельно).

- 6 – реализация *преобразованного* КВР₂, сформированного для события *отказа* при выполнении одной из тройки контрольных работ;
- 7 – штатное выполнение базового КВР₃;
- 8 – реализация *преобразованного* КВР₃, сформированного для события *сбоя* при выполнении одной из тройки контрольных работ: работы-оригинала a_5 (рис. 2,а), работы-версии a_5' (рис. 2,б), работы сравнения b_5 (рис. 2,в);
- 9 – реализация *преобразованного* КВР₃, сформированного для события *отказа* при выполнении одной из тройки тех же контрольных работ соответственно;

В табл. 1.1, 1.2, 1.3 приведены значения вероятностей p времен успешных реализаций упомянутых выше КВР при заданном директивном времени $T_{max}=800$ т. Таблицы 1.1, 1.2, 1.3 соответствуют рисункам 2,а, 2,б и 2,в для рассматриваемых КВР₁, КВР₂, КВР₃ соответственно.

Проведенный программный эксперимент выявил следующий эффект: при использовании разработанных в [16] БД и правил функционирования математической модели, специализированных для рассматриваемого варианта МР, вероятность успешного выполнения *преобразованного* КВР со *сбоем* или *отказом* ОП может оказаться **большей**, чем вероятность *штатного* выполнения базового КВР, т.е. при отсутствии ошибок ОП (см. табл. 1 и рис. 2).

Данное различие в вероятностях успешного выполнения *преобразованного* КВР и *штатного* выполнения базового КВР особенно значительно при возникновении неисправности (*сбой* или *отказ*) во время выполнения работы-оригинала a_5 или работы сравнения b_5 , т.е. при неисправности процессора Π_f по [16]: до 336% для КВР₁, до 48% для КВР₂ и до 27% для КВР₃. И наоборот менее значительно при возникновении неисправности (*сбой* или *отказ*) во время выполнения работы-версии a_5' , т.е. при неисправности процессора Π_h по [16]: до 17% для КВР₁, до 2% для КВР₂ и до 1% для КВР₃.

В рассматриваемом эксперименте, вероятность успешного выполнения *преобразованного* КВР оказалась меньше, чем при *штатном* выполнении базового КВР, только при возникновении *отказа* во время выполнения работы-версии a_5' : для КВР₂ -4%, и для КВР₃ -1%, (см. табл. 1.2 и рис. 2,б).

Данный эффект является следствием специфических особенностей рассматриваемого варианта МР. После обнаружения неисправности процессора Π_f вычисления продолжают только для работ ветки базового КВР состоящей из быстрых, но менее точных (по алгоритмам реализации) работ-версий. При обнаружении неисправности процессора Π_h вычисления продолжают только для работ ветки базового КВР состоящей из медленных, но более точных (по алгоритмам реализации) работ-оригиналов.

Как можно заметить, степень проявления данного эффекта также зависит от коэффициента параллелизма [17] рассматриваемых КВР. Для КВР с малым коэффициентом параллелизма данный эффект наиболее выражен, а для КВР с большим коэффициентом параллелизма наименее выражен.

Кроме того, данный эксперимент выявил эффект практически полного совпадения графиков функций $F(t)$ распределения времен выполнения *преобразованных* КВР, сформированных для событий *сбоя* и *отказа* во время выполнения любой из тройки контрольных работ. Так на рис. 2,а и 2,в графики 2 и 3; 5 и 6; 8 и 9 практически совпадают. Отметим, что на рисунках обозначены только по одному графику из каждой пары возможных событий (*сбой* или *отказ*), в нашем случае, - 2, 5 и 8 (последние соответствуют событию *сбоя*), аналогично совпадают графики 2 и 3 на рис. 2,б. При этом вероятности успешного выполнения *преобразованных* КВР для случая *сбоя* или *отказа* при выполнении одной из тройки контрольных работ отличаются лишь на десятые доли процентов. По табл. 1.1, 1.2, 1.3 разница составляет от 0,01% до 1,35%. Таким образом, для рассматриваемых в работе структур КВР₁ – КВР₃ вероятности их успешного завершения для *преобразованных* КВР практически совпадают (табл. 1.1, 1.2, 1.3), что является достоинством разработанного в [16] БД, который обеспечивает описанный выше эффект.

3. Эксперимент и численные результаты (синхронный запуск работ)

Для оценки эффективности варианта 1 многоверсионного резервирования программных модулей при синхронном запуске, аналогично п.2 рассматриваются три исходных КВР: КВР₁, КВР₂, КВР₃ (рис. 1,а,б,в соответственно).

На рис. 3,а,б,в представлены графики функций $F(t)$ распределения времен выполнения трех базовых и шести преобразованных КВ с синхронным запуском работ, полученные с помощью упомянутого комплекса.

Графики пронумерованы следующим образом:

- 1 – штатное выполнение базового КВР₁;
- 2 – реализация преобразованного КВР₁, сформированного для события *сбоя* при выполнении одной из тройки контрольных работ (в нашем примере - работ с номером 5): работы-оригинала a_5 (рис. 3,а), работы-версии a_5' (рис. 3,б), работы сравнения b_5 (рис. 3,в);
- 3 – реализация преобразованного КВР₁, сформированного для события *отказа* при выполнении одной из тройки тех же контрольных работ соответственно;
- 4 – штатное выполнение базового КВР₂;
- 5 – реализация преобразованного КВР₂, сформированного для события *сбоя* при выполнении одной из тройки контрольных работ: работы-оригинала a_5 (рис. 3,а), работы-версии a_5' (рис. 3,б), работы сравнения b_5 (рис. 3,в);
- 6 – реализация преобразованного КВР₂, сформированного для события *отказа* при выполнении одной из тройки контрольных работ;
- 7 – штатное выполнение базового КВР₃;
- 8 – реализация преобразованного КВР₃, сформированного для события *сбоя* при выполнении одной из тройки контрольных работ: работы-оригинала a_5 (рис. 3,а), работы-версии a_5' (рис. 3,б), работы сравнения b_5 (рис. 3,в);
- 9 – реализация преобразованного КВР₃, сформированного для события *отказа* при выполнении одной из тройки тех же контрольных работ соответственно;

В табл. 2.1, 2.2, 2.3 приведены значения вероятностей p времен успешных реализаций упомянутых выше КВР при заданном директивном времени $T_{max}=800$ τ. Таблицы 2.1, 2.2, 2.3 соответствуют рисункам 3,а, 3,б и 3,в для рассматриваемых КВР₁, КВР₂, КВР₃ соответственно.

Проведенный программный эксперимент для синхронного запуска выявил эффект, присущий и для асинхронного запуска: при использовании разработанных в [16] БД и правил функционирования математической модели, специализированных для рассматриваемого варианта МР, вероятность успешного выполнения преобразованного КВР со *сбоем* или *отказом* ОП может оказаться (в большинстве случаев проведенных экспериментов, для 8 из 9 базовых КВР) большей, чем вероятность *штатного* выполнения базового КВР, т.е. при отсутствии ошибок ОП (см. табл. 1 и рис. 2).

Данное различие в вероятностях успешного выполнения преобразованного КВР и *штатного* выполнения базового КВР особенно значимо при возникновении неисправности (*сбой* или *отказ*) во время выполнения работы-оригинала a_5 или работы сравнения b_5 , т.е. при неисправности процессора Π_f по [16]: до 203% для КВР₁, до 44.2% для КВР₂ и до 31.4% для КВР₃. И наоборот менее значимо при возникновении неисправности (*сбой* или *отказ*) во время выполнении работы-версии a_5' , т.е. при неисправности процессора Π_h по [16]: до 4.71% для КВР₁, до -1.97% для КВР₂ и до 0.556% для КВР₃.

В рассматриваемом эксперименте для синхронного случая, вероятность успешного выполнения преобразованного КВР оказалась меньше, чем при *штатном* выполнении базового КВР, только для КВР₂ при возникновении *сбоя* или *отказа* во время выполнении работы-версии a_5' : для *сбоя* -1.97%, и для *отказа* -2.58%, (см. табл. 2.2 и рис. 3,б).

Как и для асинхронного случая можно заметить, что степень проявления данного эффекта также зависит от коэффициента параллелизма [17] рассматриваемых КВР. Кроме

того, эксперимент выявил эффект практически полного совпадения графиков функций $F(t)$ распределения времен выполнения *преобразованных* КВР, сформированных для событий *сбоя* и *отказа* во время выполнения любой из тройки контрольных работ. Так на рис. 3,а, б, в графики 2 и 3; 5 и 6; 8 и 9 практически совпадают. При этом вероятности успешного выполнения *преобразованных* КВР для случая *сбоя* или *отказа* при выполнении одной из тройки контрольных работ отличаются лишь на десятые доли процентов. По табл. 2.1, 2.2, 2.3 разница составляет от 0,0115% до 0,409%, что является достоинством разработанного в [16] БД, который обеспечивает описанный выше эффект.

4. Сравнение результатов при различных режимах запуска работ (асинхронном и синхронном)

В табл. 3.1, 3.2, 3.3 приведено сравнение вероятностей времен успешных реализаций упомянутых выше КВР при заданном директивном времени $T_{max}=800$ τ для синхронного и асинхронного запусков для рассматриваемых КВР₁, КВР₂, КВР₃ соответственно и для случаев *сбоя* или *отказа* во время выполнении каждой из контрольных работ.

Синхронный запуск в случае многоверсионного резервирования типа 1 дает преимущество только для КВР с низким показателем параллелизма в режиме штатного выполнения. Во всех остальных случаях асинхронный запуск работ оказывается более эффективным. Это объясняется более рациональным использованием ресурсов при наличии параллелизма, за счет уменьшения времени простаивания процессора при ожидании выполнения более длительной работы версии-оригинала. Эффект уменьшения времени особенно заметен на КВР₁ для событий *сбоя* и *отказа* во время выполнении работы-оригинала a_5 или работы сравнения b_5 . Это достигается за счет более быстрого выполнения «короткой» ветки КВР (с работами меньшими по длительности) и отсутствием необходимости дорабатывать длинную ветку в связи с обнаружением неисправности.

Исследование выявило более явное преимущество синхронного метода над асинхронным на КВР смешенного типа в смысле параллелизма (рис. 1б) для случаев *сбоя* или *отказа*, а также относительно аномальное совпадения результатов экспериментов для случаев *сбоя* и *отказа* во время выполнении работы-оригинала a_5 или работы сравнения b_5 для высокопараллельного КВР₃, что является достаточно неожиданным, так как синхронный запуск в случае с КВР₃ традиционно дает более лучший результат.

5. Заключение

Излагается подход к оценке отказоустойчивости управляющих параллельных ВС при выполнении сложных комплексов взаимосвязанных работ (программных модулей) – КВР, со *случайными* временами реализации и резервированием работ: на основе формализованных описаний и математических моделей вычислительных процессов аналитически определяются *вероятности успешного завершения* каждого конкретного (задаваемого пользователем) КВР за заданное (пользователем) директивное время при *многоверсионном резервировании* (МР) программных модулей КВР с их асинхронным запуском. Эти оценки определяются как для нормального, штатного выполнения КВР с резервированными программными модулями (базового КВР), так и для случаев *одиночной* неисправности (ошибки) – сбоя или отказа – одного (любого) из процессоров ВС при реализации им *любой* из работ базового КВР.

Предложенный в [16] БД является менее сложным (с меньшими затратами вычислительных ресурсов) и менее инерционным (с меньшим временем реализации) по сравнению с унифицированным БД по [11, 12] и со специализированным БД для традиционного синхронного резервирования программных модулей КВР [10].

Отметим, что для оценки отказоустойчивости – определения вероятности отказа управляющей ВС при выполнении КВР с резервированием его работ (или, наоборот, вероятности успешного завершения его за директивное время) – здесь *не задаются и не*

используются традиционные параметры надежности технических средств (интенсивности или вероятности сбоев или отказов, средняя наработка на отказ и т.п.

На примере программного эксперимента показано, что изложенный в работе [16] подход в 89% случаев (для 16 из 18 преобразованных КВР) позволяет *не ухудшить* надежность системы (в терминах функции $F(t)$ распределения времени выполнения КВР) при возникновении одиночной неисправности (сбоя или отказа) процессора ВС.

СПИСОК ЛИТЕРАТУРЫ

1. *Ignatushchenko V.V.* A principle of dynamic control of parallel computing processes on the basis of static forecasting // Proc. of the 10-th Int. Conf. on Parallel and Distributed Computing Systems (PDCS-97). New Orleans, USA, Oct. 1997. P. 593–597.
2. *Игнатущенко В.В., Подшивалова И.Ю.* Динамическое управление параллельными вычислительными процессами на основе статического прогнозирования их выполнения // Автоматика и телемеханика. 1997. № 5. С. 160-173.
3. *Игнатущенко В.В., Подшивалова И.Ю.* Динамическое управление надежным выполнением параллельных вычислительных процессов для систем реального времени // Автоматика и телемеханика. 1999. № 6. С.142-157.
4. *Елисеев В.В., Игнатущенко В.В.* Проблема надежного выполнения сложных наборов задач в управляющих параллельных вычислительных системах // Проблемы управления. 2006. № 6. С. 6-18.
5. *Игнатущенко В.В., Тепляков А.В.* Об эффективности асинхронного резервирования параллельных вычислительных процессов // Автоматика и телемеханика. 1994. № 6. С. 154-169.
6. *Елисеев В.В., Игнатущенко В.В., Подшивалова И.Ю.* Оценка отказоустойчивости управляющих параллельных вычислительных систем: новый подход // Автоматика и телемеханика. 2007. № 6. С. 166-185.
7. *Подшивалова И.Ю.* Асинхронное резервирование комплексов взаимосвязанных программных модулей при их выполнении на отдельных вычислительных ресурсах // Труды XXXV Междунар. Конф. «Информационные технологии в науке, социологии, экономике и бизнесе» (IT+SE'08). Май 2008. Украина. Ялта-Гурзуф. С. 77-79.
8. *Трубкин Д.В., Шастун В.В.* Синхронное резервирование взаимосвязанных программных модулей в параллельных вычислительных системах: новый подход // Труды XXXIV Междунар. Конф. «Информационные технологии в науке, социологии, экономике и бизнесе» (IT+SE'07). Май 2007. Украина. Ялта-Гурзуф. С. 63–66.
9. *Королев С.С.* Новые математические модели синхронного резервирования взаимосвязанных программных модулей в параллельных ВС // Труды XXXV Междунар. Конф. «Информационные технологии в науке, социологии, экономике и бизнесе» (IT+SE'08). Май 2008. Украина. Ялта-Гурзуф. С. 69-72.
10. *Исаева Н.А., Королев С.С.* Синхронное резервирование взаимозависимых параллельных задач для управляющих параллельных вычислительных систем: формализованное описание, оценка отказоустойчивости // Надежность. 2009. №1.
11. *Исаева Н.А.* Логический синтез процедур резервирования взаимосвязанных программных модулей в параллельных вычислительных системах // Труды XXXV Междунар. Конф. «Информационные технологии в науке, социологии, экономике и бизнесе» (IT+SE'08). Май 2008. Украина. Ялта-Гурзуф. С. 64-68.
12. *Игнатущенко В.В., Исаева Н.А.* Резервирование взаимосвязанных программных модулей для управляющих параллельных вычислительных систем: организация, оценка отказоустойчивости, формализованное описание // Автоматика и телемеханика. 2008. №10. С. 142-161.

13. *Авиженис А.* Отказоустойчивость – свойство, обеспечивающее постоянную работоспособность цифровых систем // ТИИЭР. Т.66. 1978. № 10. С. 5-25.
14. *Головкин Б.А.* Параллельные вычислительные системы. М.: Наука. 1980. 519 с.
15. *Милков М.Л., Сидоров А.В.* Математические модели многоверсионного резервирования комплексов взаимосвязанных программных модулей // Труды XXXV Междунар. Конф. «Информационные технологии в науке, социологии, экономике и бизнесе» (IT+SE'08). Май 2008. Украина. Ялта-Гурзуф. С. 80-83.
16. *Игнатущенко В.В., Милков М.Л., Сидоров А.В.* Многоверсионное резервирование взаимозависимых параллельных задач для управляющих параллельных вычислительных систем: формализованное описание, оценка отказоустойчивости // Надежность. 2009. №4. С. 44-61.
17. *Иванов Н.Н., Игнатущенко В.В., Михайлов А.Ю.* Статическое прогнозирование времени выполнения комплексов взаимосвязанных работ в многопроцессорных вычислительных системах // Автоматика и телемеханика. 2005. № 6. С. 89-103.
18. *Игнатущенко В.В.* Организация структур управляющих многопроцессорных вычислительных систем // М.: Энергоатомиздат. 1984. 178 с.

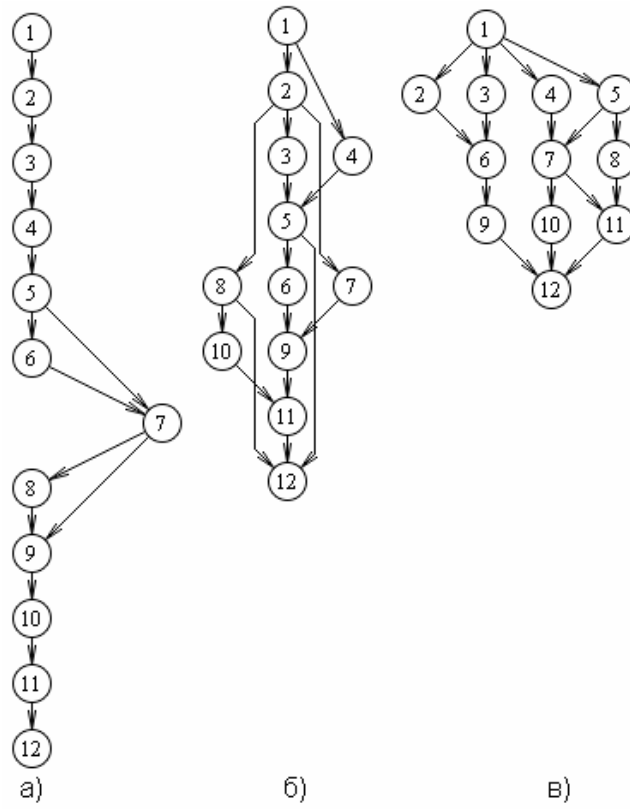
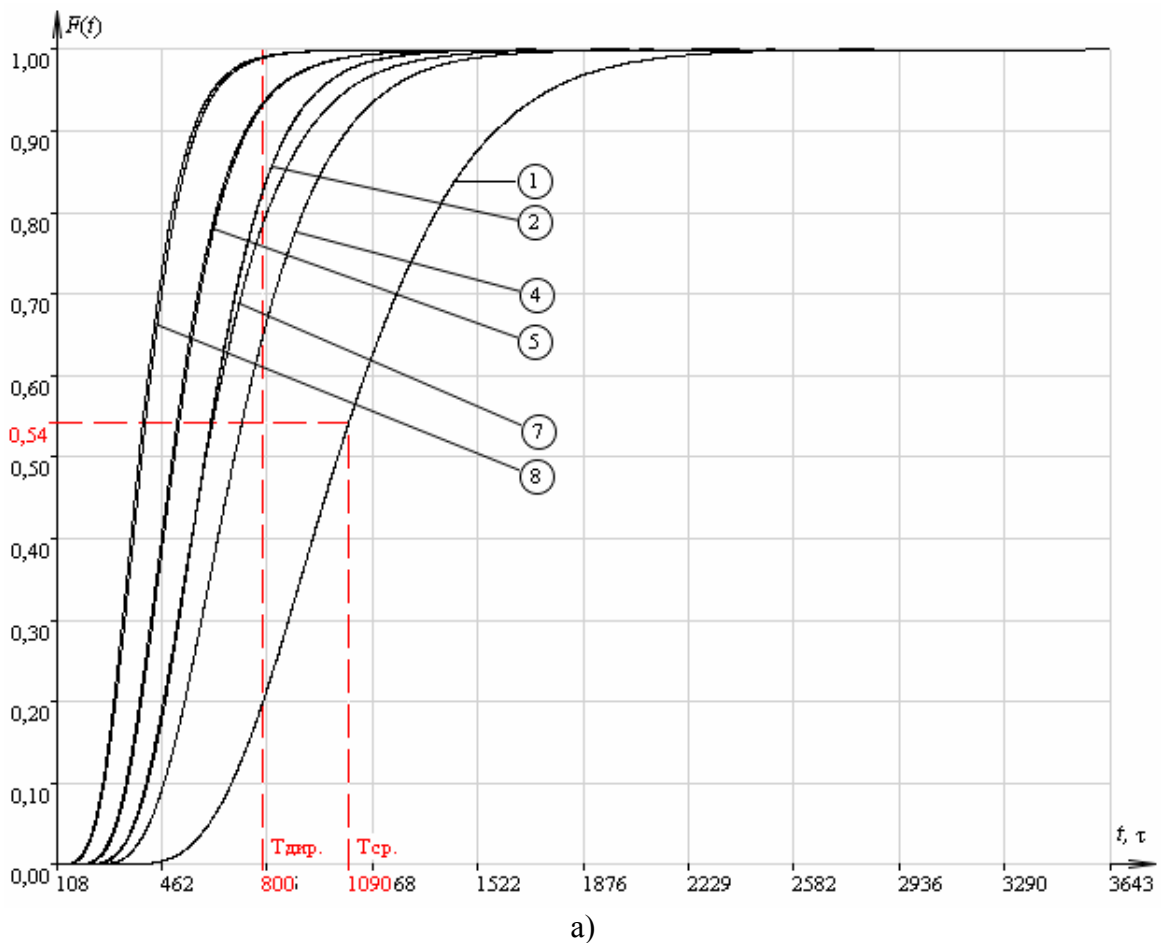
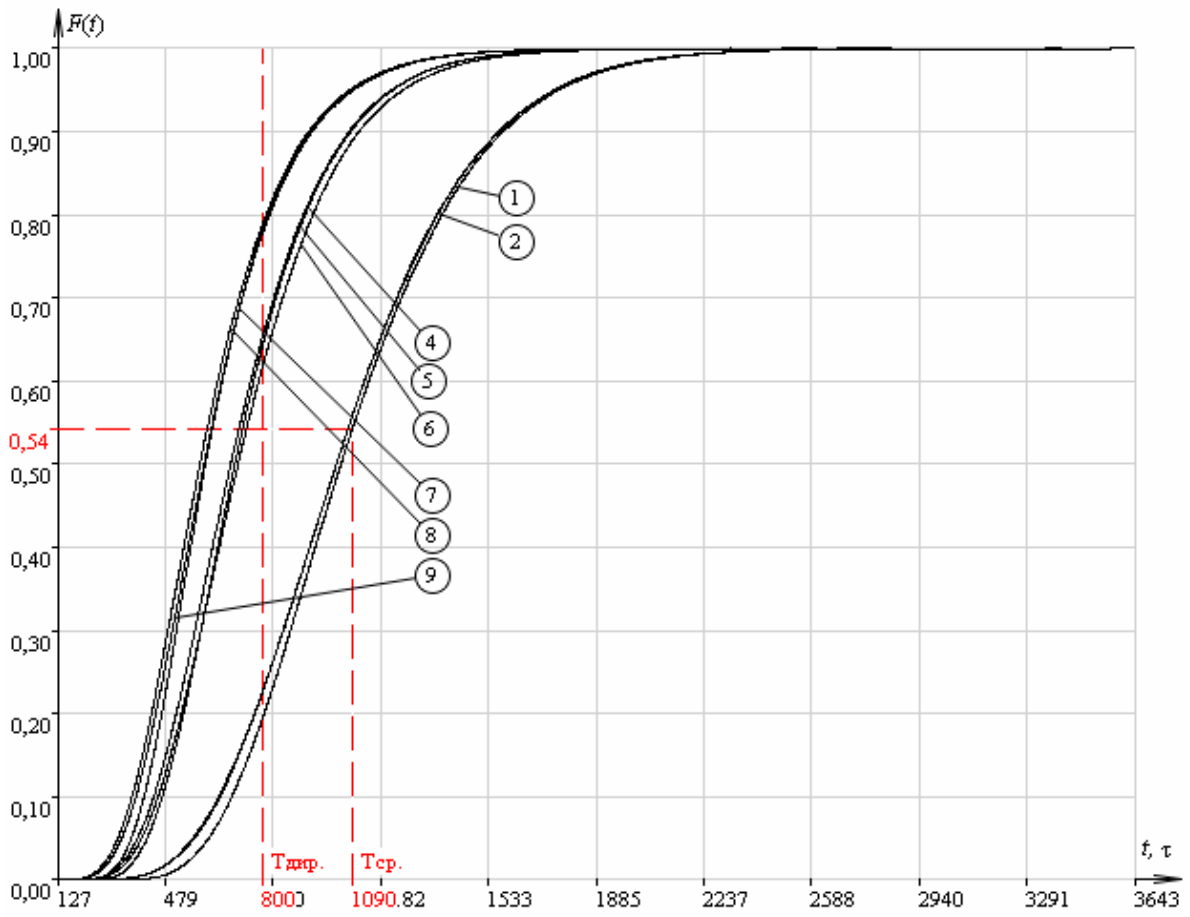
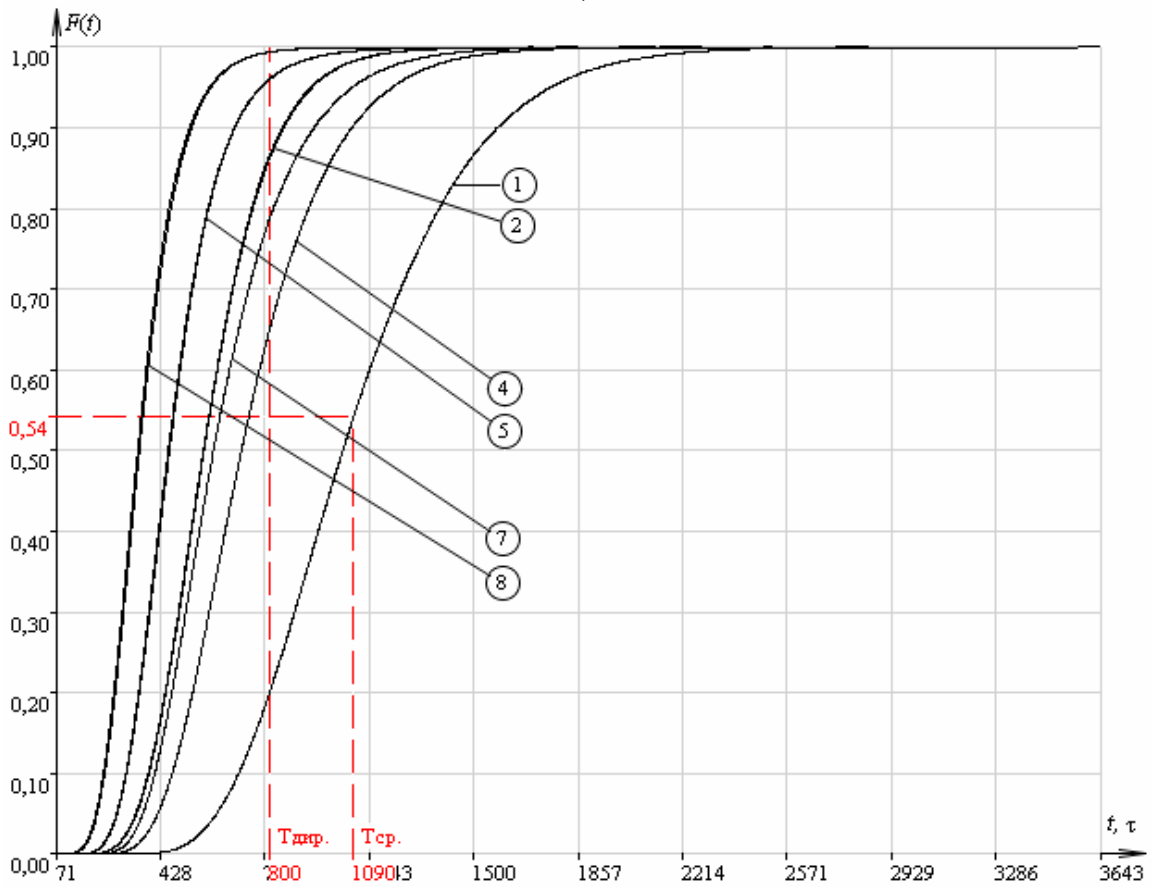


Рис. 1





б)



в)

Рис. 2

Таблица 1.1

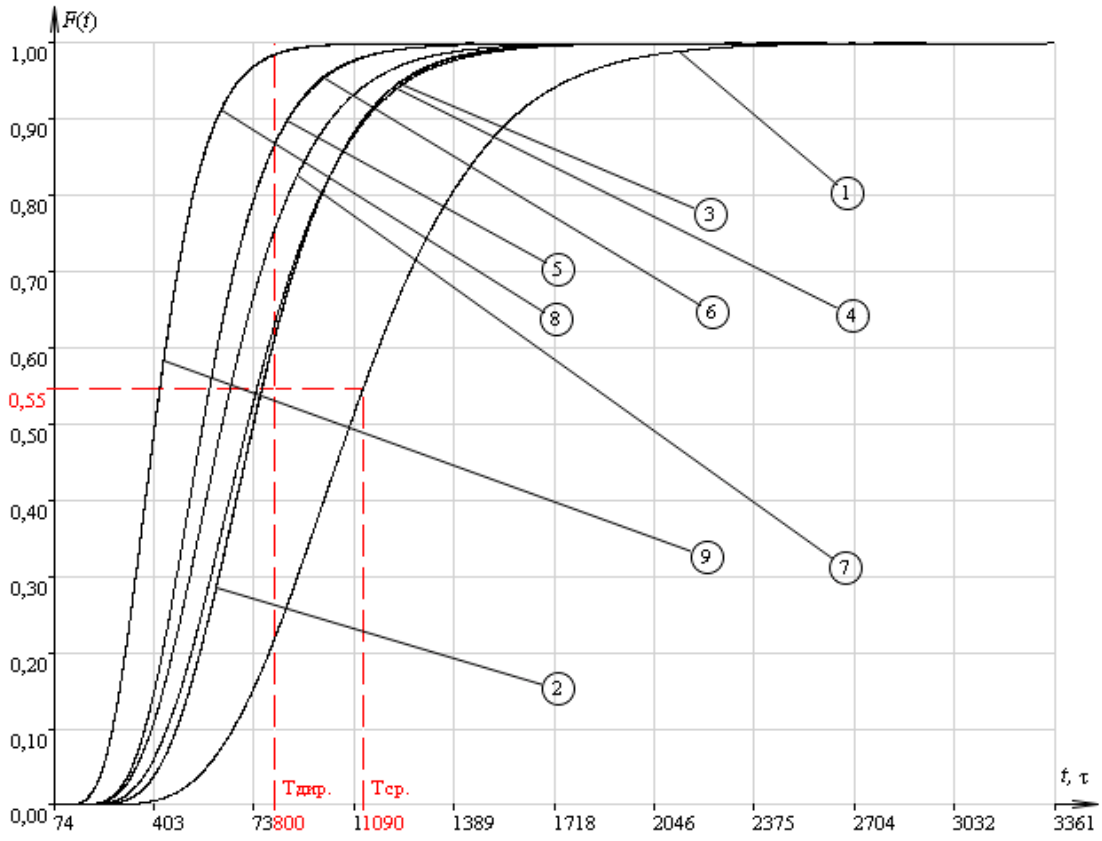
Номер графика	Вероятность p
1	0,1982
2	0,8262
3	0,8250
4	0,6470
5	0,9331
6	0,9317
7	0,7855
8	0,9909
9	0,9897

Таблица 1.2

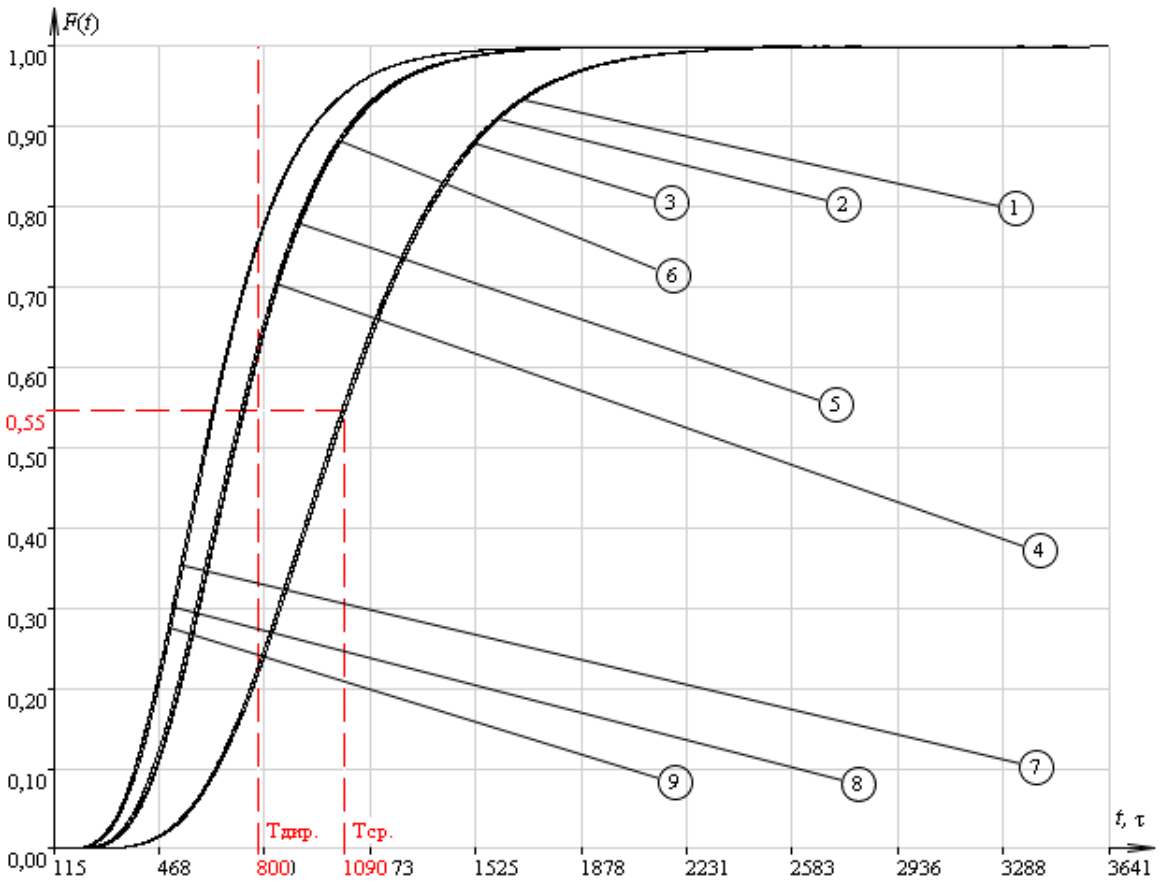
Номер графика	Вероятность p
1	0,1982
2	0,2312
3	0,2299
4	0,6470
5	0,6579
6	0,6219
7	0,7855
8	0,7918
9	0,7811

Таблица 1.3

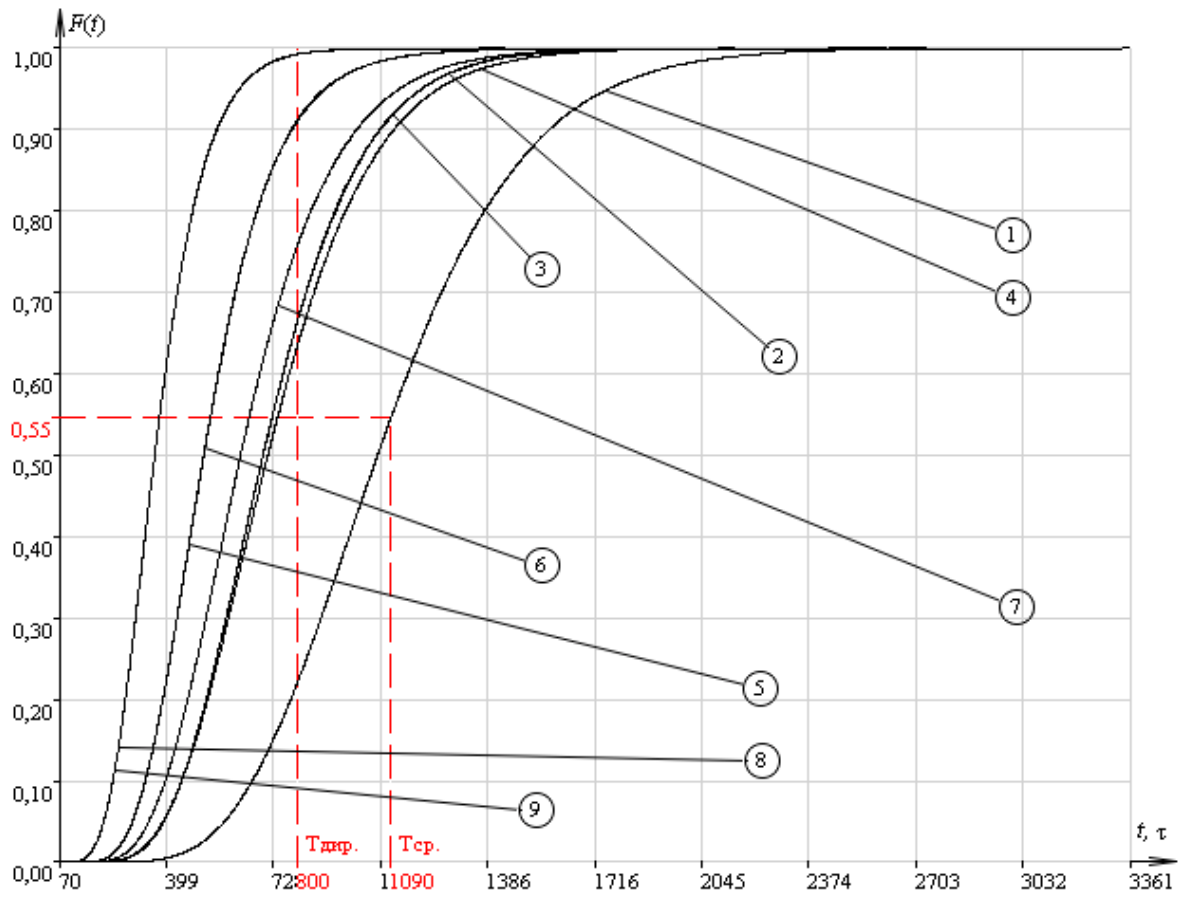
Номер графика	Вероятность p
1	0,1982
2	0,8638
3	0,8602
4	0,6470
5	0,9602
6	0,9598
7	0,7855
8	0,9940
9	0,9939



а)



б)



в)
Рис. 3

Таблица 2.1

Номер графика	Вероятность p
1	0,2188
2	0,6115
3	0,6140
4	0,6315
5	0,8678
6	0,8677
7	0,7556
8	0,9854
9	0,9851

Таблица 2.2

номер графика	Вероятность p
1	0,2188
2	0,2291
3	0,2283
4	0,6315
5	0,6190
6	0,6152
7	0,7556
8	0,7575
9	0,7598

Таблица 2.3

Номер графика	Вероятность p
1	0,2188
2	0,6621
3	0,6610
4	0,6315
5	0,9091
6	0,9107
7	0,7556
8	0,9928
9	0,9926

Таблица 3.1

Минимальное значение вероятности	$p_{ac} - p_c$	$(p_{ac} - p_c)\%$
0,1982	-0,0206	-10,394
0,6115	0,2147	35,1104
0,614	0,211	34,3648
0,6315	0,0155	2,45447
0,8678	0,0653	7,52478
0,8677	0,064	7,37582
0,7556	0,0299	3,95712
0,9854	0,0055	0,55815
0,9851	0,0046	0,46696

Таблица 3.2

Минимальное значение вероятности	$p_{ac} - p_c$	$(p_{ac} - p_c)\%$
0,1982	-0,0206	-10,394
0,2291	0,0021	0,91663
0,2283	0,0016	0,70083
0,6315	0,0155	2,45447
0,619	0,0389	6,28433
0,6152	0,0067	1,08908
0,7556	0,0299	3,95712
0,7575	0,0343	4,52805
0,7598	0,0213	2,80337

Таблица 3.3

Номер графика	Минимальное значение вероятности	$p_{ac} - p_c$	$(p_{ac} - p_c)\%$
1	0,1982	-0,0206	-10,394
2	0,6621	0,2017	30,4637
3	0,661	0,1992	30,1362
4	0,6315	0,0155	2,45447
5	0,9091	0,0511	5,62094
6	0,9107	0,0491	5,39146
7	0,7556	0,0299	3,95712
8	0,9928	0,0012	0,12087
9	0,9926	0,0013	0,13097