

Software Engineering Conference Russia
October 2017, St. Petersburg



Комплекс программ оценивания
надежности и планирования
разработки программных средств на
основе динамических моделей

Андрей Данилов, Nokia

Необходимо планировать качество и ресурсы

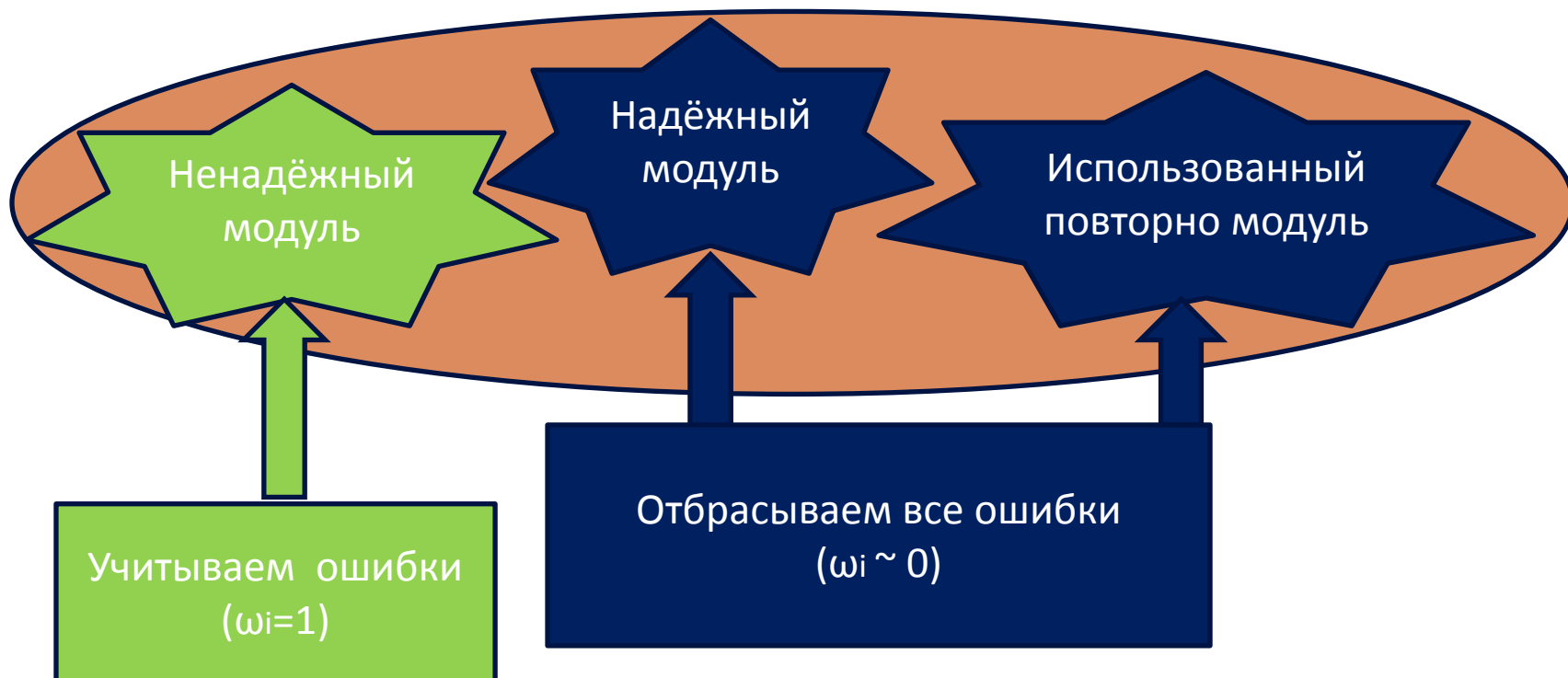
Все стандарты разработки ПО и реальные проекты требуют оценивания и планирования ресурсов на обеспечение заданных показателей качества:

1. ISO 9000.
2. CMM-I.
3. Agile.
4. ГОСТ Р 54732-2011/ISO/TS 10004:2010 - Национальный стандарт Российской Федерации менеджмент качества.



Известные модели строятся на результате двоичной классификации модулей ПС на классы потенциально приводящих к отказу ($\omega_i = 1$) и надёжных ($\omega_i = 0$)

ПО



Прогнозированию количества ошибок

1. Экспертные оценки.
2. Оценки с использованием исторических баз данных подобных проектов.
3. Оценка сложности кода.
4. Логистическая регрессия.

Как правило, используются комбинации нескольких подходов.



Логистическая регрессии и набора метрик Чайдамбера – Кемерера:

$$\omega_i = 1 / (1 + e^{-(\beta_0 + \sum \beta_j x_j)})$$

x_j - значение метрики сложности

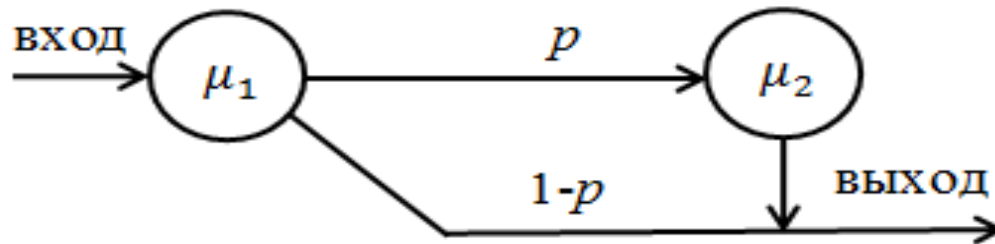
β_j - коэффициент, который находится методом максимального правдоподобия

Примеры базовых метрик:

1. NM – количество методов в классе
2. L[1..NM] - количество строк в методе класса
3. DIT – глубина дерева наследования
4. NOC – количество прямых потомков данного класса, и т.д.



Двухфазное обобщенное распределение Кокса



Параметры аппроксимирующего распределения Кокса:

$$\begin{cases} p = \mu_2 (f_1 \mu_1 - 1) / \mu_1 \\ \mu_1 = (\mu_2 f_1 - 1) / (\mu_2 f_2 - f_1) \\ \mu_2 = (f_1 f_2 - f_3 \pm \sqrt{D}) / (2(f_2^2 - f_1 f_3)), \end{cases}$$

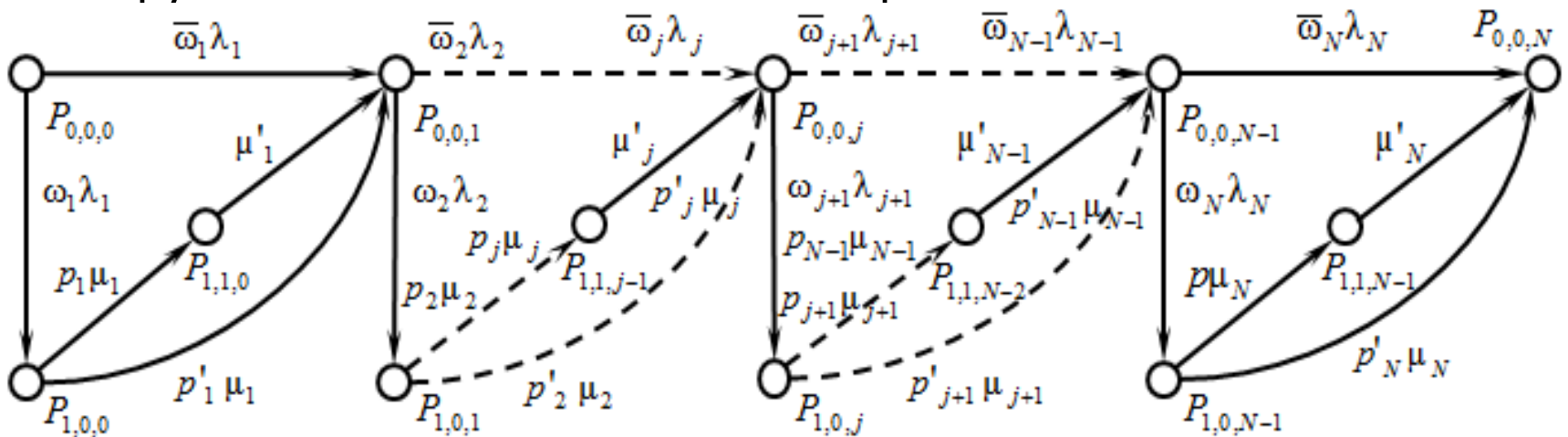
$$D = (f_1 f_2 - f_3)^2 - 4(f_2^2 - f_1 f_3)(f_1^2 - f_2); \quad f_i = g_i / i!; \quad i = \overline{1,3}$$

g_i – i -ый начальный момент исходного распределения.



Пример стратегии тестирования №1

Этап тестирования продолжается, пока не будет обнаружена или не определено отсутствие очередной ошибки. После обнаружения ошибки тестирование приостанавливается, ошибка исправляется. После исправления ошибки или её не обнаружения вновь начинается тестирование.



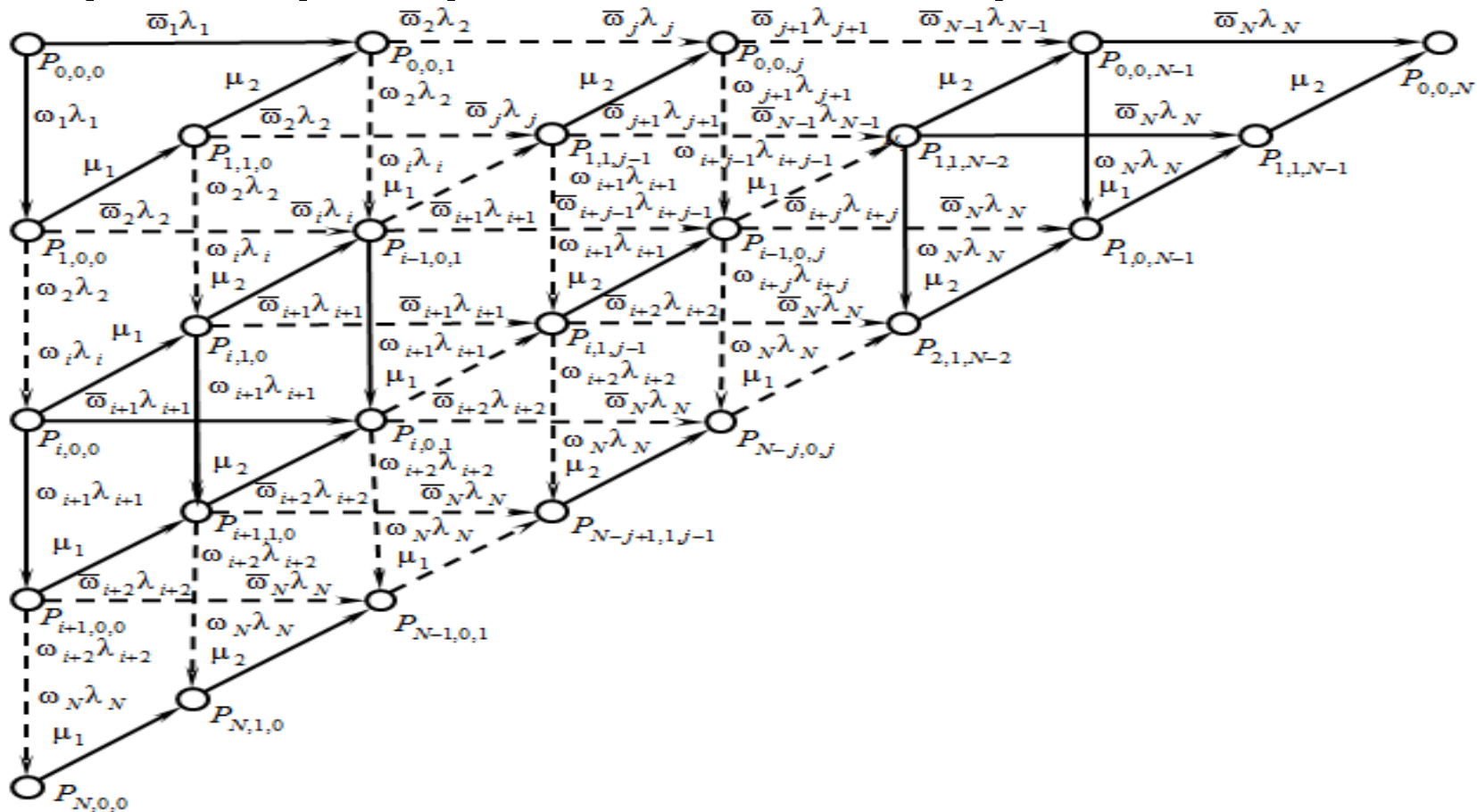
Пример стратегии тестирования №2

Обнаруженные ошибки устраняются последовательно по мере их выявления, а тестирование во время исправления найденных ошибок не приостанавливается.

Процесс исправления аппроксимируется распределением Эрланга.



Пример стратегии тестирования №2



Программный комплекс

1. На базе модифицированных размеченных графов составляются системы дифференциальных уравнений решение которых позволяет вычислить вероятностные показатели процессов испытаний и состояния программных средств.
2. Задав начальные условия к системе уравнений можно найти численное решение соответствующей задачи Коши для произвольного значения времени.



Программный комплекс

1. Решение уравнений целесообразно выполнять методом Рунге — Кутты.
2. Приемлемые характеристики устойчивости и точности решения обеспечиваются методом нумерации состояний - соответствующие матрицы состояний являются нижними треугольными.
3. Программный комплекс создан с использованием среды Matlab.



Программный комплекс

1. Для моделирования используется функция ode45 среды Matlab.
2. Эта функция использует формулы Рунге – Кутты 4-го и 5-го порядка.
3. На вход ode45 передаются функции для вычисления правых частей систем дифференциальных уравнений, интервал интегрирования и вектор начальных условий.



Матрица индексации состояний стратегия №2

Первая страница

1	2	3	4	5	6	7	8	9	10	11
22	23	24	25	26	27	28	29	30	31	0
41	42	43	44	45	46	47	48	49	0	0
58	59	60	61	62	63	64	65	0	0	0
73	74	75	76	77	78	79	0	0	0	0
86	87	88	89	90	91	0	0	0	0	0
97	98	99	100	101	0	0	0	0	0	0
106	107	108	109	0	0	0	0	0	0	0
113	114	115	0	0	0	0	0	0	0	0
118	119	0	0	0	0	0	0	0	0	0
121	0	0	0	0	0	0	0	0	0	0

Вторая страница

12	13	14	15	16	17	18	19	20	21	0
32	33	34	35	36	37	38	39	40	0	0
50	51	52	53	54	55	56	57	0	0	0
66	67	68	69	70	71	72	0	0	0	0
80	81	82	83	84	85	0	0	0	0	0
92	93	94	95	96	0	0	0	0	0	0
102	103	104	105	0	0	0	0	0	0	0
110	111	112	0	0	0	0	0	0	0	0
116	117	0	0	0	0	0	0	0	0	0
120	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0



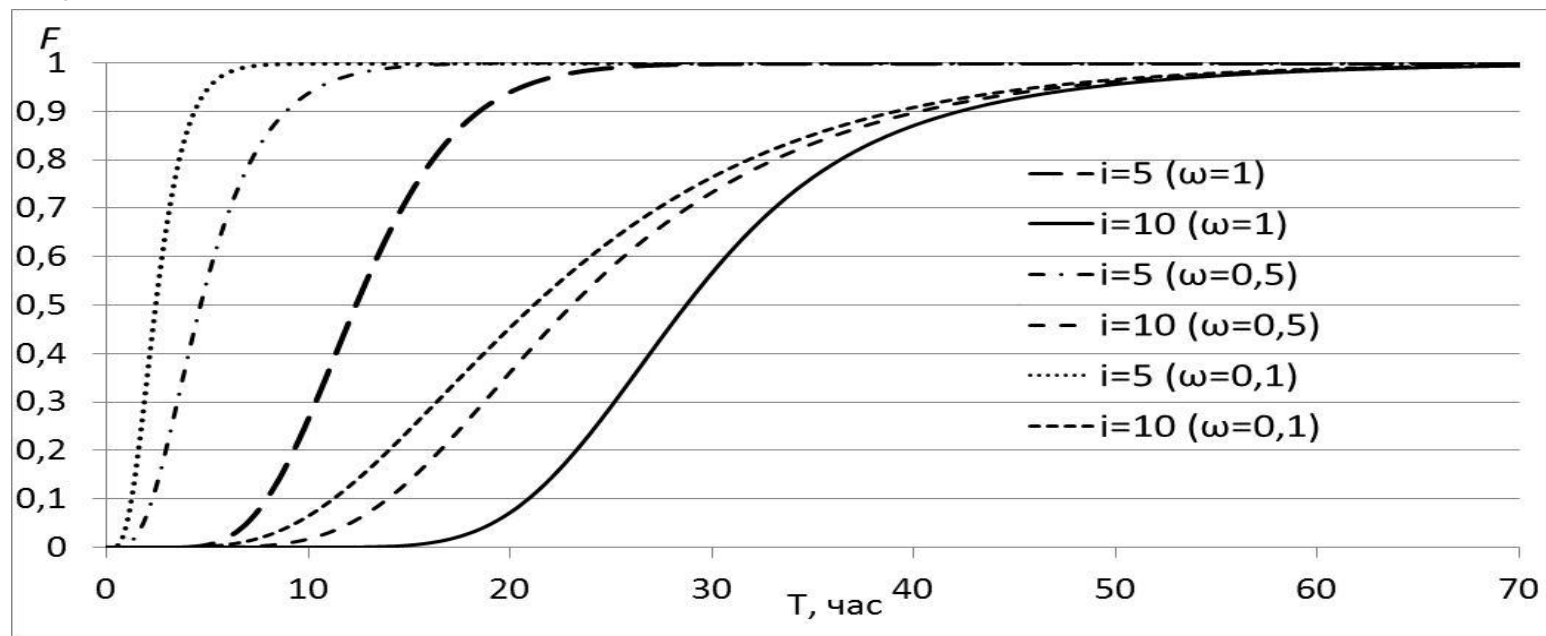
Пример исходных данных

1. Предполагается, что изначально в ПС содержится $N = 10$ ошибок со средними длительностями интервалов времени тестирования соответственно равными: 0,3; 0,3; 0,3; 0,7; 0,7; 1; 2; 3; 5; 10 (час).
2. Длительность интервала времени исправления ошибок равна 3 (час).
3. Значения вероятностей обнаружения ошибок ω_i , $i=(0,N)$, указываются непосредственно на графиках и в таблицах.



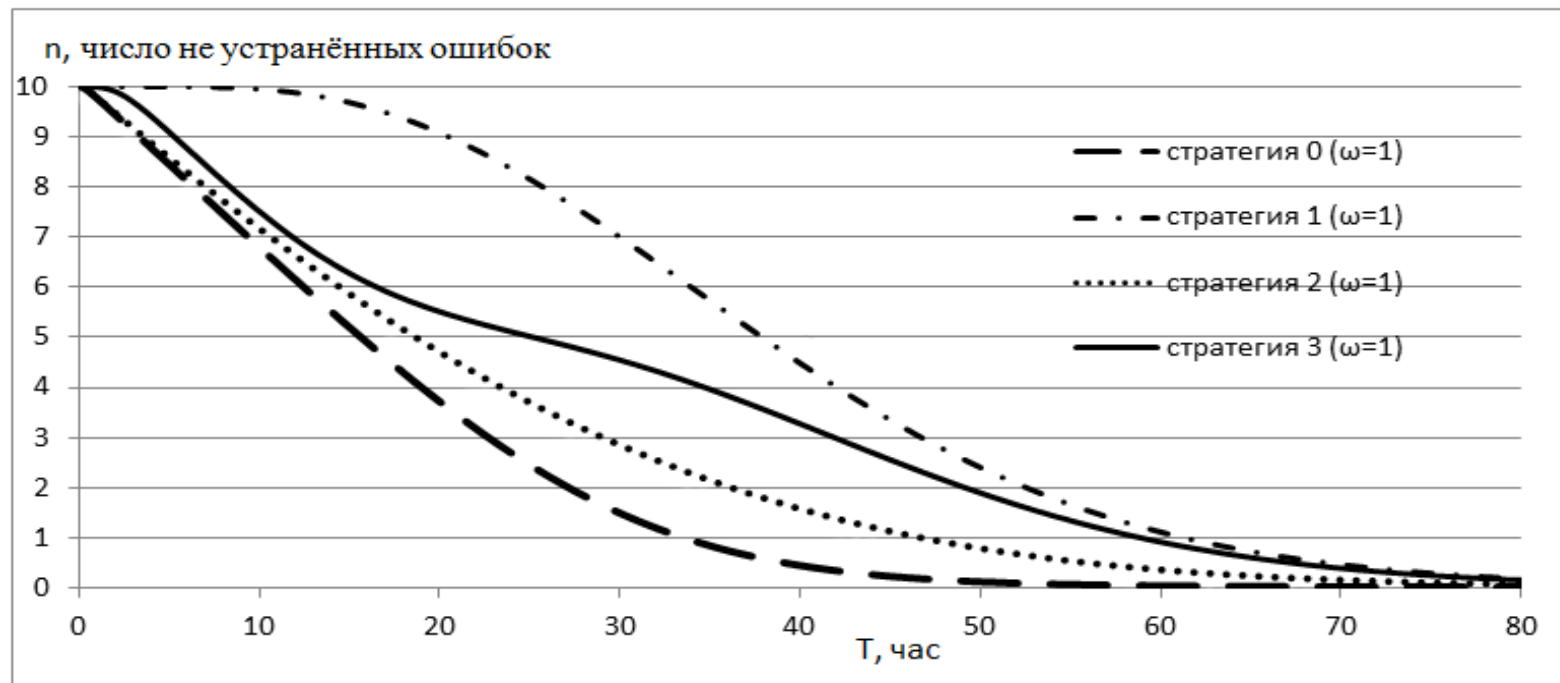
Примеры расчетов

Функции распределения времени устранения ошибок для стратегии №2



Примеры расчетов

Сравнение стратегий по числу не устраненных ошибок



Заключение

Аппроксимация произвольных законов распределений интервалов времени исправления ошибок распределением Эрланга и Кокса:

1. обобщает известные марковские модели,
2. повышает точность моделирования (до 12%)
3. делает предложенные модели более универсальными.



Заключение

Использование в моделях оценки надежности для каждого модуля позволяет:

1. Повышать точность моделирования процессов отладки ПС.
2. Учитывать состоятельность (мощность) тестов, различную степень доверия к различным тестам.
3. Искать пути повышения характеристик надёжности ПС посредством формирования тестов, обнаруживающих ошибки с высокой вероятностью.
4. Выбирать наилучшую стратегию испытаний ПС, исходя из наличия или отсутствия необходимых ресурсов, времени и статистических данных.
5. Планировать работы и ресурсы.

