ИНФОРМАЦИОННЫЙ ПОДХОД К ОРГАНИЗАЦИИ КОНТРОЛЯ ПРОЕКТОВ

А.И. Марон,
кандидат технических наук, доцент кафедры бизнес-аналитики
Национального исследовательского университета «Высшая школа экономики»

М.А. Марон,
генеральный директор ООО «Технология Бизнеса»,
студент магистерской программы «Бизнес-информатика»
Национального исследовательского университета «Высшая школа экономики»

E-mail: amaron@hse.ru
Адрес: г. Москва, ул. Кирпичная, д. 33/5

Предложен новый метод определения контрольных точек в проектах. В основу метода положен информационный подход к решению задач диагностики объектов, представленных функциональными моделями.

Ключевые слова: проект, энтропия, контрольные точки, проверка.

1. Введение

При выполнении проекта исключительно важно своевременно проверить, правильно ли были выполнены работы. В противном случае, при кажущемся соблюдении графика в какой-то момент будут выявлены ошибки, для исправления которых понадобится много времени. Для уменьшения риска срыва сроков проекта надо проводить своевременные промежуточные проверки правильности выполнения работ. Однако нельзя впадать и в другую крайность: тотальный контроль займет слишком много времени и средств. Количество проверок определяется исходя из средств, которые спонсор проекта готов выделить на их осуществление. Возникает вопрос: как выбрать в проекте контрольные точки? Эта задача существенно отличается от задачи выбора контрольных точек для проверки правильности функционирования действующих программ [1, 2].

Для ответа на поставленный вопрос рассмотрим подробнее работу руководителя проекта на этапе планирования [3]. Руководитель проекта осуществляет планирование в два этапа: структурное планирование и календарно-ресурсное планирование. На этапе структурного планирования определяются перечень работ и логические связи между ними, результатом является сетевой график проекта. На этапе календарно-ресурсного планирования определяются длительности работ, а также трудовые, материальные и финансовые ресурсы, необходимые для выполнения работ. Определение набора контрольных точек можно осуществить после того, как разработан сетевой график (граф проекта) и установлены длительности работ. Таким образом, определение набора контрольных точек является необходимым дополнительным этапом планирования проекта. Результатом этого этапа является запланированный набор проверок, который будем записывать в виде последовательности номеров работ, после выполнения которых производятся проверки. В данной статье ограничимся рассмотрением проектов, графы которых имеют одну конечную вершину. Проверка после последней работы проекта является обязательной и, соответственно, не является элементом свободного выбора (*рис. 1*). Поэтому в дальнейшем, говоря о выборе m контрольных точек, будем подразумевать выбор среди работ, которым соответствуют внутренние вершины графа проекта. Так последовательность – набор $S = (1; 5; 6)$ означает, что при реализации проекта проверки будут проведены после работ, которым соответствуют вершинам графа с номерами 1, 5 и 6, а также после конечной работы, которой соответствует вершина 8. Эту последовательность из $(m + 1)$ проверок будем обозначать как S^{+1}. В рассматриваемом примере $S^{+1} = (1; 5; 6; 8)$, на *рис. 1* эти проверки показаны в виде жирных стрелок.

Как и все другие элементы плана, запланированные проверки будут выполняться при осуществлении фазы жизненного цикла «реализация проекта». При этом выполнение проверки может иметь два исхода: положительный и отрицательный результаты. При каждом из них возникает свой сценарий дальнейших действий команды проекта. При положительном результате выполнение плана продолжается до следующей запланированной проверки. Если все проверки последовательности S^{+1} имеют положительные результаты, то проект реализован без ошибок. Если же проверка имеет отрицательный результат, то это свидетельствует о том, что работа, после которой она выполнена, содержит ошибку; либо ошибка произошла в работах, которые являются ее предшественниками, но не являются предшественниками работ, после которых ранее выполнялись запланированные проверки. Среди таких работ необходимо найти работу, содержащую ошибку. В терминах технической диагностики это задача поиска неисправного элемента в подграфе исходного графа объекта диагноза, представленного функциональной моделью, с точностью до неисправного элемента [4]. Так, применительно к рассматриваемому примеру, если проверка Π^1 дала положительный результат, а следующая проверка Π^5 имеет отрицательный результат, то ошибка может содержаться в одной из работ 2, 3, 4 или 5, которым соответствует подграф графа проекта, выделенный на *рис. 1.*

Определимся с критерием оптимальности набора контрольных точек в проекте.

2. Критерий оптимальности набора контрольных точек в проекте

Если проект состоит из *n* работ, то *m* контрольных точек ($m < n$) можно выбрать различным образом. Определим критерий, по которому можно сравнить эти варианты выбора контрольных точек в проектах. Для этого подробнее остановимся на процедуре определения контрольных точек и тех действиях, которые выполняются после проведения проверки.

Рассмотрим для конкретности изложения программные проекты, которые связаны с созданием новых решений. Для них принципиально важным является выход на рынок в запланированные сроки [5]. Соответственно, наиболее значимым для таких инновационных проектов является риск значительного увеличения сроков из-за необходимости поиска и устранения своевременно не обнаруженных ошибок. Предположим, что работа, содержащая ошибку, локализована после выполнения проверки Π^k с точностью до подграфа *G,* аналогичного приведенному на *рис. 1*, в котором вершина k является конечной и для локализации ошибки необходимо заново в некоторой последовательности выполнить работы $g \in G$ до тех пор, пока проверка Π^k не даст положительный результат. В худшем случае это произойдет после того, как заново будут выполнены все работы, которым соответствуют работы подграфа *G.* Такая ситуация будет иметь место тогда, когда последней будет переделана именно та работа, ко-

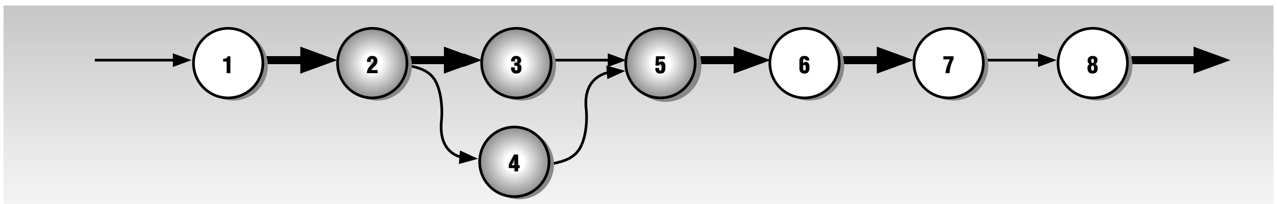

Рис. 1. Пример сетевого графика проекта с запланированными проверками

торая содержит ошибку. В этом случае время локализации ошибки будет равно суммарному времени выполнения всех работ, которым соответствуют вершины $g \in G$. Обозначим эту величину через $T(G)$:

$$T(G) = \sum_{g \in G} t_g, \tag{1}$$

где t_g – длительность работы, которой соответствует вершина с номером g.

Каждому набору контрольных точек S соответствует свой набор подграфов $\{G_s\}$, с точностью до которых может быть локализована работа, содержащая ошибку. Каждому подграфу G из множества $\{G_s\}$ соответствует своё значение $T(G)$ – максимального времени поиска работы, содержащей ошибку. Среди них есть подграф G^*, которому соответствует максимальное значение

$$T^*(S) = \max_{G \in \{G_s\}} T(G) \tag{2}$$

Так, в рассматриваемом примере имеем следующие подграфы: (1); (2,3,4,5); (6); (7,8). Допустим, работы имеют длительности (дней): 6; 4; 3; 2; 5; 1; 10, 3. Тогда суммы длительностей работ для указанных подграфов будут равны: 6; 14; 1; 13. Соответственно, $T^*(s) = 14$ дней.

Величину $T^*(s)$ будем использовать для характеристики набора контрольных точек s. При этом чем меньше значение $T^*(s)$, тем лучше набор контрольных точек. Другими словами, если предложены два различных набора контрольных точек s^1 и s^2 и $T^*(s^1) < T^*(s^2)$, то выбираем набор s^1. Обозначим множество различных наборов заданного числа контрольных точек, которые можно предложить для проекта через $\{S_m\}$. Как отмечалось ранее, если в графе проекта n вершин, а можно выбрать m контрольных точек, то количество вариантов $|\{s\}| \le C_n^m$, где C_n^m число сочетаний из n элементов по m.

Будем считать оптимальным тот набор контрольных точек S^*, при котором достигается

$$T^*(s^*) = \min_{S \in \{S_m\}} T^*(S) = \min_{S \in \{S_m\}}[\max_{G \in \{G_s\}} T(G)] = \min_{S \in \{S_m\}}[\max_{G \in \{G_s\}}(\sum_{g \in G} t_g)] \tag{3}$$

В принципе, задачу нахождения набора контрольных точек, оптимального по предложенному критерию, можно решить полным перебором, как любую задачу целочисленной оптимизации. Однако, как это и бывает на практике для большинства задач целочисленной оптимизации, выполнить такой перебор для реального проекта невозможно. Поэтому придется прибегнуть к эвристическим алгоритмам выбора контролируемых параметров. Сравнение их относительной эффективности будет осуществлено исходя из предложенного критерия.

3. Математическая постановка задачи

В простейшем случае изложенная проблема допускает следующую формальную постановку.

Имеется проект, состоящий из $(n+1)$ работ, связанных логическими связями типа «Finish to Start». Структурная схема проекта задана ориентированным графом, вершины которого соответствуют работам, а дуги логическим связям между ними. Дуга из вершины с номером i в вершину с номером j означает, что работа j не может быть начата раньше, чем закончена работа i. Граф имеет одну конечную вершину с номером $(n+1)$ и, соответственно, n – внутренних вершин. Для проверки правильности выполнения работ можно выполнить $m < n$ промежуточных проверок. Проверка после выполнения последней работы проекта выполняется обязательно. Известно, что проверка Π^k, осуществлённая после выполнения работы k ($k = 1, 2, ..., n$) имеет положительный результат π_1^k, тогда и только тогда, когда работа k выполнена правильно и правильно выполнены все работы ей логически предшествующие. В противном случае проверка Π^k будет иметь отрицательный результат π_0^k, означающий, что либо работа k выполнена неправильно, либо неправильно выполнена одна и только одна работа, ей предшествующая. Если проект является типовым, то из предыдущего опыта известны вероятности неправильного выполнения работ. Вероятность неправильного выполнения работы i равна p_i. В этом случае требуется определить, после каких работ следует выполнять проверки.

4. Метод расстановки точек контроля без учета длительности работ

Предлагается следующий метод решения данной задачи.

Введём случайную величину N – номер работы выполненной неправильно. По условию задачи она может принять одно из значений $\{1; 2; ...; n; (n+1)\}$. Кроме того, все $(n+1)$ работ проекта могут быть выполнены правильно. Вероятность этого события дополняет до единицы сумму всех p_i. Для простоты изложения будем считать её пренебрежимо малой. Тогда случайной величине N можно поставить в соответствие энтропию

$$H(N) = -\sum_{i=1}^{n+1} p_i \cdot \log_2 p_i \tag{4}$$

Присвоим проверкам номера, равные номерам работ, после которых они осуществляются, т. е. будем говорить, что проверка Π^k это проверка с номером k. Введем случайную величину π^k – результат проверки с номером k. Как сказано в условии, она может принять одно из двух значений: π_1^k или π_0^k. Причем вероятности этих значений зависят от того, какие работы предшествуют работе k. Осуществление проверки Π^k уменьшает неопределённость относительно того, какая работа выполнена неправильно. Соответствующую информацию можно найти как меру уменьшения неопределённости по формуле

$$I(N;\pi^k) = H(N) - H(N/\pi^k). \qquad (5)$$

Условная энтропия $H(N/\pi^k)$ является средней мерой неопределенности, которая останется при условии, что проверка будет осуществляться после выполнения работы k.

Можно доказать, что количество информации $I(N;\pi^k)$ достигает максимума, если выполняется проверка, неопределённость результата которой $H(\pi^k)$ максимальна.

При этом процедура вычисления энтропии $H(\pi^k)$ по формуле (6) проще, чем процедура расчета по формуле (5):

$$H(\pi^k) = -P(\pi_0^k)\cdot log_2[P(\pi_0^k)] - P(\pi_1^k)\cdot log_2[P(\pi_1^k)], \qquad (6)$$

где вероятность $P(\pi_0^k)$ равна сумме вероятности неправильного выполнения работы k и всех работ, ей предшествующих; вероятность $P(\pi_1^k)$ дополняет эту величину до единицы.

Нахождение проверки, имеющей максимальную энтропию, можно осуществить методом перебора. Допустим, что это проверка с номером k_1. Она наиболее информативна. Примем ее в качестве первой проверки, которую надо выполнить при реализации проекта. Рассмотрим ситуацию, которая возникнет, когда после реализации работы k_1 действительно будет выполнена проверка. Если будет получен отрицательный результат, то придется осуществить поиск неправильно выполненной работы среди подмножества работ, неправильное выполнение одной из которых приводит к этому результату. Обозначим множество номеров таких работ через $G_0(k_1)$. Заметим, что алгоритм поиска может быть различным, но в любом случае он будет продолжаться до обнаружения неправильно выполненной работы, а процесс завершается необходимыми исправлениями, а возможно и выполнением этой работы заново.

Если будет получен положительный результат проверки, то это означает, что работа k_1 и все работы ей логически предшествующие, т. е. все работы с номерами из $G_0(k_1)$, выполнены правильно. В этом случае выполнение проекта продолжается в обычном порядке. Обозначим через $G_1(k_1)$ множество номеров работ, не входящих в $G_0(k_1)$. Именно среди этих работ необходимо искать ту, после которой надо выполнить следующую проверку. Для того чтобы точно вычислить энтропию результата проверки с номером $r \in G_1(k_1)$ придётся для каждой работы $i \in G_1(k_1)$ определить вероятности неправильного выполнения, при условии, что неправильно могут быть выполнены только работы с номерами из $G_1(k_1)$. Эти байесовские вероятности $p_i(k_1)$ можно найти по формуле

$$p_i(k_1) = p_i / P(G_1(k_1)), \qquad (7)$$

где $P(G_1(k_1)$ – сумма исходных вероятностей неправильного выполнения работ $i \in G_1(k_1)$.

На основании вышеизложенного можно предложить следующий метод последовательного выбора работ проекта, после которых надо осуществлять проверки.

1. Для всех $k = 1, 2, ..., n$ вычислим энтропию результата проверки Π^k по формуле (6). Определим k_1 – номер проверки, энтропия результата которой максимальна. Примем эту проверку в качестве первой проверки в проекте.

2. Удалим из дальнейшего рассмотрения подмножество работ проекта, состоящее из работы k_1 и всех работ ей логически предшествующих. Останется подмножество работ с номерами принадлежащими $G_1(k_1)$. Осуществим перерасчёт вероятностей по формуле (7).

3. Для всех проверок с номерами r, принадлежащими $G_1(k_1)$, выполним расчет по формуле (6). Определим номер проверки, энтропия результата которой максимальна.

4. Выполним действия, аналогичные указанным в пунктах 3 и 4, пока не будут выбраны все m проверок.

5. Полученная последовательность является плановой для проекта.

Необходимо отметить, что предложенный метод основан исключительно на информационном подходе к выбору проверок. Соответственно, набор проверок будет предопределён топологией графа работ проекта и распределением вероятностей. В результате первая проверка осуществит половинное разбиение графа работ по вероятностям. Следующая будет осуществлять подобное разбиение для подграфа работ, не являющихся логическими предшественниками работы, которой соответствует первая проверка и т.д. При этом длительности работ не учитываются. В результате до первой проверки может оказаться уже выполненным слишком большой объем работ проекта. Если после проверки окажется, что какая-то работа выполнена неправильно, то суммарное время поиска и устранения ошибки может оказаться очень большим.

5. Метод последовательного выбора проверок с учетом длительности работ проекта

Оптимальным по критерию (3) является набор проверок, при котором минимальным является максимальное суммарное время поиска ошибки среди групп работ, с точностью до которых этот набор проверок позволяет локализовать неправильно выполненную работу. Поэтому при поиске мест проведения проверок надо учитывать не только топологию сетевого графика проекта и распределения вероятностей неправильного выполнения работ, но и их длительности. Постановка задачи аналогична приведенной ранее, с тем дополнением, что учитывается длительность работ: длительность работы i равна t_i.

Предлагается следующий эвристический подход к решению поставленной задачи.

Будем выбирать точку проведения каждой проверки, исходя из значения критерия выбора. Критерий выбора очередной проверки построим, основываясь на энтропии ее результата и суммарном времени работ, которым соответствуют вершины подграфа, с точностью до которого при отрицательном результате этой проверки будет локализована работа, содержащая ошибку, при условии, что все предыдущие проверки дали положительные результаты. Энтропия результата – положительная характеристика проверки: чем она больше, тем больше информации дает проверка. Суммарное время работ $T(G)$ в подграфе G, о котором шла речь выше, характеристика негативная: чем это время больше, тем труднее найти работу, содержащую ошибку. Примем в качестве критерия выбора дробь R_k, числителем которой является энтропия результата проверки, а в знаменателе стоит функция $T(G)$. Определим вид этой функции. Требования к ней таковы:

$F(T(G))$ – возрастающая положительная функция;

скорость её изменения существенно меньше скорости изменения $T(G)$;

$F(0)=1$.

Первое из этих требований следует из того, что значение R_k должно уменьшаться при росте $T(G)$. Второе требование связано с необходимостью по возможности уравнять диапазон изменения числителя и знаменателя, а также обеспечить устойчивость критерия выбора к малым ошибкам в определении времени работ. Третье требование обеспечит эквивалентность данного решающего правила и информационного подхода, когда время не учитывается.

Всем трем условиям удовлетворяет функция

$$F(T_k) = 1 + log_2(1 + T(G)) \qquad (8)$$

Соответственно, критерий выбора можно записать следующим образом:

$$R_k = H(\pi^k) / [1 + log_2(1 + T(G))] \qquad (9)$$

Основываясь на этом критерии выбора, будем на каждом шаге построения последовательности проверок, выбирать ту проверку, при которой значение (8) максимально. Такое решающее правило позволяет предложить следующий алгоритм построения набора контрольных точек в проекте.

1. Для всех k = 1, 2, ..., n вычислим энтропию результата проверки Π^k по формуле (6). Вычислим значение критерия R_k по формуле (9). Определим k_1 – номер проверки, при которой значение R_k максимально. Примем эту проверку в качестве первой проверки в проекте.

2. Удалим из дальнейшего рассмотрения подмножество работ проекта, состоящее из работы k_1 и всех работ ей логически предшествующих. Останется подмножество работ с номерами принадлежащими $G_1(k_1)$. Осуществим перерасчёт вероятностей по формуле.

3. Для всех проверок с номерами r, принадлежащими $G_1(k_1)$, выполним расчёт по формуле. Определим номер проверки, энтропия результата которой максимальна.

4. Выполним действия, аналогичные указанным в пунктах 3 и 4 пока не будут выбраны все m проверок.

5. Полученная последовательность является плановой для проекта.

6. Методика сравнения эффективности методов выбора контрольных точек в программных проектах

Для относительной оценки эффективности предложенных методов проведем расчеты для различных графов проектов при различных распределениях вероятностей ошибок и длительностей работ. Число контрольных точек также будет переменным. Кроме того, сравним эти методы с одним из методов, не учитывающих логические связи между работами. Например, с методом, при котором все работы проекта упорядочиваются в порядке убывания длительности и первые m работ принимаются в качестве мест проведения промежуточных проверок. Этот метод в дальнейшем будем называть простейшим.

Таким образом, переменными являются:

✦ граф проекта;

✦ длительности работ;

✦ вероятности неправильного выполнения работ, при условии, что одна из работ проекта содержит ошибку;

✦ количество контрольных точек.

Рассмотрим каждую из этих величин и примем ограничения, при которых будет проведен эксперимент.

Граф проекта может быть самым различным. Ограничимся случаями линейной структуры, при которой проект состоит из простой последовательности работ (цепочки), и ветвления, подобного приведенному на *рис. 2.* Этот тип графа будем называть разветвленным. В нем на каждом уровне, кроме последнего, находятся две вершины, одна из которых является предшественником одной вершины следующего уровня, а вторая напрямую связана с конечной вершиной.

Расчёты будем проводить при числе вершин 10, 15, 20, 30 для каждого из таких графов.

Для указанного выше числа вершин количество контрольных точек примем равным: 3, 5. 6 и 8, соответственно.

Длительности работ будем генерировать из целых чисел диапазона от 1 до 5 с помощью датчика случайных чисел. Такой диапазон принят исходя из рекомендаций о детализации задач в проектах, в целях лучшего контроля исполнения [6].

Вероятности ошибок в работах будем считать равными отношению их длительности к суммарной длительности всех работ проекта. На основании того, что в условиях отсутствия статистических данных вероятность неправильного выполнения работы тем больше, чем больше ее трудоемкость, которую и отражает длительность.

Для каждого графа при заданном количестве контрольных точек расчеты будут выполнены для трех вариантов распределения длительностей работ и, соответственно, вероятностей ошибок. Для каждого сгенерированного таким образом проекта выбор контрольных точек будет произведен тремя методами. Для каждого полученного набора будет рассчитано значение критерия оптимальности (3) и определен наилучший набор и, соответственно, метод, ему соответствующий. Для выполнения указанных действий разработан специальный программный комплекс.

Результаты проведенного исследования для топологий линейная цепочка и разветвленная структура представлены в *табл. 1* и *2.*

Наилучшие значения, полученные после сравнения результатов расчета критерия, выделены цветом в соответствующих ячейках.

По результатам проведенного исследования можно сделать следующие выводы.

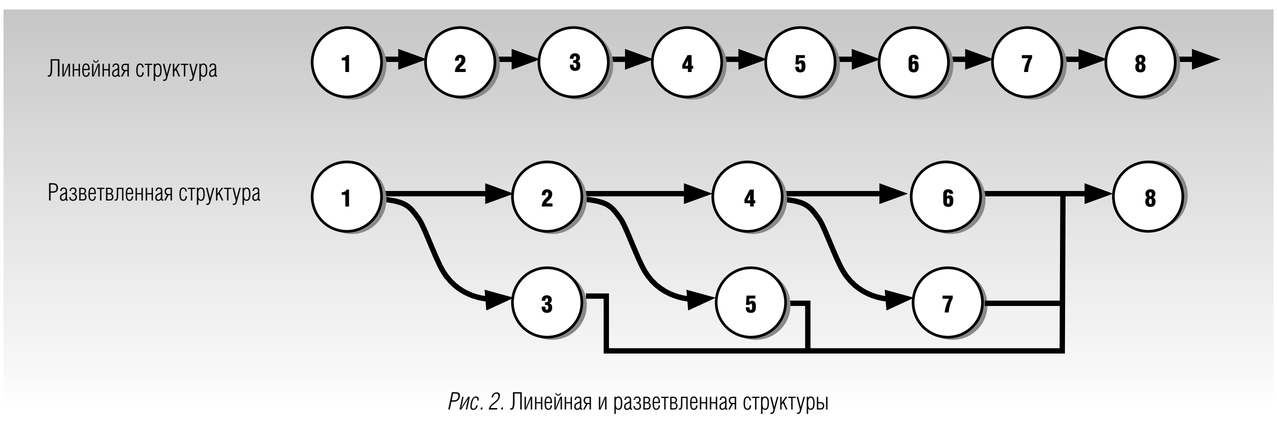

Рис. 2. Линейная и разветвленная структуры

Таблица 1.

Кол-во работ	Номер	Кол-во проверок	Значение критерия для «простейшего» метода	Значение критерия для информационного метода	Значение критерия для информационного метода с учетом длительности работ
10	1	3	14	13	8
10	2	3	15	13	9
10	3	3	14	12	8
15	1	5	26	23	19
15	2	5	24	23	19
15	3	5	27	22	17
20	1	6	30	31	27
20	2	6	29	29	27
20	3	6	33	32	28
30	1	8	28	45	35
30	2	8	34	43	34
30	3	8	31	41	34

Таблица 2.

Кол-во работ	Номер	Кол-во проверок	Значение критерия для «простейшего» метода	Значение критерия для информационного метода	Значение критерия для информационного метода с учетом длительности работ
10	1	3	12	13	9
10	2	3	13	13	10
10	3	3	14	12	10
15	1	5	23	23	16
15	2	5	22	23	16
15	3	5	21	22	15
20	1	6	25	31	20
20	2	6	27	29	21
20	3	6	26	32	19
30	1	8	43	45	38
30	2	8	44	43	39
30	3	8	45	41	37

Оба предложенных метода в большинстве случаев дают лучшие результаты, чем простейший метод. Однако для линейной структуры сетевого графика проекта максимальное количество контрольных точек, которое может быть выбрано с помощью информационного метода, не превышает

$$m \leq \lceil log_2 N \rceil \tag{10}$$

Метод, учитывающий длительности работ показал результаты лучшие, чем метод, не учитывающий длительности работ. Однако при линейной структуре работ проекта ограничение по числу контрольных точек существует и для него.

7. Заключение

В работе предложен метод расстановки точек контроля в проектах, основанный на максимизации информации о номере работы, содержащей ошибку. Предложена модификация информационного подхода для поиска мест расстановки точек контроля, при которой учитывается длительности работ и глубина локализации ошибок. Предложен критерий оптимальности относительно, которого будет производиться сравнение наборов контрольных точек в проектах.

Разработана программа для выбора контрольных точек в проектах, управление которыми осуществляется с применением MS Project. Предложена методика проведения экспериментов по сравнению эффективности разработанных методов выбора контрольных точек и ранее известного метода, основанного исключительно на учёте длительностей работ проекта. Для проведения эксперимента разработана программа генератор исходных данных для двух типовых структур графа проекта: простая последовательность и разветвлённый граф. ■

Литература

1. Ананьин В.М. Устойчивость управления IT-проектами в условиях неопределенности // Управление проектами. – 2005, №1-2.
2. Марон А.И. Марон М.А. Оптимизация контроля в программных проектах разработки больших систем //Труды XXXVIII Международной конференции «Информационные технологии в науке, социологии и бизнесе. IT+S&E' 10», Украина, Крым, Ялта-Гурзуф, 20-30 мая 2010 г. – с. 46-48.
3. Ципес Г.Л., Товб А.С. Проекты и управление проектами в современной компании. Учебное пособие. – М.: Олимп-Бизнес, 2009.
4. Пархоменко П.П., Согомонян Е.С. Основы технической диагностики: Оптимизация алгоритмов диагностирования, аппаратурные средства / Под ред. П.П.Пархоменко. – М.: Энергия, 1981.
5. Липаев В.В. Программная инженерия. Методологические основы. – М.: ГУ-ВШЭ, ТЕИС, 2006.
6. Баркалов С.А., Воропаев В.И., Секлетова Г.И. Математические основы управления проектами.– М.: Высшая школа, 2005.