

А.И. Крапчатов

ОПТИМАЛЬНАЯ СИСТЕМА ОТЛАДКИ СЛОЖНЫХ КОМПЛЕКСОВ ПРОГРАММ

Рассматриваются методы разбиения структуры комплекса программ на части, их автономной отладки, последовательного объединения отлаженных частей и их последующей связанной отладки.

Ключевые слова: стратегия системной отладки, горизонтальные и вертикальные стратегии, затраты на системную отладку. технология программных модулей.

В настоящее время применяются различные стратегии системной отладки, среди которых выделяются группы горизонтальных и вертикальных стратегий. Стратегии отладки первой группы предусматривают последовательную отладку комплекса программ по уровню структуры («сверху вниз» или «снизу вверх»). Ко второй группе относятся стратегии отладки, учитывающие технологию (последовательность) реализации программных модулей в процессе функционирования комплекса программ¹.

Реализация любой стратегии требует разбиения структуры комплекса программ на части, автономной отладки полученных частей программы, последовательного объединения автономно отлаженных частей и их связной отладки вплоть до исходной структуры комплекса программ.

Выбор стратегии системной отладки предлагается осуществлять на основе использования графа Г и схем исследуемого комплекса программ. Граф Г представляет собой ориентированный граф, вершинами которого являются подпрограммы комплекса, дугами – связи по управлению между ними. Вершинам графа поставлены в соответствие возможные последовательности реализации вершин нижестоящего уровня.

Множество стратегий системной отладки определяется в первую очередь множеством возможных вариантов разбиения графа Γ на подграфы с целью их автономного тестирования и множеством возможных вариантов объединения полученных подграфов с целью их связного тестирования. Поэтому любая стратегия системной отладки задает множество подграфов разбиения графа Γ :

$P^m = \{p_1, \dots, p_6, \dots, p_M^m\}$ и последовательность объединения этих подграфов до исходной графовой структуры:

$$\tilde{P}^{mn} = \{\tilde{p}_1^{mn}, \dots, \tilde{p}_k^{mn}, \dots, \tilde{p}_N^{mn}\},$$

где \tilde{p}_N^{mn} совпадает с графом Γ .

Задача выбора оптимальной стратегии системной отладки состоит в выборе разбиения P^m^* , графа Γ и последовательности объединения подграфов \tilde{P}^{mn}^* , обеспечивающих оптимальные временные или стоимостные характеристики системной отладки.

Временные (стоимостные) затраты на системную отладку с использованием определенной m -стратегии ее организации складываются из двух составляющих $\bar{T}_{mn}^{разб}(C_{mn}^{разб})$, $T_{mn}^{объед}(C_{mn}^{объед})$, первая из которых характеризует время (стоимость) автономной отладки всех подграфов графа Γ , полученных в результате его разбиения по m -стратегии, вторая – время (стоимость) проведения последовательности этапов объединения подграфов и их связной отладки по m -стратегии.

Время и стоимость автономной отладки подграфов разбиения определяются выражениями:

$$\bar{T}_{mn}^{разб} = \sum_v t_{vmn}; \quad C_{mn}^{разб} = \sum_v C_{vmn},$$

где t_{vmn} и C_{vmn} – время и стоимость автономной отладки v -го подграфа разбиения.

Время автономной отладки может быть также определено с учетом возможностей одновременной отладки некоторого множества подграфов разбиения в случае наличия соответствующих ресурсов вычислительной техники и программистов:

$$\bar{T}_{mn}^{разб} = \max_{v \in V_j} \{t_{vmn}\},$$

где V_j – множество индексов подграфов, которые отлаживаются параллельно.

Время и стоимость отладки при объединении подграфов разбиения определяются следующим образом:

$$\bar{T}_{mn}^{объед} = \sum_k b_{kmn}, \quad \bar{C}_{mn}^{объед} = \sum_v S_{vmn},$$

где b_{kmn} , S_{kmn} – время и стоимость связанной отладки на k -м этапе объединения подграфов.

Задача выбора оптимальной стратегии с учетом временного критерия в общем виде формируется следующим образом: найти минимум

$$\sum_{mn} (\bar{T}_{mn}^{разб} + \bar{T}_{mn}^{объед}) x_{mn}$$

при ограничении

$$\sum_{mn} (\bar{C}_{mn}^{разб} + \bar{C}_{mn}^{объед}) x_{mn} \leq C_{зад},$$

где

$$x_{mn} = \begin{cases} 1, & \text{если выбирается } mn\text{-стратегия системной отладки;} \\ 0 & \text{в противном случае;} \\ C_{зад} & \text{допустимые затраты на проведение системной отладки.} \end{cases}$$

Выбор варианта разбиения графа Γ на подграфы при решении задачи выбора оптимальной стратегии системной отладки сводится к выбору состава V групп, где V – число программных модулей комплекса при соответствующих ограничениях на возможные комбинации программных модулей в группах. Выбор варианта объединения полученных при разбиении подграфов графа Γ до исходной графовой структуры сводится к выбору состава этапов объединения V^* непустых подграфов в исходную графовую структуру. Максимальное число этапов равно числу непустых подграфов V^* .

Для постановки задачи выбора оптимальной стратегии системной отладки с учетом приведенных положений введем переменные:

$$x_{rv} = \begin{cases} 1, & \text{если } r\text{-й программный модуль включается в } v\text{-ю} \\ & \text{группу (подграф) при разбиении графа } \Gamma; \\ 0 & \text{в противном случае;} \end{cases}$$

$$y_{vk} = \begin{cases} 1, & \text{если } v\text{-я группа (подграф) включается в } k\text{-й этап} \\ & \text{связной отладки, и } \sum_v x_{rv} > 0 \\ 0 & \text{в противном случае.} \end{cases}$$

А.И. Крапчатов

Рассмотрим получение временных и стоимостных характеристик $\bar{T}^{разб}$, $\bar{C}^{разб}$, $\bar{T}^{объед}$, $\bar{C}^{объед}$ этапов системной отладки при решении задачи выбора оптимальной стратегии с использованием описанного выше подхода.

Временные и стоимостные характеристики системной отладки при использовании определенной стратегии определяются временными и стоимостными характеристиками множества тестов, необходимых для проверки соответствующих структур графа Г.

При использовании различных стратегий отладки применяются тесты двух типов. Тесты первого типа используются для автономной отладки структур комплекса программ, полученных при его разбиении, тесты второго вида – для проверки объединений структур комплекса программ.

Специфической особенностью тестов первого типа является использование таких инструментов тестирования, как «заглушки» и «драйверы». Необходимость их использования объясняется тем, что при разбиении исходной структуры комплекса программ на части требуется моделирование внешних (для полученных частей комплекса) связей по управлению и информации. При этом моделирование информационных связей и связей по управлению, направленных к исследуемой части программ, осуществляется при помощи подпрограмм «драйверов». Моделирование информационных связей и связей по управлению, направленных от исследуемой части программы, осуществляется при помощи подпрограмм «заглушек».

При разбиении исходного комплекса программ на части существенно снижается трудоемкость тестирования и локализации ошибок, так как тесты создаются для несложных маршрутов обработки информации. Однако на создание подпрограмм «заглушек» и «драйверов» требуются дополнительные затраты времени и средств. Эти затраты могут быть достаточно большими, так как создание моделей внешних информационных и логических связей по трудоемкости приближается к трудоемкости создания частей программы, которые они замещают.

При синтезе тестов второго типа каждая часть программы, проверенная на предшествующих этапах отладки, заменяется одной вершиной на графике Г, которая выполняет известную функцию, что существенно облегчает организацию процесса тестирования для определенных структур комплекса программ.

Времена и стоимости автономной отладки подграфов разбиения графа Г с использованием тестов первого типа определяются путем выделения на общей схеме комплекса программ частей, соответствующих программным модулям, включенным в подграф, и

анализа временных и стоимостных характеристик тестов, необходимых для проверки этого подграфа.

Время t_v и стоимость C_v автономной отладки каждого v -го подграфа разбиения включают времена и стоимости подготовки данных для тестирования, реализации процесса тестирования и локализации обнаруженных ошибок:

$$t_v^{nod} + t_v^{peal} + t_v^{lok}; \quad C_v = C_v^{nod} + C_v^{peal} + C_v^{lok}, \quad (1)$$

где

$$t_v^{nod} = t_v^{gen} + t_v^{zagn} + t_v^{\partial p}; \quad C_v^{nod} = C_v^{gen} + C_v^{zagn} + C_v^{\partial p}; \quad (2)$$

t_v^{gen} и C_v^{gen} – время и стоимость генерации тестовых данных для v -го подграфа разбиения; t_v^{zagn} и C_v^{zagn} – время и стоимость создания необходимых для организации отладки v -го подграфа разбиения подпрограмм «заглушек»; $t_v^{\partial p}$ и $C_v^{\partial p}$ – время и стоимость создания необходимой для проведения отладки v -го подграфа разбиения подпрограммы «драйвера»; t_v^{peal} и C_v^{peal} – время и стоимость реализации тестов для v -го подграфа разбиения; t_v^{lok} и C_v^{lok} – время и стоимость локализации ошибок, обнаруженных в процессе тестирования v -го подграфа разбиения.

Времена и стоимости, участвующие в выражениях (1) и (2), целесообразно определять по суммарным временам и стоимостям тестов, необходимых для проверки v -го подграфа разбиения:

$$\begin{aligned} t_v^{gen} &= \sum_{i=1}^{I_v} \hat{t}_{iv}^{gen}; \quad t_v^{peal} = \sum_{i=1}^{I_v} \hat{t}_{iv}^{peal}; \quad t_v^{lok} = \sum_{i=1}^{I_v} \hat{t}_{iv}^{lok} \rho; \\ C_v^{gen} &= \sum_{i=1}^{I_v} \hat{C}_{iv}^{gen}; \quad C_v^{peal} = \sum_{i=1}^{I_v} \hat{C}_{iv}^{peal}; \quad C_v^{lok} = \sum_{i=1}^{I_v} \hat{C}_{iv}^{lok} \rho. \end{aligned} \quad (3)$$

Здесь ρ – вероятность обнаружения ошибки; I_v – количество тестов, необходимых для проверки v -го подграфа разбиения.

Временные и стоимостные характеристики тестов определяются структурной и вычислительной сложностью соответствующих маршрутов обработки информации, а также квалификацией программиста, его опытом в разработке и отладке программ определенного класса, языком, на котором написаны программные модули комплекса, степенью автоматизации процесса системной отладки и т. п.

А.И. Крапчатов

Структурная сложность определяется длиной маршрута обработки информации (l) в операторах.

Оценка вычислительной сложности обычно проводится на основании выделения нескольких классов вычислительной сложности и отнесения исследуемого комплекса программ к одному из них.

Для комплексов программ различных классов и фиксированного коллектива программистов методами экспертных оценок могут быть получены графические зависимости времени и стоимости генерации тестовых данных от длины маршрута обработки информации: $T^{gen}(l)$, $C^{gen}(l)$.

Время реализации процесса тестирования при отладке комплексов программ определяется средним количеством тестируемых маршрутов обработки информации N_{cp} и загруженностью технических средств, которая может быть охарактеризована коэффициентом K_p . Коэффициент K_p задает среднее количество обращений к ЭВМ. Таким образом, время и стоимость реализации процесса тестирования определяются выражениями:

$$T^{peak} = \left(\frac{1}{K_p} \right) \cdot N_{cp}; C^{peak} = \tilde{C} \cdot N_{cp},$$

где \tilde{C} – стоимость одного обращения к ЭВМ.

Время и стоимость локализации обнаруженных ошибок определяются средним количеством обращений к ЭВМ, необходимых для их локализации, и загруженностью используемых технических средств (коэффициентом K_p). Среднее количество обращений к ЭВМ для локализации обнаруженной ошибки (K_l) определяется структурной сложностью соответствующего маршрута обработки информации, вычислительной сложностью комплекса программ, квалификацией и опытом программиста и т. п.

Для комплексов программ различных классов и фиксированного коллектива программистов методами экспертных оценок могут быть получены графические и табличные зависимости K_l от длины маршрута обработки информации в операторах $K_l(l)$ или зависимости

$$T^{loc}(l) = \frac{K_l}{K_p} \quad \text{и} \quad C^{loc}(l) = K_l(l) \tilde{C}.$$

С учетом зависимостей $T^{gen}(l)$, $T^{peak}(N_{cp})$, $T^{loc}(l)$, $C^{gen}(l)$, $C^{peak}(N_{cp})$, $C^{loc}(l)$ выражения для временных и стоимостных характеристик множества тестов для v -го подграфа будут иметь вид:

$$\begin{aligned} t_v^{e\text{en}} &= \sum_{i=1}^{I_v} T^{e\text{en}}(l_{iv}); \quad t_v^{\text{лок}} = \sum_{i=1}^{I_v} T^{\text{лок}}(l_{iv}) \rho; \\ C_v^{e\text{en}} &= \sum_{i=1}^{I_v} (C^{e\text{en}}(l_{iv})); \quad C_v^{\text{лок}} = \sum_{i=1}^{I_v} C^{\text{лок}}(l_{iv}) \rho \end{aligned} \quad (4)$$

либо

$$\begin{aligned} t_v^{e\text{en}} &= T^{e\text{en}}(l_v^{cp}) N_v^{cp}; \quad t_v^{peal} = T^{peal}(N_v^{cp}); \quad t_v^{\text{лок}} = T^{\text{лок}}(l_v^{cp}) N_v^{cp} \rho; \\ C_v^{e\text{en}} &= C^{e\text{en}}(l_v^{cp}) N_v^{cp}; \quad C_v^{peal} = C^{peal}(N_v^{cp}); \quad C_v^{\text{лок}} = C^{\text{лок}}(l_v^{cp}) N_v^{cp} \rho, \end{aligned}$$

где l_v^{cp} – средняя длина маршрута в v -м подграфе разбиения; N_v^{cp} – среднее количество маршрутов в v -м подграфе разбиения; ρ – вероятность ошибки.

Времена и стоимости создания подпрограмм «заглушек» для v -го подграфа разбиения могут быть получены на основании следующих выражений:

$$t_v^{загл} = \sum_i \sum_j a_{ij} x_{iv} (1 - x_{jv}) \cdot t_j^{загл}; \quad C_v^{загл} = \sum_i \sum_j a_{ij} x_{iv} (1 - x_{ij}) \cdot C_j^{загл}, \quad (5)$$

где

$$x_{iv} = \begin{cases} 1, & \text{если между } i\text{-я вершина включена в } v\text{-й подграф;} \\ 0, & \text{если нет,} \end{cases}$$

$$a_{ij} = \begin{cases} 1, & \text{если между } i\text{-й и } j\text{-й вершинами графа } \Gamma \text{ есть дуга} \\ & \text{связи по управлению;} \\ 0 & \text{в противном случае.} \end{cases}$$

Следует отметить, что в число допустимых разбиений графа Γ целесообразно включать лишь такие разбиения, в которых все вершины имеют связи по управлению, прямо или опосредованно (т. е. осуществляемые через другие программные модули, включенные в v -й подграф разбиения). Следствием такого ограничения является то, что любой v -й подграф разбиения требует создания только одной подпрограммы «драйвера», время и стоимость создания которой определяется следующими выражениями:

$$t_v^{\partial p} = \sum_i \sum_j a_{ij} (1 - x_{iv}) x_{jv} t_i^{\partial p}; \quad C_v^{\partial p} = \sum_i \sum_j a_{ij} (1 - x_{iv}) x_{jv} C_i^{\partial p}. \quad (6)$$

Длины маршрутов обработки информации в подграфах разбиения, средняя длина маршрута и среднее количество маршрутов в подграфах разбиения зависят только от набора программных мо-

А.И. Крапчатов

дулей, включенного в подграф, следовательно, можно записать следующие соотношения:

$$\begin{aligned} t_v^{e\text{en}} &= T^{e\text{en}}(X_v); \quad t_v^{peal} = T^{peal}(X_v); \quad t_v^{\text{лок}} = T^{\text{лок}}(X_v)\rho; \\ C_v^{e\text{en}} &= C^{e\text{en}}(X_v); \quad C_v^{peal} = C^{peal}(X_v); \quad C_v^{\text{лок}} = C^{\text{лок}}(X_v)\rho, \end{aligned} \quad (7)$$

где $X_v = \{x_{rv}\}$, $r = \overline{1, V}$ характеризует состав программных модулей v -го подграфа разбиения.

С учетом соотношений (4)–(6) время и стоимость автономной отладки подграфа разбиения графа Γ будут иметь вид:

$$\begin{aligned} T^{\text{разб}} &= \sum_v (T^{e\text{en}}(X_v) + T^{peal}(X_v) + T^{\text{лок}}(T_v)\rho) + \\ &+ \sum_i \sum_j a_{ij} (x_{iv}(1 - x_{jv}) \cdot t_j^{\text{загл}} + x_{jv}(1 - x_{iv}) \cdot t_i^{\partial p}); \\ C^{\text{разб}} &= \sum_v (C^{e\text{en}}(X_v) + C^{peal}(X_v) + C^{\text{лок}}(T_v)\rho) + \\ &+ \sum_i \sum_j a_{ij} (x_{iv}(1 - x_{jv}) \cdot C_j^{\text{загл}} + x_{jv}(1 - x_{iv}) \cdot C_i^{\partial p}). \end{aligned}$$

При связной отладке полученных подграфов разбиения используются тесты второго типа. Время и стоимость связной отладки на каждом этапе объединения определяются путем выделения на общей схеме комплекса программ частей, соответствующих программным модулям, включенным в этап связной отладки, и определения временных и стоимостных характеристик тестов, необходимых для проверки подграфов этого этапа в совокупности. При этом части общей схемы, соответствующие подграфам, включенным в рассматриваемый этап объединения, заменяются одной вершиной. Кроме того, одной вершиной заменяются совокупности подграфов, которые участвовали в предшествующих этапах связного тестирования.

Время и стоимость связной отладки включают время и стоимость подготовки данных $b_k^{\text{под}}$, $S_k^{\text{под}}$; время и стоимость реализации тестов b_k^{peal} , S_k^{peal} ; время и стоимость локализации обнаружения ошибок $b_k^{\text{лок}}$, $S_k^{\text{лок}}$:

$$b_k = b_k^{\text{под}} + b_k^{\text{peal}} + b_k^{\text{лок}}; \quad S_k = S_k^{\text{под}} + S_k^{\text{peal}} + S_k^{\text{лок}}. \quad (8)$$

Здесь k – индекс этапа объединения.

Времена и стоимости, участвующие в выражении (8), могут быть определены как суммарные временные и стоимостные характеристики тестов, необходимых для проверки структуры k -го этапа объединения:

$$\begin{aligned} b_k^{nod} &= \sum_{j=1}^{J_k} \hat{b}_{jk}^{nod}; \quad b_k^{peal} = \sum_{j=1}^{J_k} \hat{b}_{jk}^{peal}; \quad b_k^{lok} = \sum_{j=1}^{J_k} \hat{b}_{jk}^{lok} \rho; \\ S_k^{nod} &= \sum_{j=1}^{J_k} \hat{S}_{jk}^{nod}; \quad S_k^{peal} = \sum_{j=1}^{J_k} \hat{S}_{jk}^{peal}; \quad S_k^{lok} = \sum_{j=1}^{J_k} \hat{S}_{jk}^{lok} \rho, \end{aligned} \quad (9)$$

где J_k – количество тестов k -го этапа объединения.

Для программ различных классов и фиксированного коллектива программистов методами экспертизы оценок могут быть получены графические и табличные зависимости временных и стоимостных характеристик подготовки данных и локализации ошибок от длины маршрутов, включающих объединенные вершины $B_k^{nod}(l)$, $B_k^{lok}(l)$, $S_k^{nod}(l)$, $S_k^{lok}(l)$.

Для определения времени и стоимости реализации процесса тестирования используются зависимости $B^{peal} = \frac{1}{K_p} N^{cp}$, $S^{peal} = \tilde{C} N^{cp}$.

С учетом приведенных рассуждений выражения для временных и стоимостных характеристик k -го этапа связного тестирования будут иметь вид:

$$\begin{aligned} b_k^{nod} &= \sum_{j=1}^{J_k} B^{nod}(l_{jk}); \quad b_k^{lok} = \sum_{j=1}^{J_k} B^{lok}(l_{jk})\rho; \\ S_k^{nod} &= \sum_{j=1}^{J_k} S^{nod}(l_{jk}); \quad S_k^{lok} = \sum_{j=1}^{J_k} S^{lok}(l_{jk})\rho \end{aligned} \quad (10)$$

либо

$$\begin{aligned} b_k^{nod} &= B^{nod}(l_k^{cp}) N_k^{cp}; \quad b_k^{peal} = B^{peal}(N_k^{cp}); \quad b_k^{lok} = B^{lok}(l_k^{cp}) N_k^{cp} \rho; \\ S_k^{nod} &= S^{nod}(l_k^{cp}) N_k^{cp}; \quad S_k^{peal} = S^{peal}(N_k^{cp}); \quad S_k^{lok} = S^{lok}(l_k^{cp}) N_k^{cp} \rho, \end{aligned}$$

где l_{jk} - длина j -го маршрута обработки информации на k -м этапе связной отладки, l_k^{cp} – средняя длина маршрута на k -м этапе связ-

А.И. Крапчатов

ной отладки, N_k^{cp} – среднее количество маршрутов обработки информации на k -м этапе связной отладки.

Длина маршрута, средняя длина маршрута, среднее количество маршрутов для участков общей схемы комплекса программ, исследуемых на различных этапах связной отладки, зависят от набора подграфов, присоединяемого к отлаживаемой части комплекса, программных модулей, входящих в эти подграфы, набора подграфов, участвующих в предшествующих этапах связной отладки. Поэтому можно записать следующие выражения для временных и стоимостных характеристик k -го этапа связной отладки:

$$\begin{aligned} b_k^{no\partial} &= B^{no\partial}(y_k, Y_{K-1}); b_k^{peal} = B^{peal}(y_k, Y_{K-1}); b_k^{lok} = B^{lok}(y_k, Y_{K-1})\rho; \\ S_k^{no\partial} &= S^{no\partial}(y_k, Y_{K-1}); S_k^{peal} = S^{peal}(y_k, Y_{K-1}); S_k^{lok} = S^{lok}(y_k, Y_{K-1})\rho, \end{aligned} \quad (11)$$

где $y_k = \{y_{kv} X_{rv}\}$, $r = 1, \bar{V}$, $v = 1, V^*$ – переменные, определяющие состав подграфов и программных модулей, участвующих в k -м этапе связной отладки; $Y_{K-1} = y_1, y_2, \dots, y_{k-1}$ – множество переменных, определяющих состав ($k-1$) предшествующих этапов связной отладки.

Таким образом, с использованием выражений (8)–(11) определяются время и стоимость связной отладки комплекса программ:

$$\bar{T}^{объед} = \sum_K [B^{no\partial}(y_k, Y_{K-1}) + B^{peal}(y_k, Y_{K-1}) + B^{lok}(y_k, Y_{K-1})\rho];$$

$$\bar{C}^{объед} = \sum_K [S^{no\partial}(y_k, Y_{K-1}) + S^{peal}(y_k, Y_{K-1}) + S^{lok}(y_k, Y_{K-1})\rho].$$

Задача выбора оптимальной стратегии системной отладки с учетом временного критерия будет иметь вид:

$$\begin{aligned} &\sum_v \left\{ T^{gen}(X_v) + T^{peal}(X_v) + T^{lok}(X_v)\rho + \right. \\ &\left. + \sum_i \sum_j a_{ij} \left[x_{iv} (1 - x_{jv}) \cdot t_j^{загл} + x_{jv} (1 - x_{iv}) \cdot t_i^{оп} \right] \right\} + \\ &+ \sum_k [B^{gen}(y_k, Y_{K-1}) + B^{peal}(y_k, Y_{K-1}) + B^{lok}(y_k, Y_{K-1})\rho] \rightarrow \min \end{aligned} \quad (12)$$

при ограничениях на стоимость организации и проведения системной отладки:

$$\begin{aligned} & \sum_v \left\{ C^{\text{сен}}(X_\nu) + C^{\text{пев}}(X_\nu) + C^{\text{лок}}(X_\nu) \rho + \right. \\ & \left. + \sum_i \sum_j a_{ij} \left[x_{i\nu} (1 - x_{j\nu}) \cdot C_j^{\text{загл}} + x_{j\nu} (1 - x_{i\nu}) \cdot C_j^{\text{оп}} \right] \right\} + \\ & + \sum_k \left[S^{\text{сен}}(y_k, Y_{K-1}) + S^{\text{пев}}(y_k, Y_{K-1}) + S^{\text{лок}}(y_k, Y_{K-1}) \rho \right] \leq C_{\text{зад}} \end{aligned} \quad (13)$$

и ограничениях, задающих возможные варианты объединения и разбиения графа Γ :

$$\sum_\nu x_{r\nu} = 1; r = \overline{1, V}, \quad (14)$$

т. е r -й программный модуль, который должен включаться только в один подграф разбиения графа Γ при автономной отладке;

$$\sum_i \sum_j a_{ij} (1 - x_{i\nu}) \cdot x_{j\nu} \leq 1, \nu = \overline{1, V}, \quad (15)$$

т. е. есть для каждого ν -го подграфа разбиения необходимо создание только одной подпрограммы «драйвера»;

$$y_{jk} \leq a_{ij} x_{\nu k} x_{jf}, i = \overline{1, V}; j = \overline{1, V}; \nu = \overline{1, V}; f = \overline{1, V}; k = \overline{1, f}; \quad (16)$$

т. е. в k -й этап связной отладки могут включаться только подграфы, имеющие связи по управлению;

$$y_{k+1,\nu} \geq y_{k,\nu}; \nu = \overline{1, V}; k = \overline{2, V}; \quad (17)$$

(если ν -й подграф включен в k -й этап связной отладки, он включается во все последующие этапы связной отладки);

$$y_{k,\nu} \leq \sum_r x_{r\nu}; \nu = \overline{1, V}; k = \overline{1, V}; \quad (18)$$

т. е. в ν -й подграф может включаться в k -й этап связной отладки тогда и только тогда, когда множество его вершин непустое;

$$\sum_r x_{r\nu} \left(\sum_k y_{k\nu} \right) \geq \sum_r x_{r\nu}; r = \overline{1, V}; \quad (19)$$

(если ν -й подграф непустой, он должен быть включен хотя бы в один этап связной отладки).

Задача выбора оптимальной стратегии системной отладки с учетом стоимостного критерия будет иметь вид:

А.И. Крапчатов

$$\begin{aligned} \sum_v \left\{ C^{zen}(X_v) + C^{peal}(X_v) + C^{lok}(X_v) \rho + \right. \\ \left. + \sum_i \sum_j a_{ij} \left[x_{iv} (1 - x_{jv}) \cdot C_j^{zaen} + x_{jv} (1 - x_{iv}) \cdot C_i^{op} \right] \right\} + \\ + \sum_k \left[S^{zen}(y_k, Y_{K-1}) + S^{peal}(y_k, Y_{K-1}) + S^{lok}(y_k, Y_{K-1}) \rho \right] \rightarrow \min \end{aligned}$$

при ограничении на общее время проведения системной отладки

$$\begin{aligned} \sum_v \left\{ T^{zen}(X_v) + T^{peal}(X_v) + T^{lok}(X_v) \rho + \right. \\ \left. + \sum_i \sum_j a_{ij} \left[x_{iv} (1 - x_{jv}) \cdot t_j^{zaen} + x_{jv} (1 - x_{iv}) \cdot t_i^{op} \right] \right\} + \\ + \sum_k \left[S^{zen}(y_k, Y_{K-1}) + S^{peal}(y_k, Y_{K-1}) + S^{lok}(y_k, Y_{K-1}) \rho \right] \leq T_{za\partial}, \end{aligned}$$

где $T_{za\partial}$ – допустимое время на проведение системной отладки, и ограничениях типа (14) – (19).

Решение задачи выбора оптимальной стратегии системной отладки в общем виде представляет значительные трудности, так как даже для участков общей схемы с фиксированной структурой определение средних длин (ℓ^p) и среднего количества маршрутов обработки информации (N^p), определяющих временные и стоимостные характеристики системной отладки, могут быть получены только алгоритмическим путем. Поэтому в ряде практических случаев необходимо рассматривать постановки и решения частных задач выбора оптимальной стратегии системной отладки, имеющих более простую структуру и меньшую вычислительную сложность.

Примечания

¹ Липаев В.В. Обеспечение качества программных средств. М.: СИНТЕГ, 2001.
Достоверность, защита и резервирование информации в АСУ / А.Г. Мамиканов,
В.В. Кульба, А.Б. Шелков. М.: Энергоатомиздат, 1986.