

Задача контроля данных системы управления работой космического аппарата с охватом проблемной области

Шихин С.М.

*Научно-производственное объединение им. С.А. Лавочкина,
ул. Ленинградская, 24, Химки, Московская область, 141411, Россия
e-mail:wooling@rambler.ru*

Статья поступила 07.11.2019

Аннотация

Опыт создания и эксплуатации сложных искусственных систем позволяет говорить о несовершенстве современных инструментальных программных средств, широко применяемых в таких системах, в том числе, для решения задач контроля данных системы планирования и управления работой аппаратуры космического аппарата, управления ее жизненным циклом. Рассматриваемая в данной статье проблемная область содержит большое число составляющих, их свойств и связей, образующих сложные структуры. Подлежащие обработке и контролю данные имеют сложную организацию и большой объем. Кроме того, проблемная область изменчива во времени. Современные инструментальные программные средства, позволяющие разработчикам программного обеспечения допускать семантические ошибки, не соответствуют сложности проблемной области.

На практике, их применение обеспечивает лишь фрагментарное покрытие проблемной области. Отсюда – трудности контроля данных системы планирования

и управления, трудности управления ее жизненным циклом. Важнейшие для практического применения программно-технических средств проблемы верификации и безотказности, в общем случае, продолжают оставаться нерешенными.

Ключевые слова: автоматизированная система, космический аппарат, система планирования и управления. логика предикатов, логическое программирование, программно-технические средства, стандартизация, интеллектуализация.

Введение

Состояние современных инструментальных программных средств и объективные трудности обеспечения контроля данных системы [1] планирования и управления работой аппаратуры космического аппарата, управления ее жизненным циклом [2] (СПУ КА), связанные со сложностью проблемной области [3], требуют совершенствования подходов к решению соответствующих задач. Проблема верификации [4] программно-технических средств (ПТС) и их безотказности, в общем случае, не решена. Представляется, что причина этого – отсутствие при моделировании полноты охвата всей широкой проблемной области, включающей как прикладные задачи, так и соответствующие ПТС. Таким образом, необходим полный охват проблемной области обеспечения контроля данных, обеспечение

управления ею на основе значимых данных и знаний о всей рассматриваемой проблемной области [5].

По этой причине, СПУ КА необходимо рассматривать как сложную целенаправленную автоматизированную систему (АС), имеющую такие виды обеспечения, как математическое, программное и другие. Создаваться такие виды обеспечения должны как совокупность связанных между собой компонентов [6]. Контроль данных должен осуществляться на всем жизненном цикле системы планирования и управления работой аппаратуры космического аппарата с помощью модификаций компонентов обеспечений разных видов.

Такой подход требует единообразной формализации всей сложной, разнородной, но внутренне единой проблемной области, в рассматриваемом случае – применительно к задаче контроля данных СПУ КА и управления его жизненным циклом. Для обеспечения такой формализации [7] предлагается радикальное моделирование и универсальный язык [8] схем радикалов RADICAL.

Таким образом, цель работы - краткое описание подхода к обеспечению, на основе радикального моделирования, контроля данных системы планирования и управления работой аппаратуры космического аппарата. В целом, предлагаемый подход направлен на широкое применение программно-технических средств, на стандартизацию проблемной области СПУ КА и других сложных АС, на их интеллектуализацию – умение решать нештатные задачи и самообучаться на этой основе.

1. Проблемная область контроля данных системы планирования и управления работой аппаратуры космического аппарата

Опыт создания и эксплуатации сложных искусственных систем позволяет утверждать о несовершенстве современных инструментальных программных средств (ПС), широко применяемых в таких системах, в том числе, для решения задач контроля данных. Рассматриваемая проблемная область содержит большое число составляющих, их свойств и связей. (Это должны быть все значимые составляющие, все их значимые свойства и связи, причем с избытком – на случай нештатных ситуаций.) Подлежащие обработке и контролю данные имеют сложную организацию и большой объем. Проблемная область включает как прикладные задачи [9], так и ПТС, необходимые для их решения. Кроме того, проблемная область изменчива во времени.

Современные инструментальные ПС не соответствуют такой сложности. На практике, их применение обеспечивает лишь фрагментарное покрытие проблемной области. Отсюда – трудности контроля данных СПУ, трудности управления ее жизненным циклом. Важнейшие для практического применения ПТС проблемы верификации и безотказности, в общем случае, продолжают оставаться нерешенными. Все это говорит о необходимости [10] совершенствования подходов, в том числе, к решению задач контроля данных СПУ КА и управления ее жизненным циклом. Требуется полный охват проблемной области обеспечения

контроля данных, обеспечение управления проблемной областью на основе всех значимых знаний и данных. Требуется единообразные формальные описания проблемной области, реализуемые с помощью ПТС.

СПУ КА необходимо рассматривать как сложную целенаправленную автоматизированную систему (АС).

Согласованные между собой модификации компонентов обеспечений разных видов реализуют жизненный цикл СПУ КА от постановки целевой задачи [10] до утилизации. Для осуществления такого подхода необходимо обеспечить единообразную формализацию всей объективно единой проблемной области.

2. Радикальное моделирование, язык RADICAL – основа формализации, автоматизации и контроля данных

В качестве основного средства единообразной формализации проблемной области предлагается использовать радикальное моделирование и универсальный язык схем радикалов RADICAL [2]. Радикальное моделирование основано, в том числе, на логике предикатов и логическом программировании.

Центральным понятием радикального моделирования является понятие радикала. Радикал – это функциональная система, имеющая два типа состояний – активные и пассивные. Активный радикал функционирует согласно своему назначению, а пассивный радикал – нет, он до востребования выключен и находится в резерве. Множество связанных между собой радикалов образуют среду радикалов.

Среда радикалов представляется с помощью схемы радикалов – конструкции языка RADICAL. Компоненты обеспечений разных видов – составляющие среды радикалов – также представляются схемами радикалов, подлежащими реализации с помощью ПТС. (Отметим, что примеры схем радикалов, рассматриваемые в данной статье, имеют упрощенный, демонстрационный характер). В качестве основы СПУ КА используем структуру верхнего уровня (схему радикалов) типовой автоматизированной подсистемы управления [4], см. рис.1.

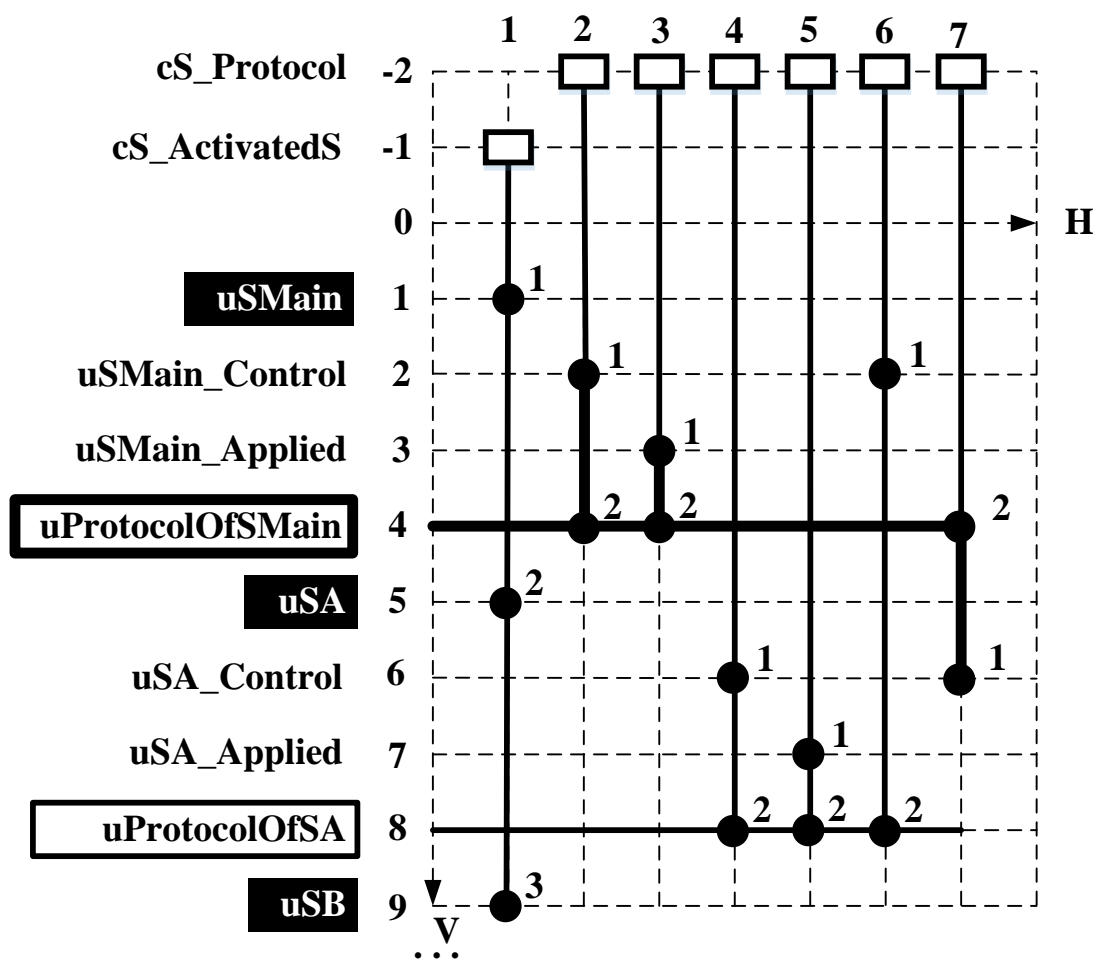


Рис. 1. Структура верхнего уровня типовой автоматизированной подсистемы управления – основы СПУ КА

Для представления структуры используется схема радикалов в геометрическом представлении. На плоскости введена прямоугольная система координат с горизонтальной осью H , направленной вправо и вертикальной осью V , направленной вниз. Символом 'u' начинаются имена радикалов-уникумов, соответствующих объектам, составляющим проблемной области. Символ 'с' – первый символ имен радикалов-контейнеров, связывающих радикалы-уникумы, а также, в общем случае, другие контейнеры.

Горизонтالي $V=1, \dots, 9$ соответствуют уникамам. В случае появления новых уникамов, соответствующие им новые горизонтали $V=10, 11, \dots$ будут добавляться в нижнюю часть плоскости. Горизонтали $V=-1, -2, \dots$ - это горизонтали типов используемых контейнеров. Новые уникамы и новые горизонтали будут добавляться в верхнюю часть плоскости. Такое «устройство» оси ординат можно назвать естественным, поскольку, при реализации схемы радикалов с помощью таблицы баз данных (поле V – поле первичного ключа, столбцы соответствуют экземплярам контейнеров), строки – горизонтали будут автоматически занимать нужное положение [11].

Прямоугольники с белыми символами на черном фоне соответствуют уникамам-подсистемам. Прямоугольники с черными символами на белом фоне и черной рамкой соответствуют уникамами-протоколам. Уникум-протокол главной подсистемы имеет широкую рамку. Прямоугольники, расположенные на оси H ,

соответствуют экземплярам контейнеров. Для горизонталей и вертикалей уникама-протокола главной подсистемы использованы более толстые линии.

Рассматриваемая структура [12] содержит следующие радикалы-подсистемы: uSMain (H=0, V=1) – уникам-«главная» подсистема; uSA, uSB,_(H=0, V=5, 9)... - уникамы-подсистемы, которые ориентированы на решение задач того или иного класса. В состав каждой подсистемы входят: подсистема управления; прикладные подсистемы и протоколы. Взаимодействие подсистем АС осуществляется с помощью протоколов, содержащих уникамы-сообщения и характеризующие их контейнеры (эти уникамы и контейнеры в рассматриваемом сечении среды радикалов не показаны). Подсистемы используют сообщения в своей работе, динамически связывая составляющие АС в единое целое – единую АС. Основное назначение подсистемы uSMain – анализ ситуации, сложившейся в проблемной области. Далее, по результатам такого анализа должна быть сформирована схема радикалов - постановка задачи, выбрана (построена) и активирована схема-метод ее решения. Подсистема uSMain активирует подсистемы, ориентированные на решение задач того или иного класса (запускает соответствующие программные модули), а также контролирует состояние информационных ресурсов.

Рассмотрим вертикали схемы, представленной на рис. 1 (вертикали соответствуют экземплярам контейнеров).

Вертикаль H=1 – вертикаль экземпляра контейнера cSActivated_S («S» от «System») – контейнера активирующей подсистемы uSMain (V=1, направление 1 в

контейнере) и активируемых ею подсистем uSA (V=5, направление 2 в контейнере)

и uSB (V=9, направление 3 в контейнере).

Вертикали H=2,..., 7 соответствуют экземплярам контейнера cS_Protocol – контейнера подсистемы и используемого ею протокола.

Вертикаль H=2. Здесь объединены подсистема управления uSMain_Control (V=2, направление 1 в контейнере) подсистемы uSMain и протокол uProtocolOfSMain (V=4, направление 2 в контейнере), с которым эта подсистема управления работает.

Вертикаль H=3 – представляет объединение прикладных подсистем uSMain_Applied (V=3, направление 1 в контейнере) подсистемы uSMain и доступного этим подсистемами протокола uProtocolOfSMain (V=4, направление 2 в контейнере) подсистемы uSMain.

Вертикаль H=4. Здесь доступны принадлежащие подсистеме uSA подсистема управления uSA_Control (V=6, направление 1 в контейнере) и протокол uProtocolOfSA (V=9, направление 2 в контейнере), с которым эта подсистема управления работает.

Вертикаль H=5 объединяет принадлежащие подсистеме uSA прикладные подсистемы uSA_Applied (направление 1 в контейнере) и протокол uProtocolOfSA (направление 2 в контейнере), с которым эти прикладные подсистемы работают.

Отметим, что прикладные подсистемы uSA_Applied (V=7, H=0) и другие прикладные подсистемы работают также с соответствующими локальными

протоколами прикладных[13] подсистем, которые в данном сечении[14] среды радикалов не показаны.

Вертикаль Н=6. В этом экземпляре контейнера cS_Protocol доступна подсистема управления uSMain_Control подсистемы uSMain (V=2, направление 1 в контейнере), которая работает с протоколом uProtocolOfSA подсистемы uSA (V=8, направление 2 в контейнере).

Вертикаль Н=7. Последняя вертикаль рассматриваемого сечения среды радикалов. Экземпляр контейнера cS_Protocol, соответствующий этой вертикали, объединяет подсистему управления uSA_Control подсистемы uSA (V=6, направление 1 в контейнере) и доступный ей протокол uProtocolOfSMain подсистемы uSMain (V=4, направление 2 в контейнере).

Для уникумов uSB, имеют место аналогичные описания.

3. Схемы радикалов, характерные для проблемной области контроля данных

Перейдем к рассмотрению некоторых схем радикалов, характерных для проблемной области контроля данных СПУ КА. В задаче контроля данных СПУ КА выделим следующие подзадачи.

1. Задача контроля внешних исходных данных СПУ КА.

Подлежащие контролю внешние исходные данные могут находиться в разных источниках, в таблицах баз данных, файлах данных. Такие данные могут быть по-

разному организованы. Состав данных, правила их проверки могут зависеть от режима работы СПУ КА, могут изменяться в течение ее жизненного цикла. Все это должно быть учтено в программном и информационном обеспечении.

2. Задача контроля внутренних исходных данных, необходимых для начала работы внутренней подсистемы СПУ КА. Эти данные могут принадлежать как исходным данным, так и быть полученными в результате работы других подсистем СПУ КА. Как и в случае задачи (1), данные могут находиться в разных источниках, могут быть по-разному организованы, их состав и правила проверки могут изменяться.

3. Задача контроля выходных данных СПУ КА.

Выходные данные [15] могут принадлежать как исходным данным СПУ КА, так и являться результатом работы ее подсистем. Требования к размещению выходных данных в «приемниках» данных и их организации являются внешними по отношению к СПУ КА. Состав данных и правила их проверки могут изменяться. Рассмотрим верхний уровень структуры, обеспечивающей контроль данных СПУ КА (см. сечение среды радикалов, представленное на рис. 2), основанной на структуре типовой автоматизированной подсистемы управления (см. рис. 1).

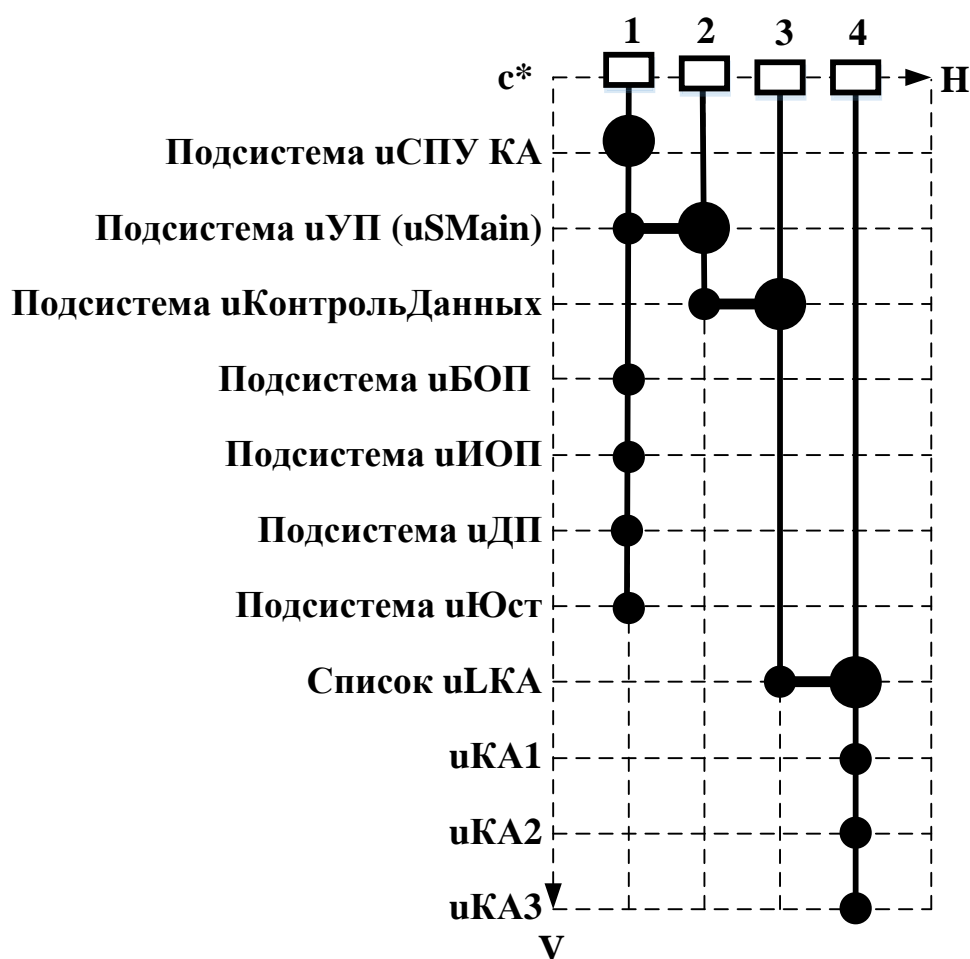


Рис. 2. Верхний уровень структуры обеспечения контроля данных СПУ КА

(Здесь и далее черные круги большего диаметра соответствуют доминирующим в данном экземпляре контейнера уникамам, круги меньшего диаметра - доминируемым уникамам). Рассмотрим вертикали представленного сечения (здесь c^* означает «некоторый контейнер»).

Вертикаль $H=1$ – вертикаль контейнера подсистемы uСПУ КА и входящих в нее подсистем:

uУП (uSMain) – управляющая подсистема;

uБОП - подсистема баллистического обеспечения планирования;

uОП – подсистема оперативного планирования;

uДП – подсистема долгосрочного планирования;

uЮст – подсистема юстировки.

Вертикаль Н=2. Контейнер этой вертикали объединяет управляющую подсистему uУП (uSMain) и входящую в нее подсистему контроля данных uКонтрольДанных.

Вертикаль Н=3. Контейнер этой вертикали говорит о том, что подсистема uКонтрольДанных дает доступ к списку КА uLKA.

Вертикаль Н=4, последняя вертикаль рассматриваемого сечения, в контейнер которой вложены уникам списка КА uLKA, и уникамы-КА uKA1, uKA2 и uKA3, образующие этот список.

Работа по контролю данных осуществляется с помощью протоколов, содержащих генерируемые и анализируемые по совершению тех или иных событий уникамы-сообщения. В рассматриваемом сечении протоколы не показаны. В подсистеме контроля данных доступны процедуры, обеспечивающие контроль [5]. Некоторые из них принадлежат сечению среды радикалов, представленному на рис.3.



Рис. 3. Процедуры обеспечения контроля данных

Рассмотрим вертикали, доступные в представленном сечении.

Вертикаль $H=1$. Контейнер этой вертикали объединяет подсистему контроля данных `uКонтрольДанных` и доступные в ней процедуры (`uВыборКА`, `uВыборПодсистемы` и `uПроверкаИсхДанных`).

Вертикаль $H=2$. Контейнером этой вертикали объединены процедура `uПроверкаИсхДанных` и список доступных ей протоколов `uLProtocols`.

Рассмотрим пример входных данных, структура которых представлена в сечении среды радикалов рис. 4.

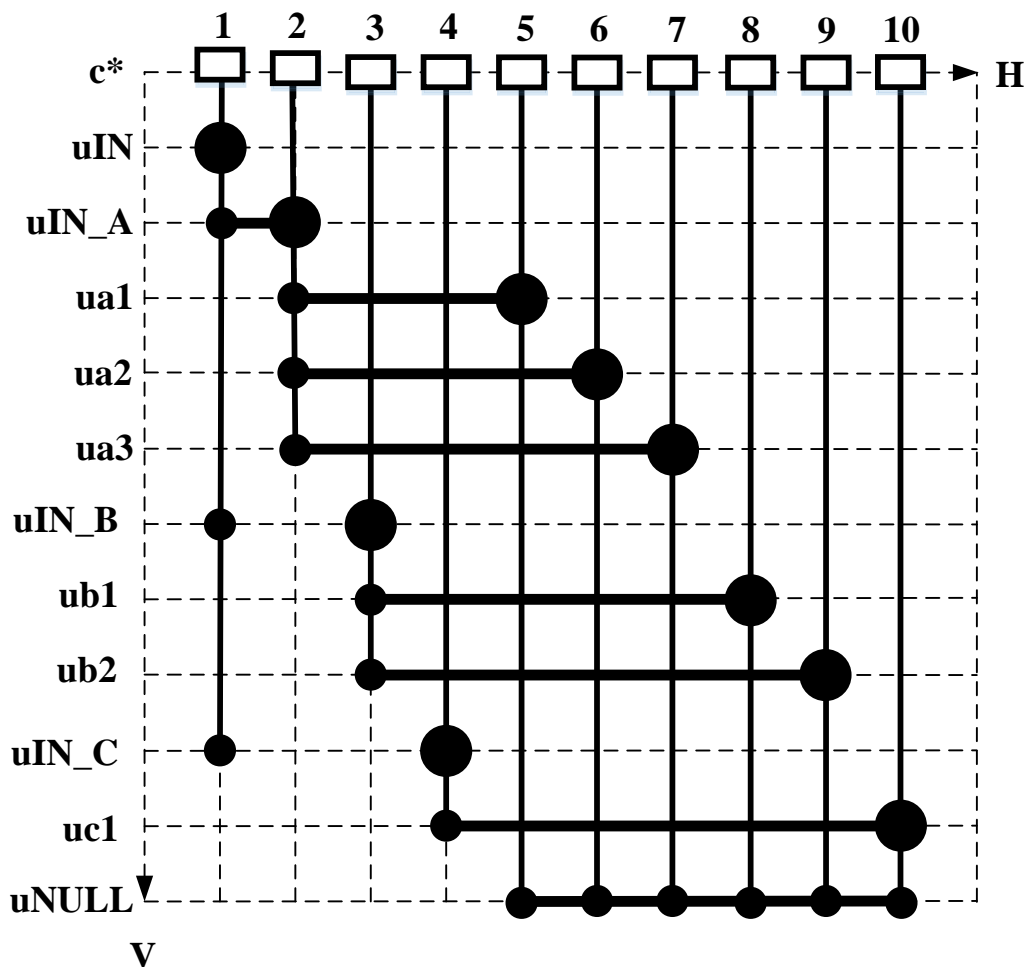


Рис. 4. Структура входных данных

Вертикаль $H=1$. Контейнер этой вертикали объединяет уникам uIN , именуемый входные данные и вектора входных данных uIN_A , uIN_B и uIN_C .

Вертикаль $H=2$. Контейнером этой вертикали объединены вектор uIN_A и его компоненты - переменные $ua1$, $ua2$ и $ua3$, допустимые значения которых – положительные целые числа.

Вертикаль $H=3$ – это вертикаль контейнера, в который вложен вектор uIN_B со своими компонентами-переменными $ub1, ub2$, допустимые значения которых также положительные целые числа.

Вертикаль $H=4$ «устроена» аналогично вертикалям $H=2, 3$ и объединяет вектор uIN_C и его единственную компоненту – переменную (положительное целое число) $uc1$.

Вертикали $H=5, \dots, 10$ показывают (с помощью соответствующих контейнеров), что значения переменных $ua1, ua2$ и $ua3, ub1, ub2, uc1$ вначале не определены. (Для представления неопределенных значений используется неопределенный радикал $uNULL$). Пусть все переменные-компоненты входных векторов приняли конкретные значения, которые необходимо оценить для того, чтобы принять или отвергнуть входные данные (см. сечение среды радикалов на рис. 5). Пусть для оценки используется правило, согласно которому весь вход должен быть отвергнут, если хотя бы одна из переменных имеет недопустимое значение.

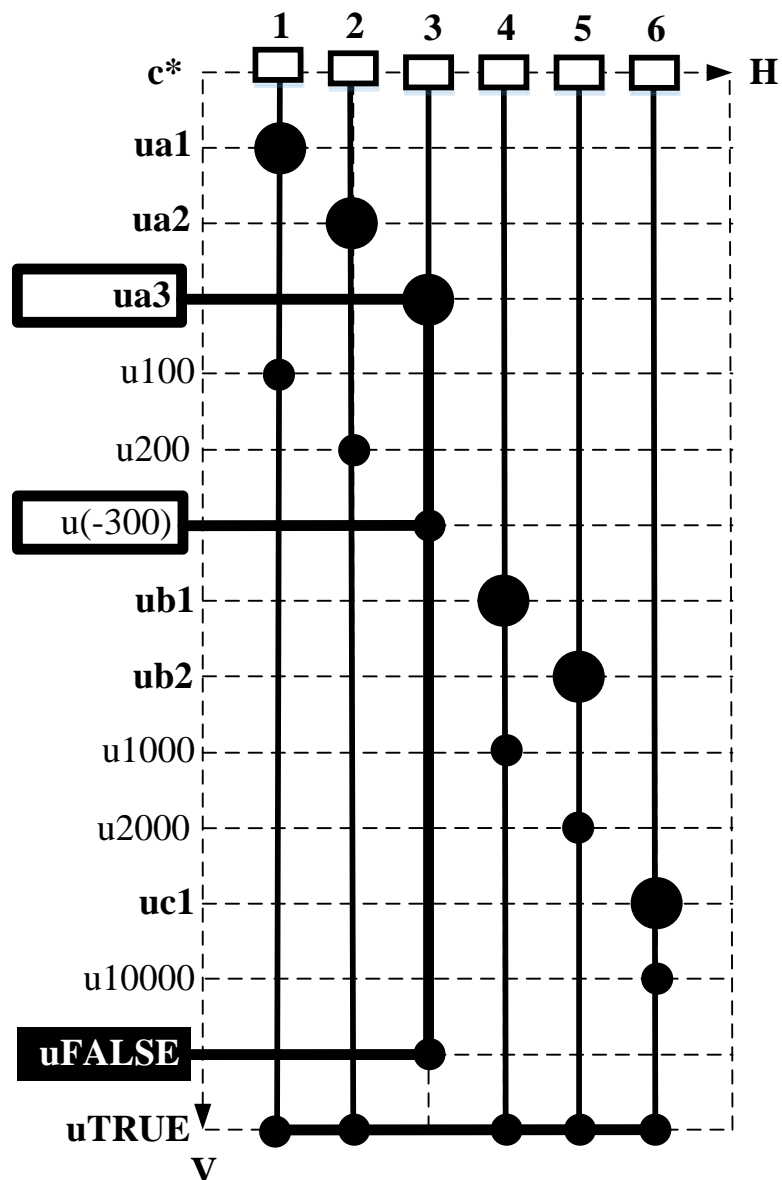


Рис. 5. Оценка значений переменных – компонентов
входных векторов

Рассмотрим вертикали представленного сечения среды радикалов.

Вертикаль $H=1$. Контейнер этой вертикали объединяет переменную $ua1$ (компонент вектора uIN_A), ее значение $u100$ и оценку этого факта с помощью уникама $uTRUE$ (т.е. значение $u100$ для этой переменной – допустимое).

Вертикаль Н=2. Контейнером этой вертикали объединены переменная ua_2 , ее значение $u100$ и оценка $uTRUE$.

Вертикаль Н=3. Согласно принятому правилу работы со входными переменными, эта вертикаль оказывается критичной - переменная ua_3 имеет недопустимое отрицательное значение $u(-300)$.

Имеет место оценка $uFALSE$.

Вход - отвергается.

Тем не менее, проверка (в данном случае) будет продолжена в целях поиска других возможных ошибок [6].

Вертикаль Н=4. Контейнер этой вертикали объединяет переменную ub_1 (компонент вектора uIN_B), ее значение $u1000$ и оценку $uTRUE$.

Вертикаль Н=5. «Устройство» этой вертикали аналогично «устройству» вертикали Н=4.

Контейнер вертикали Н=5 объединяет переменную ub_1 (компонент вектора uIN_B), ее значение $u2000$ и оценку $uTRUE$.

Вертикаль Н=6. Контейнером этой вертикали объединены переменную uc_1 (единственный компонент вектора uIN_C), ее значение $u10000$ и оценку $uTRUE$. Таким образом, ошибок входных данных [16] больше не найдено. Общую картину состояния входа дает сечение среды радикалов, показанное на рис. 6, 7.

На рис. 6 показано состояние входа до проверки на наличие ошибок.

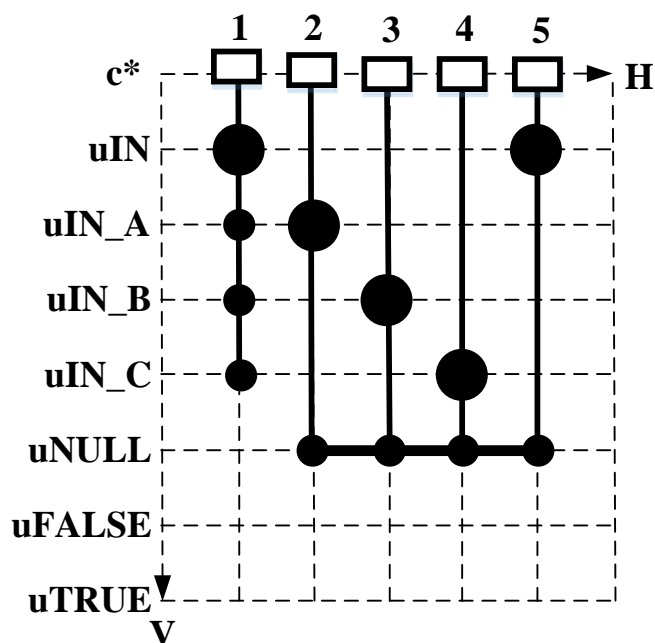


Рис. 6. Общее состояние входа до проверки на наличие ошибок

Вход uIN состоит из трех векторов uIN_A , uIN_B и uIN_C (см. вертикаль $H=1$). Проверка пока не проводилась, поэтому оценка векторов uIN_A , uIN_B и uIN_C , а также входа uIN в целом (вертикали $2, \dots, 5$) – неопределенный радикал $uFALSE$. На рис. 7 показано состояние входа после проверки на наличие ошибок.

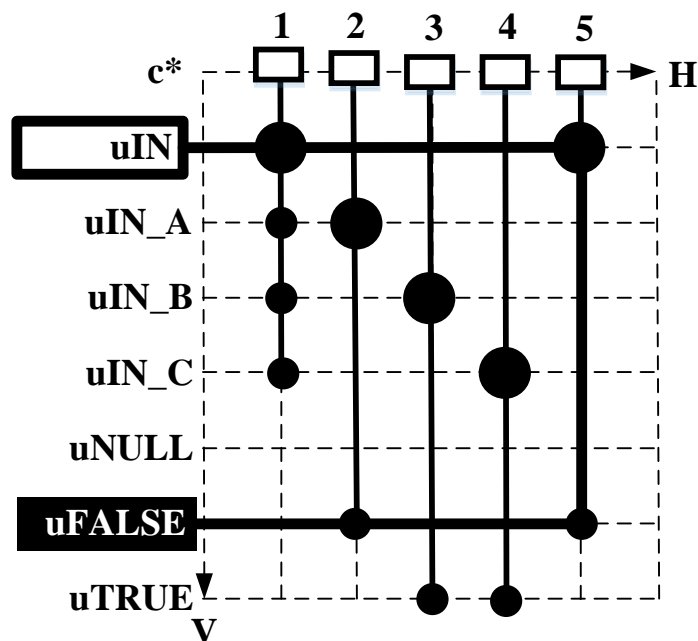


Рис. 7. Общее состояние входа после проверки
на наличие ошибок

Вход uIN состоит из трех векторов uIN_A , uIN_B и uIN_C (вертикаль $H=1$).

Вход uIN оценен как $uFALSE$ (см. вертикаль $H=5$) из-за оценки $uFALSE$ для вектора uIN_A (вертикаль $H=2$), которая вызвана оценкой $uFALSE$ для компонента этого вектора – переменной $u\alpha_3$, принявшей недопустимое значение $u(-300)$ (см. рис. 5). Вектора uIN_B и uIN_C оценены как $uTRUE$ (вертикали $H=3, 4$), что, однако, не спасает положения и вход uIN отвергается. Для схем радикалов, в целях наглядности, могут использоваться не только геометрические, но и другие представления[18].

На рис. 8 приведен пример сечения среды радикалов в табличном представлении.

A	B	C
ua1	100	uTRUE
ua2	200	uTRUE
ua3	-300	uFALSE
ub1	1000	uTRUE
ub2	2000	uTRUE
uc1	10000	uTRUE

Рис. 8. Сечение среды радикалов в табличном представлении

Столбец А таблицы содержит переменные ua_1, \dots, uc_1 – компоненты входных векторов uIN_A , uIN_B и uIN_C .

Ячейки **столбца В** содержат значения переменных ua_1, \dots, uc_1 (символ ‘u’ для уникамов-значений в этом представлении опущен, т.е. вместо $u100$ соответствующая ячейка содержит 100 и т.п.)

Столбец С содержит уникамы-оценки $uFALSE$ и $uTRUE$ переменных, принявших определенные значения. Строка ($ua_3 -300; uFALSE$) выделена. Здесь оценка $uFALSE$ привела к тому, что весь вход в целом был отвергнут. Представляется, что контроль данных сложной системы [19] должен обеспечиваться

[20] средствами специально предназначенной для этого подсистемы. Отметим, что в течение жизненного цикла сложной системы, состав и структуры подлежащих контролю данных, а также правил их проверки могут, по тем или иным причинам, изменяться [21]. Таким образом, должна быть обеспечена контролируемая модификация как информационного обеспечения [22], так и программного обеспечения (процедур проверки).

Заключение

Предлагаемый подход к обеспечению контроля данных системы планирования и управления работой аппаратуры космического аппарата, основанный на применении радикального моделирования, направлен на широкое и планомерное применение программно-технических средств, на единообразную формализацию и стандартизацию рассматриваемой проблемной области и проблемных областей других сложных автоматизированных систем[23], а также, в перспективе, на их интеллектуализацию (на основе единообразной формализации и стандартизации). Представляется, что систематическое применение радикального моделирования может существенно улучшить качество решения задач жизненного цикла системы планирования[20] и управления работой аппаратуры космического аппарата на основе формализации всех значимых аспектов проблемной области и реализации соответствующих моделей с помощью программно-технических средств.

Библиографический список

1. Информационная технология. Комплекс стандартов на автоматизированные системы. Автоматизированные системы. Термины и определения. ГОСТ 34.003.90. - М.: Стандартиформ, 1992.
2. Баринов К.Н., Бурдаев М.Н., Мамон П.А. Динамика и принципы построения орбитальных систем космических аппаратов. – М.: Машиностроение, 1975. - 232 с.
3. Хартов В.В., Ефанов В.В., Занин К.А. Основы проектирования орбитальных оптико-электронных комплексов. - М: Изд-во МАИ, 2011. – 127 с.
4. Чечкин А.В., Пирогов М.В. Необходимость радикальной стандартизации в формализме радикального моделирования и радикального программирования целенаправленных автоматизированных систем // Нейрокомпьютеры: разработка, применение. 2018. № 8. С. 3 - 19.
5. Верба В.С., Неронский Л.Б., Осипов И.Г., Турук В.Э. Радиолокационные системы землеобзора космического базирования. – М.: Радиотехника, 2010. - 675 с.
6. Кодратенков Г.С. Радиолокационные системы дистанционного зондирования Земли. - М.: Радиотехника, 2005. - 368 с.
7. Бусленко Н.П., Шрейдер Ю.А. Метод статистических испытаний (Монте-Карло) и его реализация на цифровых вычислительных машинах. – М.: Физматлит, 1961. - 228 с.

8. Кристофидес Н. Теория графов. Алгоритмический подход. - М.: Мир, 1978. – 432

с.

9. Бусурин В.И., Медведев В.М., Карабицкий А.С., Гроппа Д.В. Алгоритмы анализа цифровой информации для оптимизации контроля систем управления // Труды МАИ. 2017. № 97. URL: <http://trudymai.ru/published.php?ID=87277>

10. Поляков В.Б., Неретин Е.С., Иванов А.С., Будков А.С., Дяченко С.А., Дудкин С.О. Архитектура перспективных комплексов управления бортовым оборудованием // Труды МАИ. 2018. № 100. URL: <http://trudymai.ru/published.php?ID=93459>

11. Бузуев К.В. Определение оптимальных планов наблюдения космического аппарата дистанционного зондирования земли с помощью графа // Вестник Самарского государственного аэрокосмического университета. 2013. № 1 (39). С. 63 – 72.

12. Бусленко Н.П., Шрейдер Ю.А. Метод статистических испытаний и его реализация на цифровых вычислительных машинах. – М.: Физико-математическая литература, 1961. – 228 с.

13. Маслов А.Е., Пирогов М.В., Рожков В.В., Шихин С.М. Разработка информационного обеспечения системы планирования и управления работой целевой аппаратуры космического аппарата на основе радикального моделирования // Нейрокомпьютеры: разработка, применение. 2018. № 4. С. 50 – 61 с.

14. Васенин В.А., Пирогов М.В., Чечкин А.В. Информационно-системная безопасность критических систем. - М.: Изд-во Курс, 2018. - 352 с.

15. Иванов Н.М., Лысенко Л.Н. Баллистика и навигация космических аппаратов. - М.: Радиотехника, 2016. – 523 с.
16. Серапинас Б.Б. Глобальные системы позиционирования. - М.: Каталог, 2002. - 106 с.
17. Федоровский А.Д., Даргейко Л.Ф., Зубко В.П., Якимчук В.Г. Системный подход к оценке эффективности аппаратурных комплексов дистанционного зондирования Земли // Космическая наука и технология. 2001. Т. 7. № 5–6. С. 75 – 79.
18. Гарбук С.В., Гершензон В.Е. Космические системы дистанционного зондирования Земли. – М.: Изд-во А и Б, 1997. - 296 с.
19. Занин К.А. Москатыньев И.В. Рациональное баллистическое построение космической системы радиолокационного наблюдения // Полет. 2018. № 9. С. 23 – 29.
20. Занин К.А. Москатыньев И.В. Метод оценки пространственного и фазового разрешения космического радиолокатора с синтезированной апертурой // Космонавтика и ракетостроение. 2018. № 3(102). С. 53 – 64.
21. Вернигора Л.В., Казмерчук П.В. Оптимизация траекторий КА с малой тягой методом линеаризации // Труды МАИ. 2019. № 106. URL: <http://trudymai.ru/published.php?ID=105759>
22. Васильев В.А., Федюнин П.А., Данилин М.А., Васильев А.В. Проблемные вопросы организации информационного обеспечения управления ударными

Труды МАИ. Выпуск № 109

DOI: 10.34759/trd-2019-109-28

авиационными комплексами // Труды МАИ. 2019. № 105. URL:

<http://trudymai.ru/published.php?ID=104241>

23. Гусев С.Н., Сахно И.В., Хуббиев Р.В. Методика оценивания качества формирования виртуальных объектов на радиолокационных изображениях // Труды МАИ. 2019. № 104. URL: <http://trudymai.ru/published.php?ID=102169>