

КОМПЬЮТЕРНАЯ ИНЖЕНЕРИЯ И ТЕХНИЧЕСКАЯ ДИАГНОСТИКА



УДК 681.3.001.57

ПРОЦЕССНЫЙ МЕТОД ОПИСАНИЯ ЭЛЕКТРОННЫХ СИСТЕМ

ГОРБАЧЕВ В.А., БАБАЕВ А.П., ВОЛК М.А.

В статье рассмотрены средства описания электронных систем (ЭС) на архитектурном уровне, где наиболее эффективной признана алгоритмическая модель на основе параллельных процессов. ЭС рассмотрено с точки зрения теории систем и объяснена сущность процессного метода построения имитационных моделей.

Моделирование занимает важное место в современных САПР электронных устройств. Системы моделирования используются на всех стадиях разработки, начиная с эскизного проектирования и заканчивая автоматическим синтезом и верификацией проекта. Поэтому моделирование охватывает несколько уровней представления ЭС – функционально-логический, архитектурный и системный. В связи с этим при разработке системы моделирования возникает проблема выбора способа описания объекта проектирования. При описании системы на функционально-логическом и системном уровнях применяются соответственно системы логических уравнений и средства теории массового обслуживания. Эти способы описания хорошо изучены и формализованы. Для задания объекта проектирования на архитектурном уровне используется имитационное моделирование. Оно предоставляет разработчику модели богатый набор средств описания объекта, с использованием которых модель может быть построена разными методами. Метод описания объекта моделирования при этом определяет, каким образом известная исследователю информация о структуре и функциях изучаемой системы отражается в имитационном алгоритме. От выбранного метода описания в значительной мере зависит эффективность всего процесса моделирования. Объектом исследования являются методы поведенческого описания моделируемой системы.

Для того чтобы понять, каким образом составляются поведенческие описания ЭС на архитектурном уровне, рассмотрим её с точки зрения общей теории систем [1].

Пусть задано семейство непустых множеств $\bar{V} = \{V_i : i \in I\}$, где I – множество индексов. Множество V_i будем называть объектом системы. Задано

также отношение на этом множестве $S \subset \times_{i \in I} V_i$.

Тогда пару $D = (\bar{V}, S)$ назовем миром объектов.

Системой $m_{\bar{V}_1}^D$ в мире объектов D называется пара $(\bar{V}_1, S_1)$, где $\bar{V}_1 \subset \bar{V}$, а S_1 – сужение S на множество $\bar{V}_1$. Окружающей (внешней) средой системы $m_{\bar{V}_1}^D$ называется пара $(\bar{V} \setminus \bar{V}_1, S_2)$, где S_2 – сужение S на $\bar{V} \setminus \bar{V}_1$. Связями между системой и окружающей средой называется множество $S \setminus (S_1 \cup S_2)$.

Данные определения отражают интуитивное понимание системы как совокупности некоторых объектов и связей между ними. Для того чтобы, исходя из этих определений, построить любую теорию, нужно систему наделить некоторой структурой. Различные варианты представления структуры множеств V_i приводят к разным подходам в описании систем, в частности, так возникают понятия временных и алгебраических систем. Оба эти подхода в основном эквивалентны, однако мы ограничимся рассмотрением ЭС как временных, поскольку этот подход более очевиден с точки зрения содержательного описания функционирования ЭС.

Для описания ЭС на архитектурном уровне важную роль играет понятие системы “вход-выход” или “черного ящика”. Множество индексов I разобьем на два подмножества: $I_x \subset I$ и $I_y \subset I$, так что

$I_x \cap I_y = \emptyset$, $I_x \cup I_y = I$. Входом (входным объектом) системы и ее выходом (выходным объектом) назовем соответственно множества

$$X = \times_{i \in I_x} \{V_i : i \in I_x\}; \quad Y = \times_{i \in I_y} \{V_i : i \in I_y\}. \quad (1)$$

Тогда $S_1 \subset X \times Y$, а система $m_{\bar{V}_1}^D$ представима тройкой (X, Y, S_1) . Назовем ее системой типа “вход-выход” или “черный ящик”.

Здесь S_1 отражает функции системы и её структурные свойства, а X и Y содержат информацию обо всех поступающих на входы системы воздействиях или всех её ответах, выдаваемых в окружающую среду. Таким образом, система определяется в терминах ее наблюдаемых свойств (возможные комбинации значений ее входов и выходов) и взаимосвязей между ними.

С помощью данного аппарата можно описать любую техническую систему, однако он не отражает ни структуру, ни поведение ЭС во времени в той степени, в какой это нам требуется на выбранном уровне абстракции. Для того чтобы получить возможность описывать системы более детально, введем дополнительные понятия.

Для изучения структуры системы нам нужно нечто большее, чем её понимание как совокупности взаимосвязанных входов и выходов. При проектировании и исследовании любой ЭС, независимо от уровня рассмотрения, её принято разбивать на

некоторое множество элементов $\bar{E} = \{E_i : i \in I\}$, где I — множество индексов. Тогда, со структурной точки зрения, ЭС представима как пара $(\bar{E}, S)$, где S — множество отношений над элементами базового множества $\bar{E}$: $S = \{R_i, i = \overline{1, n}\}$, $R_i \subseteq \bar{E} \times \bar{E}$, $i = \overline{1, n}$. При этом для описания структуры системы наиболее часто используются следующие отношения [2]:

1) включения RV_i , чтобы определить, какие элементы включают в себя другие элементы системы, формируя таким образом ее иерархию;

2) связи RP_i , чтобы установить, как элементы системы взаимодействуют друг с другом.

Кроме описания структуры ЭС, для нас представляет интерес описание ее динамики (поведения). Для построения такого описания необходимо, чтобы при задании системы “вход-выход” (X, Y, S_1) отношения между входными и выходными объектами учитывали время. Множеством моментов времени назовем линейно упорядоченное множество, в качестве минимального элемента включающее 0: $T = \{t : t \geq 0\}$.

Пусть существуют семейства множеств A и B , называемые соответственно алфавитом входа и алфавитом выхода системы. Структура этих множеств такова, что если каждый объект системы V_i , где $i \in I_X$, может принимать значения из множества A' , а каждый объект V_i , где $i \in I_Y$, может принимать значения из множества B' , то $A = \times_{i \in I_X} A'$,

$B = \times_{i \in I_Y} B'$. Значит, A и B содержат все возможные варианты входных и выходных векторов для системы “вход-выход”.

Пусть также задано множество моментов времени T . Обозначим через A^T и B^T множества всевозможных отображений из T в A и B (т.е. множества всех возможных функций входных и выходных векторов от времени для алфавитов A и B). Тогда входным и выходным временными объектами системы назовем множества

$$X \subset A^T, Y \subset B^T. \quad (2)$$

Если $S_1 \subset X \times Y$, то тройку (X, Y, S_1) будем называть временной системой. Элементами множеств X и Y являются временные функции $x: T \rightarrow A$ и $y: T \rightarrow B$, их значения в момент времени t_1 будем обозначать $x(t_1)$ и $y(t_1)$. Можно сказать, что в каждый момент времени $t_1 \in T$ множества X и Y содержат все возможные значения, которые в данный момент могут поступать на вход или сниматься с выхода системы.

Как отмечалось выше, понятие временной системы может быть получено из определения системы вообще. Для этого нужно предположить, что

объекты системы V_i являются множествами, состоящими из временных функций $v: T \rightarrow A$. Если области определения и области значений для всех функций v объекта V_i совпадают, то A называется алфавитом объекта. Следует отметить, что хотя, по определению, вход и выход системы (1) представляют собой множества n -ок, элементами которых являются временные функции v , а входной и выходной временные объекты (2) — просто множествами временных функций, ясно, что при временных системах эти понятия эквивалентны. Только в случае (1) временными функциями описывается каждый объект V_i , а в случае (2) — объекты, принадлежащие входу или выходу системы и сгруппированные в векторы, и только для каждого такого вектора построена временная функция. Итак, оба определения, несмотря на разницу в структурах множеств, описывают разными способами одно и то же, поэтому вход и выход системы будем в дальнейшем рассматривать, используя оба варианта определений.

При исследовании поведения системы мы должны иметь возможность в любой момент времени $t_1 \in T$ по вектору входных воздействий $x(t_1)$ однозначно определить значение выхода системы $y(t_1)$.

Однако для временной системы (X, Y, S_1) S_1 в момент времени t_1 содержит все допустимые для системы пары типа $(x(t_1), y(t_1))$, и одному значению входного вектора может соответствовать несколько возможных значений выходного. Для однозначного воспроизведения динамики системы необходимо ввести понятие динамической системы, которое в общей теории систем применяется тогда, когда возникает необходимость исследовать, как система развивается во времени, т.е. установить взаимосвязь между значениями объектов, относящимися к различным моментам времени. При рассмотрении ЭС на выбранном уровне абстракции видно, что она относится к классу динамических систем.

Введем дополнительный объект системы $C = \times_{i \in I_C} \{V_i : i \in I_C\}$ — объект ее состояний, где

$I_C \subset I$, $I_X \cap I_Y \cap I_C = \emptyset$, $I_X \cup I_Y \cup I_C = I$. Теперь “черный ящик” представляется в виде четверки (X, Y, C, S_1) , где $S_1 \subset X \times Y \times C$. При этом для C также определен алфавит состояний Z , формируемый из алфавитов всех объектов V_i , $i \in I_C$, и $C \subset Z^T$.

Объект состояний C в любой момент времени содержит в себе всю информацию о предыстории системы. Для определения значений выхода системы нужно построить зависимость Y от значений входного объекта и объекта состояний. Введем семейство вспомогательных функций, называемых функциями выходов или реакций системы p_t :

$$\bar{p} = \{p_t : C_t \times X_t \rightarrow Y_t \text{ \& } t \in T\}. \quad (3)$$

Функции выходов в каждый момент времени t_1

для любой из всех возможных комбинаций значений входа и объекта состояний системы однозначно определяют выходной объект.

С течением времени объект состояний также изменяется — он накапливает информацию о поведении системы в предыдущие моменты времени. Поэтому для описания ее динамики необходимо ввести также функцию, управляющую состоянием системы, исходя из значений её входа и состояния. Семейство таких функций назовем семейством функций перехода состояния $\varphi_{tt'}$:

$$\bar{\varphi} = \{ \varphi_{tt'} : C_t \times X_{tt'} \rightarrow C_{t'}, \& t, t' \in T \& t' > t \}. \quad (4)$$

В приведенных выше определениях считается, что имеется два множества моментов времени:

$T_t = \{ t' : t' \geq t \}$ и $T_{tt'} = \{ t^* : t' \geq t^* \geq t \}$, X_t , Y_t , C_t — сужения X , Y , C на T_t , а $X_{tt'}$, $Y_{tt'}$, $C_{tt'}$ — сужения X , Y , C на $T_{tt'}$. Система, представляемая пятеркой $(X, Y, C, \bar{\varphi}, \bar{p})$, называется динамической.

Исходя из сказанного выше, можно сделать вывод, что для построения имитационной модели ЭС нужно иметь ее структурно-динамическое описание в виде $(A, B, Z, X, Y, C, \bar{p}, \bar{\varphi}, E, RV, RP)$. Далее с использованием средств, предоставляемых системой моделирования, нужно каждый элемент данного множества отобразить в имитационной модели. Предполагается, что в системе моделирования имеется специализированный язык описания, позволяющий представить A, B, Z, X, Y, C, E, RV и RP соответствующими структурами данных, а семейства функций переходов и выходов — некоторыми алгоритмами. Каким образом строятся алгоритмы имитации, зависит от многих факторов, среди которых: выбранный способ описания объекта моделирования, используемая технология имитационного моделирования, назначение создаваемой модели (для исследования или для автоматического синтеза схемы) и т.д. Остановимся лишь на методе описания функций переходов и выходов объекта исследования имитационными алгоритмами.

Рассмотрев семейства функций (3) и (4), делаем вывод, что с точки зрения алгоритма, описывающего данные функции, все объекты системы можно разделить на два вида — объекты, значения которых могут изменяться алгоритмом (Y для $\bar{p}$ и C для $\bar{\varphi}$), и объекты (X и C), значения которых анализируются алгоритмом для определения, каким образом изменять объекты первого вида. Разобьем множество индексов I объектов системы V_i на два пересекающихся подмножества: $I_R \subset I$ и $I_W \subset I$, так что $I_R \cap I_W = I_C$, $I_R \cup I_W = I$. Объект V_i системы отнесем к множеству анализируемых объектов, если $i \in I_R$, и к множеству изменяемых объектов, если $i \in I_W$. Тогда, сделав некоторые упрощения, любую функцию из (3) и (4) можно представить в виде

$$F: R \rightarrow W, \quad (5)$$

где $R = \times_{i \in I_R} \{ V_i : i \in I_R \}$, $W = \times_{i \in I_W} \{ V_i : i \in I_W \}$. Отно-

шение F можно представить как таблицу переходов-выходов системы для каждого момента времени. Именно функцию такого вида описывает имитационный алгоритм, поэтому в дальнейшем не будем оговаривать особо, куда относится изменяемый объект — к выходу системы или к её состоянию. Решим теперь задачу перехода от представления системы в виде (5) к имитационному алгоритму.

Современные системы имитационного моделирования предоставляют исследователю специализированные языковые средства для описания объекта моделирования. Языки описания, как правило, содержат набор стандартных алгоритмических средств, расширенный дополнительными языковыми конструкциями, отражающими специфику исследуемых явлений. Для простоты будем считать, что любой оператор языка описания представим в виде функции F_n , которая однозначно определяет единственный объект модели по значениям некоторых объектов-параметров. Суперпозиция нескольких таких операторов, описывающая изменение значения только одного объекта, также представима некоторой функцией F_n .

Теперь, выбрав индекс $n \in I_W$, образуем для него разбиение множества F на подмножества F_{ni} , такие что $\bigcup_i F_{ni} = F$, $\bigcap_i F_{ni} = \emptyset$ и

$$(\exists F_n) \left(\forall (r_{ni}, w_{ni}) \in F_{ni} \right) \left(F_n(r_{ni}) = v_n; v_n \in w_{ni} \right),$$

где v_n — временная функция, принадлежащая объекту V_n системы, который входит в декартово произведение W под порядковым номером n ; каждое подмножество F_{ni} представляет собой отображение $F_{ni} : R_{ni} \rightarrow W_{ni}$, а $r_{ni} \in R_{ni}$, $w_{ni} \in W_{ni}$. Отношение F для выбранного n можно представить в виде $F_n = \bigcup_i F_{ni} = \bigcup_i (R_{ni} \rightarrow W_{ni})$. Обозначим через $\bar{F}$

множество всех возможных разбиений F_{ni} для всех $n \in I_W$: $\bar{F} = \bigcup_{n \in I_W} F_n$.

Смысл приведенных преобразований отношения (5) заключается в следующем. Для каждого объекта системы $V_n, n \in (I_Y \cup I_C)$ мы нашли множество функций, отражающих закон изменения данного V_n во времени в зависимости от состояния системы и её входа, причем эти функции таковы, что их можно описать с помощью языковых средств выбранной системы моделирования. Тогда при моделировании работы ЭС в данный момент времени нужно для каждого отношения из $\bar{F}$ выполнить следующие действия:

1. Проверить, входят ли текущие значения входа и состояния в R_{ni} . Операцию проверки условий в имитационном алгоритме будем обозначать P_{ni} .

2. Если результат проверки условий — истина, нужно выполнить функцию F_n , передав ей в качестве параметров R_{ni} . В результате объект V_n должен принять новое значение, спустя T_{ni} временных единиц. Выполнение такой операции имитационным алгоритмом обозначим A_{ni} .

Таким образом, с помощью подмножеств F_{ni} задано элементарное изменение, которое может произойти в системе, называемое событием. Будем говорить, что функция F_{ni} задает функциональное действие (ФД). ФД становятся причиной событий, а события, изменяя состояние системы, могут инициировать ФД. Тогда $\bar{F}$ — множество всех возможных для системы функциональных действий; R_{ni} — множество условий выполнения ФД F_{ni} ; F_n — алгоритм, описывающий данное ФД. В имитационной модели ФД представляется парой (P_{ni}, A_{ni}) , где A_{ni} выполняется только в случае истинности P_{ni} .

Разбиение поведения системы на отдельные ФД является ключевым моментом при переходе от представлений о динамике системы как единого целого к её имитационной модели. Однако данный подход фактически реализуем лишь в применении к описанию простейших систем, так как при достаточно большой мощности множества $\bar{V}_1$ количество проверок условий P_{ni} огромно. Значительного повышения эффективности моделирования можно достичь, если учитывать особенности объекта исследования — ЭС [3]. Во-первых, условия выполнения ФД содержат большую избыточность и могут быть минимизированы. Это объясняется тем, что в ЭС каждый объект V_i имеет небольшое количество связей с другими объектами. Тогда в элементах множеств R_{ni} для многих V_i содержатся все возможные переборы их временных функций, и соответствующие проверки можно опустить.

Во-вторых, в любой ЭС в конкретный момент времени может выполняться только очень низкий процент ФД от общего их количества. Следует также учитывать еще один широко известный факт — неоднородность распределения событий в ЭС во времени. События вдоль оси времени собираются в группы, началом каждой такой группы служит некоторое значимое для ЭС событие, называемое событием синхронизации. Его приход вызывает выполнение большого количества ФД, после чего плотность событий во времени снова падает. Кроме того, подсистемы ЭС обычно реализуют четко установленные последовательности ФД. Тогда проверку условий нужно проводить лишь для первого ФД (которое обычно и вызывается событием синхронизации), а для следующих ФД проверки либо не требуются вообще, либо очень упрощены.

В имитационной модели эти особенности предлагается отразить следующим образом. Пусть моделируемая подсистема ЭС выполняет определенную последовательность действий, некоторое ФД₁ является первым ФД последовательности и иницииру-

ется проверкой условий R_1 . За ним следует ФД₂, для выполнения которого необходимо проверить условия R_2 . Если условия ложны, выполнение последовательности приостановится и будет продолжено лишь в случае выполнения этих условий. При выполнении последнего ФД из последовательности управление передается снова в её начало. Таким образом, множество всех ФД системы $\bar{F}$ мы разбили на группы вида $((P_1, A_1), (P_2, A_2), \dots, (P_n, A_n))$. Логически связанный набор ФД, выполняющихся последовательно, назовем процессом [4]. Будем говорить, что процесс находится в состоянии ожидания или в неактивном состоянии, если условия выполнения текущего ФД процесса ложны, и в активном состоянии, если они истинны. Условия R_{ni} , которые должны быть выполнены для перехода процесса в активное состояние, назовем условиями активизации процесса.

Поскольку одни и те же условия R_{ni} могут входить сразу в несколько F_{ni} , в имитационной модели определенное количество процессов одновременно могут находиться в активном состоянии и воздействовать на различные V_n . Можно сказать, что, в то время, как внутри каждого процесса все ФД выполняются последовательно, ФД различных процессов могут выполняться параллельно во времени (одновременно).

Итак, процессный подход при построении имитационной модели ЭС предполагает выполнение исследователем действий в такой последовательности:

1. Изучение ЭС и её декомпозиция на законченные функциональные модули, каждый из которых реализует определенную последовательность ФД.

2. Разбиение поведения каждого модуля во времени на некоторые элементарные действия и выявление условий выполнения этих действий в последовательности.

3. Составление программной модели устройства, в которой каждый из модулей п.1 описывается одним параллельным процессом, каждый процесс состоит из последовательности описаний ФД, а каждое ФД состоит из оператора проверки условий выполнения и операторов изменения значения некоторого объекта модели.

Данный подход имеет ряд преимуществ как при описании систем, так и при проведении вычислительных экспериментов. В первом случае он позволяет легко осуществлять переход непосредственно от содержательных представлений о поведении системы к её модели. В самом деле, исследователь, составляющий имитационную модель устройства, обычно имеет описание системы не в виде таблиц истинности (для сложных ЭС они слишком громоздки), а в виде спецификации на её отдельные функции или описания алгоритмов её работы в отдельных режимах. Тогда он без труда может каждой функции системы поставить в соответствие процесс. Таким образом, модель на основе процессов имеет интуитивно понятную для исследователя структуру, достаточно проста для составления, чтения и модификации.

С точки зрения процесса моделирования данный подход значительно повышает производительность имитационной системы. Поскольку большинство процессов модели в конкретный момент времени находится с состоянием ожидания, для каждого из них осуществляется минимальное количество проверок условий активизации — только для того ФД процесса, на котором он был приостановлен.

Процессный метод предполагает наличие сложных алгоритмов управления процессом моделирования (УПМ) [5], осуществляющих переключение процессов из активного состояния в неактивное и наоборот, обеспечивающих их квазипараллельное выполнение, разрешающих конфликты при совместном использовании параллельными процессами общих данных и т.д. Однако решение этих задач полностью возлагается на программу УПМ и не требует затрат времени со стороны пользователя.

В заключение следует отметить, что в статье рассмотрены лишь вопросы моделирования поведения ЭС. Моделирование же её структуры, способы представления самих объектов системы, построение алгоритмов УПМ и т. д. являются отдельными вопросами, которые, однако, нельзя оставить без внимания при построении процессно-ориентированной системы имитационного моделирования. Исследование перечисленных проблем и дальнейшее развитие приведенного процессного подхода позволит в будущем не только формализовать процесс

построения моделей, но и решать вопросы автоматического синтеза схем ЭС на основе их имитационных моделей.

Литература: 1. Месарович М., Такахара Я. Общая теория систем: математические основы. М.: Мир, 1978. 311 с. 2. Литвинов В.В. Математическое обеспечение проектирования вычислительных систем и сетей. М.: Техника, 1982. 176 с. 3. Аврамчук Е., Вавилов А., Емельянов С. Технология системного моделирования. М.: Машиностроение; Берлин: Техник, 1988. 520 с. 4. Анисимов А.В., Бореиша Ю.Е. Проектирование сложных систем посредством параллельных взаимодействующих процессов в режиме имитационного моделирования. К.: ИК, 1983. 38 с. 5. Максимей И. Имитационное моделирование на ЭВМ. М.: Радио и связь, 1988. 232 с.

Поступила в редколлегию 13.04.98

Рецензент: д-р техн. наук, проф. Евдокимов А.Г.

Горбачев Валерий Александрович, канд. техн. наук, профессор кафедры ЭВМ ХТУРЭ. Научные интересы: теория моделирования. Адрес: 310135, Украина, Харьков, ул. Героев Труда, 133, кв. 33, тел. 66-53-61, 40-93-54.

Бабаев Андрей Павлович, аспирант, младший научный сотрудник кафедры ЭВМ ХТУРЭ. Научные интересы: теория систем, теория моделирования. Адрес: 310100, Украина, Харьков, ул. Танкопия, 24, кв. 136, тел. 97-97-39, 40-93-54.

Волк Максим Александрович, аспирант, младший научный сотрудник кафедры ЭВМ ХТУРЭ. Научные интересы: теория проектирования электронных систем, теория моделирования. Адрес: 310124, Украина, Харьков, пр. Гагарина, 175а, кв. 9, тел. 52-04-17, 40-93-54.

УДК 681.323

ОЦЕНКА ТОЧНОСТИ ГЕОМЕТРИЧЕСКИХ ПРЕОБРАЗОВАНИЙ В СПЕЦПРОЦЕССОРЕ РАСТРОВОЙ ГРАФИКИ

ГУСЯТИН В.М.

Предложено оценить точности геометрических преобразований в спецпроцессорах растровой графики характеризовать угловым разрешением. Получено аналитическое соотношение, которое позволяет связать параметры спецпроцессора с угловым разрешением. Варьируя эти параметры, можно решать задачу оптимизации системы в целом.

При проектировании спецпроцессоров (СП) растровой графики [1,2] для цифровых систем визуализации (СВ) актуальной является задача выбора параметра для оценки точности геометрических преобразований, а также получения соотношений, устанавливающих связь между этими параметрами, техническими характеристиками отображаемого экрана, СП и разрешающей способностью зрения человека, для которого в конечном итоге и выполняется синтез изображения.

1. Постановка задачи

В качестве основного ориентира, к которому следует стремиться при оценке требуемой точности геометрических преобразований проектируемой СВ,

выберем угловую разрешающую способность глаза человека [3]: $\gamma \approx 2^{-12} \dots 2^{-13}$ рад.

Синтезируемое изображение отображается на экране с растровой разверткой [1]. Введем параметр системы отображения ω , равный отношению размера пиксела экрана l к модулю вектора наблюдения $|\bar{V}_c|$, проведенного от центра проекции к центру экрана: $\omega = l/|\bar{V}_c|$ рад.

Вычисление точки пересечения проекционного луча с поверхностью объекта сцены [1] выполняется в соответствии с соотношением

$$\bar{r}_p = \bar{r}_h + \bar{V}_p. \quad (1)$$

Слагаемое $\bar{r}_h$ является константой для каждого вновь синтезируемого кадра изображения и вычисляется с высокой точностью на ПЭВМ [2]. Таким образом, вся погрешность вычислений сосредоточена в слагаемом $\bar{V}_p$. Вектор $\bar{V}_p$ — геометрическая интерпретация проекционного луча в обратном методе трассирования, используемого для синтеза объектов в трехмерной графике [1]:

$$\bar{V}_p = \lambda \bar{V}. \quad (2)$$

Координаты начала вектора $\bar{V}_p$ вычисляются точно, так как являются координатами конца вектора $\bar{r}_h$. Ошибка накапливается при вычислении точки пересечения конца вектора $\bar{V}_p$ с поверхностью объекта сцены и, следовательно,