

кольных автоматов специфицируется  $N$ -компонентным вектором  $R=(R_1, \dots, R_n)$ , каждая компонента которого представляет регулярное выражение в алфавите  $Z=\{z_k | z_k=(E_{C_k}, LA_k), k=1, n\}$ . Буква  $z_k$  входного алфавита  $Z$  обозначает элементарное событие, наступление которого становится возможным только при истинности набора условий  $E_{C_k}$ , т.е.  $E_{C_k}=1$  определяет возможность перехода  $PA_i$  в очередное состояние. Причиной перехода  $i$ -го протокольного автомата из одного состояния в другое является либо прием сообщения от  $PA_j (j \neq i)$ , либо посылка сообщения  $j$ -му ПА, либо результат некоторых внутренних действий, не связанных с посылкой или приемом сообщений (например, таймаут). Находясь в конкретном состоянии,  $PA_i$  обязан выполнить список действий  $LA_k$  ( $k$ -ю процедуру), что, в свою очередь, может быть одной из причин наступления следующего события. Переход в одно из конечных состояний свидетельствует о выполнении протокольным автоматом какой-либо услуги из предоставляемого сервиса.

Из сказанного выше следует, что ПА в чистом виде не является конечным автоматом. Однако если каждой выполняемой ПА процедуре ( $LA_k$ ) сопоставить внутреннее состояние автомата, а функцию переходов  $\Phi$  при этом определить на множестве выделенных процедур (алгоритмов выполнения  $LA_k$ ), то  $PA_i$  в такой интерпретации, безусловно, может быть воспринят как рассмотренная модель конечного автомата со всеми вытекающими последствиями. А это значит, что все без исключения методы преобразования, анализа и синтеза конечных автоматов могут быть применены и для протокольных автоматов. Сам же ПА может быть задан на любом из автоматных языков.

Использование языка регулярных выражений для спецификации ПА позволяет, во-первых, получить довольно компактные описания и, во-вторых, организовать автоматическое проведение таких процедур проектирования протоколов как логическое проектирование, установление логической корректности, моделирование, декомпозиция и, наконец, реализация.

Полученные на этапе спецификации регулярные выражения  $R_1, \dots, R_n$ , определяющие поведение протокольных автоматов  $PA_1, \dots, PA_n$ , являются исходным материалом для процедуры алгебраической верификации. Используя аксиоматическую систему алгебры регулярных событий, расширенную рядом тождественных преобразований, регламентирующих взаимодействие процессов, можно предложить эффективную процедуру установления логической корректности (полнота и непротиворечивость) первичной спецификации анализируемого протокола.

Этап логического проектирования протоколов базируется на использовании алгоритма абстрактного синтеза конечного автомата. Результат этого этапа проектирования ПА (таблица переходов) с успехом может быть использован для проведения верификации протокола методом анализа достижимых состояний. Этот метод, являясь одним из самых мощных, позволяет установить наличие или отсутствие практически всех свойств, которыми должен обладать протокол, независимо

от выполняемых им функций. Это и функциональная полнота, и непротиворечивость, и отсутствие неспецифицированных приемов, статических и динамических тупиков и, наконец, выполнение назначения протокола. Возникающая при реализации метода анализа достижимых состояний проблема размерности может быть сведена к задаче декомпозиции конечного автомата.

Этап реализации протоколов в той или иной аппаратной или программной среде может быть осуществлен соответственно методами структурного синтеза автоматов или путем преобразования регулярных выражений в алгоритмические структуры протоколов на каком-либо языке программирования высокого уровня.

**Список литературы:** 1. Апраксин Ю.К. К вопросу автоматизации проектирования протоколов вычислительных сетей на основе конечноавтоматной модели // Автоматика и вычислительная техника. 1987. №2. С.12-19. 2. Kleene S.C. Representation of events in nerve nets and finite automata // In C.E. Shannon, J.McCarthy (eds) Automata Studies.-Princeton Univers. Press, 1956. P. 3-41. 3. Salomaa A. The complete axiom systems for the algebra of regular events // Journal of the Association for Computing Machinery. 1966. Vol.13, №1. P.152-169. 4. Brzozowski J.A. Derivatives of regular expressions// Journal of the Association for Computing Machinery. 1964. Vol.11, №4. P.481-494.

Поступила в редколлегию 12.01.2000

**Апраксин Юрий Константинович**, канд. техн. наук, профессор кафедры КиВТ СевГТУ. Научные интересы: проектирование и анализ протоколов распределенных систем. Адрес: 99040, Севастополь, ул. Хрусталева, 87, кв. 38, тел. (0692) 44-31-32.

УДК 681.323

*В.М. ГУСЯТИН, А.П.ОСТРОУШКО*

## **МАТЕМАТИЧЕСКАЯ МОДЕЛЬ И АЛГОРИТМ ОБРАБОТКИ МЕТЕОУСЛОВИЙ ДЛЯ СИСТЕМ ВИЗУАЛИЗАЦИИ**

Предлагается математическая модель обработки прозрачности атмосферы. Получены формулы для трех возможных вариантов положения наблюдателя относительно облачного слоя и разработан алгоритм их вычисления в системах визуализации реального масштаба времени.

### **1. Постановка задачи**

При проектировании систем визуализации (СВ) тренажеров разработчики стремятся к формированию изображения, наиболее полно соответствующего реальной картине, наблюдаемой из кабины транспортного средства (ТС). Возможность обработки в СВ атмосферной дымки, тумана, облаков повышает реалистичность синтезируемого изоб-

ражения и расширяет перечень упражнений для отработки навыков управления ТС в экстремальных ситуациях.

На видимость объектов существенно влияет ослабление слоем атмосферы света, отраженного от объекта. С некоторым приближением можно считать, что ослабление света атмосферой вызывается только его рассеянием, которое создает так называемую дымку [1]. Воздушная дымка имеет яркость, так как взвешенные в воздухе частицы пыли, кристаллы льда, капли, водяной пар и сами молекулы воздуха рассеивают падающий на них световой поток. Наличие дымки в атмосфере приводит к снижению контраста между наблюдаемым объектом и фоном. Этому способствуют два фактора. Первый фактор – это уменьшение истинной яркости объекта вследствие ослабления света слоем атмосферы. Второй – это кажущееся превращение цветного объекта в ахроматический из-за уменьшения насыщенности цвета вследствие наложения яркости атмосферной дымки на яркость объекта.

Таким образом, из [1] следует, что изменение видимости объекта днем может быть проимитировано в СВ путем изменения насыщенности и яркости цвета объекта в зависимости от расстояния между объектом и наблюдателем.

## 2 Математическая модель

Для вычисления кажущейся яркости объекта, находящегося от наблюдателя на расстоянии  $l$ , воспользуемся соотношением из [1]:

$$L = L_0\tau(l) + L_\infty[1 - \tau(l)], \quad (1)$$

где  $L_0$  – исходная яркость цвета объекта;  $L_\infty$  – коэффициент, характеризующий яркость дымки при  $l \rightarrow \infty$ ;  $\tau$  – коэффициент пропускания атмосферы.

В приведенном уравнении (1) первое слагаемое в соответствии с законом Бугера показывает ослабление яркости цвета объекта, находящегося на расстоянии  $l$  от наблюдателя. Второе слагаемое является световоздушным уравнением Кошмидера и описывает изменение яркости дымки в зависимости от толщины атмосферного слоя. Уменьшение насыщенности цвета вследствие наложения яркости атмосферной дымки на яркость объекта охарактеризуем следующим соотношением:

$$S = S_0\tau(l), \quad (2)$$

здесь  $S_0$  – исходная насыщенность цвета объекта.

В случае оптически однородного слоя атмосферы толщиной, равной единице длины,  $\tau$  определяется как удельная прозрачность атмосферы  $\tau_1$ . Зная удельную прозрачность и толщину однородного слоя атмосферы, можно определить коэффициент пропускания:

$$\tau = \tau_1^l. \quad (3)$$

Обычно  $\tau_1 = e^{-\alpha}$  или  $\tau_1 = 10^{-\delta}$ , где показатели  $\alpha$  и  $\delta$  характеризуют степень рассеяния (ослабления) света атмосферой [1]. Для упрощения дальнейших вычислений введем показательную двоичную функцию:

$$\tau = 2^{-\beta}; \quad (4)$$

$$\beta = 1/l^0, \quad (5)$$

где  $l^0$  – толщина однородного атмосферного слоя заданной прозрачности, на которой коэффициент пропускания уменьшается в два раза.

Из (4) и (5) имеем:

$$l^0 = -1/\log_2\tau_1. \quad (6)$$

Для единообразной оценки прозрачности атмосферы принята международная шкала видимости, в которой удельная прозрачность атмосферы  $\tau_1$  связана с метеорологической дальностью видимости (МДВ). Таким образом, введенный нами параметр  $l^0$  может быть однозначно привязан к МДВ, что позволяет задавать и обрабатывать метеоусловия в СВ в соответствии с принятой практикой.

В случае неоднородности слоев атмосферы коэффициент пропускания определим следующим образом:

$$\tau = 2^{-\sum_{i=1}^n \frac{l_i}{l_i^0}}, \quad (7)$$

где  $n$  – число участков (слоев) различной прозрачности на пути прохождения проекционного луча в атмосфере;  $l_i$  – длина  $i$ -го участка пути проекционного луча в атмосфере с заданным однородным слоем прозрачности;  $l_i^0$  – параметр прозрачности  $i$ -го слоя атмосферы.

Воспользуемся цветовой моделью TSL ( $T$  – цветовой тон,  $L$  – яркость,  $S$  – насыщенность) [2]:

$$\begin{aligned} R &= L - S(\cos T + \sqrt{3} \sin T)/3; \quad G = L + 2S \cos T/3; \\ B &= L - S(\cos T - \sqrt{3} \sin T)/3. \end{aligned} \quad (8)$$

Подставим выражения для  $L$  и  $S$  из формул (1) и (2):

$$\begin{aligned} R &= L_0\tau + L_\infty(1 - \tau) - S_0\tau(\cos T_0 + \sqrt{3} \sin T_0)/3; \\ G &= L_0\tau + L_\infty(1 - \tau) + 2S_0\tau \cos T_0/3; \\ B &= L_0\tau + L_\infty(1 - \tau) - S_0\tau(\cos T_0 - \sqrt{3} \sin T_0)/3. \end{aligned}$$

Произведя замены в соответствии с уравнением (8), получаем:

$$R = R_0\tau + L_\infty(1 - \tau), \quad G = G_0\tau + L_\infty(1 - \tau), \quad B = B_0\tau + L_\infty(1 - \tau).$$

Так как  $0 \leq \tau \leq 1$ , упростим это соотношение:

$$R = R_0\bar{\tau} + L_\infty\bar{\tau}, \quad G = G_0\bar{\tau} + L_\infty\bar{\tau}, \quad B = B_0\bar{\tau} + L_\infty\bar{\tau}, \quad (9)$$

где  $\bar{\tau}$  – инверсия от  $\tau$ .

Для нахождения длины  $i$ -го участка пути проекционного луча в атмосфере применим математическую модель геометрических преобразований изображения, изложенную в [3]:

$$l_i = \frac{VH_i}{V_y}, \quad (10)$$

где  $V$  – модуль вектора наблюдения;  $V_y$  – проекция вектора  $V$  на ось  $Y$ ;  $H_i$  – проекция  $l_i$  на ось  $Y$ .

Разделим в (10) левую и правую часть на  $l_i^0$  и подставим в (7):

$$\tau = 2^{-\frac{V}{V_y} \sum_{i=1}^n \frac{H_i}{l_i^0}}. \quad (11)$$

Таким образом, соотношения (9) и (11) позволяют вычислить для каждого пиксела экрана с учетом прозрачности участков атмосферы составляющие цвета RGB объекта, на который попал проекционный луч.

### 3 Алгоритм обработки

Будем полагать, что данные по вектору наблюдения и проекционному лучу вычисляются исходя из [3].

Рассмотрим пример, представленный на рисунке. Точки 1, 2, 3 определяют три возможных случая положения центра проекции на высоте  $H$  по отношению к поверхности Земли. На рисунке приняты следующие обозначения:  $V$  – расстояние от точки наблюдения до экрана монитора,  $V_y$  – проекция вектора  $V$  на ось  $Y$ ,  $H$  – высота точки наблюдения,  $H_{мин}$  – высота облачного слоя,  $\Delta H$  – толщина облачного слоя,  $H_{макс}$  – предельное значение высоты.

Для упрощения вычислений воспользуемся логарифмической системой счисления, аналогично [3]. Обозначим показатель в (11) через  $\rho$ . Тогда имеем:

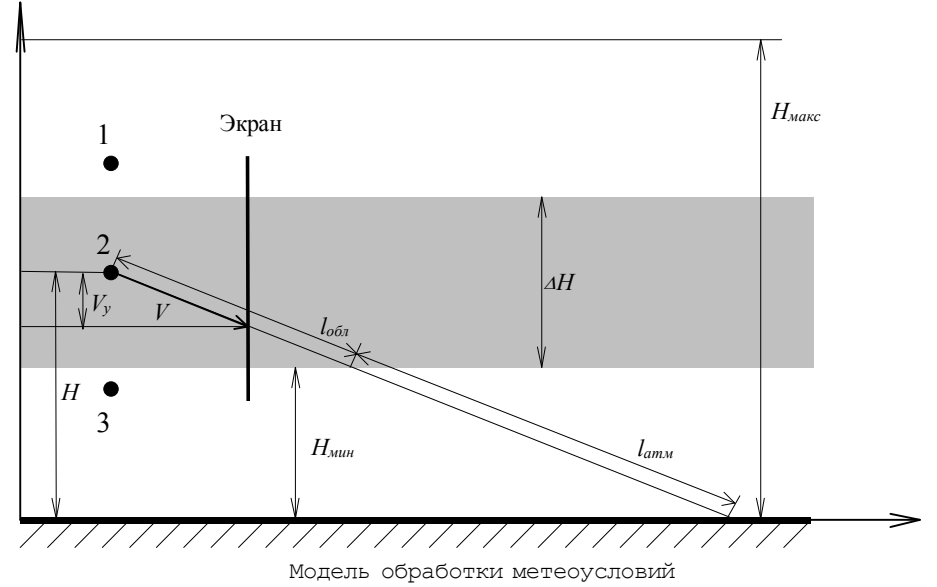
$$\log_2 \rho = \log_2 \left( V \sum_{i=1}^n \frac{H_i}{l_i^0} \right) - \log_2 V_y. \quad (12)$$

В правой части соотношения (12) уменьшаемое можно считать неизменяющимся в течение кадра и поэтому необходимо вычислять его один раз на каждый кадр. Вычитаемое в правой части уравнения вычисляется для каждого пиксела экрана [3].

В соответствии с (12) распишем три случая, представленных на рисунке, с учетом возможного положения проекционных лучей ниже (а) и выше (б) линии горизонта.

1.  $H \geq H_{обл} + \Delta H$ , точка 1 на рисунке:

$$\log_2 \frac{l}{l_0} = \log_2 \left[ \left( \frac{H - \Delta H}{l_{атм_0}} + \frac{\Delta H}{l_{обл_0}} \right) V \right] - \log_2 V_y; \quad (13a)$$



$$\log_2 \frac{l}{l_0} = \log_2 \left( \frac{H_{макс} - H}{l_{атм_0}} V \right) - \log_2 V_y. \quad (13б)$$

2.  $H_{обл} < H < H_{обл} + \Delta H$ , точка 2 на рисунке:

$$\log_2 \frac{l}{l_0} = \log_2 \left[ \left( \frac{H_{мин}}{l_{атм_0}} + \frac{H - H_{мин}}{l_{обл_0}} \right) V \right] - \log_2 V_y. \quad (14a)$$

$$\log_2 \frac{l}{l_0} = \log_2 \left[ \left( \frac{H_{макс} - \Delta H - H_{мин}}{l_{атм_0}} + \frac{H_{мин} + \Delta H - H}{l_{обл_0}} \right) V \right] - \log_2 V_y. \quad (14б)$$

3.  $H < H_{обл}$ , точка 3 на рисунке:

$$\log_2 \frac{l}{l_0} = \log_2 \frac{HV}{l_{атм_0}} - \log_2 V_y; \quad (15a)$$

$$\log_2 \frac{l}{l_0} = \log_2 \left[ \left( \frac{H_{макс} - H - \Delta H}{l_{атм_0}} + \frac{\Delta H}{l_{обл_0}} \right) V \right] - \log_2 V_y. \quad (15б)$$

Перечислим основные этапы обработки. При этом предположим, что СВ состоит из ПЭВМ и спецпроцессора, который, как минимум, включает в себя подсистему геометрической обработки (ПГО) в соответствии с [3] и рассматриваемую в данной статье подсистему обработки метеоусловий (ПОМ).

1. ПЭВМ для каждого кадра изображения вычисляет пару уменьшаемых в соотношениях (13) – (15) в зависимости от случаев 1, 2, 3, которые заносятся в ПОМ, где хранятся в течение обработки кадра изображения. В ПОМ заносится также параметр  $I_{\infty}$ .

2. На каждом такте работы конвейера в ПОМ из ПГО поступает  $\log_2 V_y$ .

3. Вычисляется  $\log_2 p$  в соответствии с (13)–(15).

4. Табличным методом вычисляется коэффициент пропускания  $\tau$  (7).

5. Одновременно вычисляются составляющие цвета (9). Умножения в (9) выполняются табличным методом.

В заключение отметим, что приведенная математическая модель и алгоритм обработки метеоусловий использованы при построении СВ, образцы которой установлены на комплексный тренажер в КТС ТУ-154М в учебно-сертификационном центре гражданской авиации в г. Киеве, а также в Харьковском институте летчиков на летный тренажер ТЛ-39.

**Список литературы:** 1. *Электросветосигнальное оборудование аэродромов* / Фрид Ю. В., Величко Ю. К., Козлов В. Д. и др. М.: Транспорт, 1988. 318 с. 2. *Мартинес Ф.* Синтез изображений: Пер. с франц. М.: Радио и связь, 1990. 192 с. 3. *Гусятин В.М.* Алгоритм геометрических преобразований изображения в растровых видеосистемах реального масштаба времени // *Авиационно-космическая техника и технологии.* Харьков, 1998.

*Поступила в редколлегию 12.11.99*

**Гусятин Владимир Михайлович**, канд. техн. наук, доцент кафедры электронных вычислительных машин ХТУРЭ. Научные интересы: теория и практика построения спецпроцессоров растровых графических систем реального времени. Адрес: Украина, 61726, Харьков, пр. Ленина, 14, тел. 40-93-54, 66-61-22.

**Остроушко Андрей Павлович**, аспирант кафедры ЭВМ ХТУРЭ. Научные интересы: растровые графические системы реального времени. Адрес: Украина, 61168, Харьков, пр. Ленина, 14, тел. 40-93-54.

---

УДК 681.325:519.713

*В.И. ХАХАНОВ, Е.В. КОВАЛЕВ, В.В. ХАНЬКО,  
МАСУД МД. МЕХЕДИ*

---

## **СИСТЕМА ГЕНЕРАЦИИ ТЕСТОВ ДЛЯ ПРОЕКТИРОВАНИЯ ЦИФРОВЫХ АВТОМАТОВ В СРЕДЕ VHDL-ACTIVE**

---

Предлагается система генерации тестов для конечных автоматов, описываемых в среде VHDL-Active [1]. Исходная информация об объекте диагностирования задается в виде графа переходов или в формате конструкций языков описания аппаратуры VHDL, Verilog. Приводится описание системы генерации тестов ASFTEST с позиции пользователя. Система может быть полезна для проектировщиков цифровых управляющих автоматов, в качестве средства верификации созданного устройства.

### **1. Постановка задачи**

Современные системы проектирования цифровых устройств, как правило, ориентируются на быстродействующие программируемые интегральные схемы (ПЛИС) – Field Programable Gate Array (FPGA), Complex Programable Logic Device (CPLD). К их достоинствам относятся сравнительно невысокая стоимость, минимальное время проектирования устройства, высокое быстродействие выполнения микроопераций.

Процессу проектирования и синтеза цифрового автомата предшествует достаточно трудоемкая процедура построения или описания модели на языках VHDL, Verilog. В качестве альтернативного способа описания цифрового автомата предлагается визуальный интерфейс создания рисунков графа переходов или структуры примитивных элементов. Далее выполняется компиляция внутренних структур данных в целях моделирования и верификации цифрового проекта. Однако для упомянутой процедуры необходимы тестовые последовательности, генерация которых представляет достаточно сложную задачу. В зависимости от сложности и функций проекта применяются несколько основных методов генерации тестов: исчерпывающий, псевдослучайный, алгоритмический.

Первый ориентирован на полную проверку цифрового устройства на всех  $2^{n+m}$  входных наборах ( $n$  – число входных переменных,  $m$  – количество внутренних состояний). Метод прост в реализации, имеет 100%-ное качество покрытия неисправностей, но требует значительных временных затрат в процессе тестирования устройства.

Второй метод использует псевдослучайные генераторы (сигнатурные анализаторы) для получения тестовых последовательностей. Он имеет максимальное быстродействие, но качество теста может быть неприемлемо низким для последовательностных схем.