

ЛОГИЧЕСКИЕ МЕТОДЫ В ОБЛАСТИ МАШИННОГО ЗРЕНИЯ

В.И.Левин, Е.З.Перельройзен

1. Постановка задачи

В последние два десятилетия ведутся интенсивные исследования по созданию специализированных вычислительных систем, способных воспринимать окружающий мир с помощью собственных "сенсорных" устройств. Возникло новое научное направление - машинное зрение, в котором, в свою очередь, можно выделить: 1) анализ и синтез пространственных сцен (совокупностей трехмерных объектов) [1]; 2) машинную графику [2,3].

Методы анализа и синтеза пространственных сцен применяются при математическом моделировании интегральных роботов [4], создании программного обеспечения сенсорной системы интегрального робота на базе стационарной вычислительной системы [5,6], в архитектурно-строительном проектировании [7], обучающих системах (имитационное моделирование) [8,9], при создании систем визуального контроля роботов-манипуляторов, предназначенных для сборки сложных узлов в машиностроении [10] и т.д. Методы машинной графики используются при автоматизации проектирования в машиностроении [2, 11], в системах управления научным экспериментом [12,13].

Вычислительная система может "видеть", если, помимо телевизионной камеры, центрального процессора и памяти, интерфейсов для передачи видеоданных в память, разработано программное обеспечение для интерпретации данных, которые вводятся в память. В существующих программах зрительного восприятия остро ощущается дефицит формальных методов и преобладают содержательные аспекты. Между тем формальное описание внешней среды, формальный алгоритм ее анализа существенно повысили бы обзорность рассматриваемой задачи, несмотря на всю ее сложность, обеспечили бы нахождение оптимального решения задачи. Таким формальным математическим аппаратом, по нашему мнению, являются бесконечнозначная логика и теория ло-

гических определителей [4]. Логические матрицы и определители, подобно обычным матрицам в теории линейных систем, дают возможность блочного описания внешней среды. В результате становится возможным: 1) представить алгоритм решения задачи в обозримой форме; 2) формально сокращать (минимизировать) алгоритм; 3) дать предварительную оценку сложности алгоритма. С помощью бесконечнозначной логики с ее структурными операциями взятия максимума (дизъюнкция) и минимума (конъюнкция) можно выразить весь "континуум" пространственных взаимоотношений между объектами сцен, т.е. решать задачи, относящиеся к так называемой комбинаторной геометрии восприятия, которая в настоящий момент только начинает создаваться. Развитие формальных методов в области машинного зрения позволяет приблизиться к решению серьезной научной проблемы моделирования объектов и явлений реального мира, их машинного восприятия и интерпретации.

2. Основы бесконечнозначной логики и теории логических определителей

Основные операции бесконечнозначной логики для любого числа вещественных переменных $a_1, a_2, \dots, a_n \in [-\infty, \infty]$ суть

$$a_1 \vee a_2 \vee \dots \vee a_n = \bigvee_{i=1}^n a_i = \max(a_1, a_2, \dots, a_n) - \text{дизъюнкция,}$$

$$a_1 \wedge a_2 \wedge \dots \wedge a_n = \bigwedge_{i=1}^n a_i = \min(a_1, a_2, \dots, a_n) - \text{конъюнкция.}$$

Введенные операции обобщают соответствующие операции в двузначной логике, где $\{a_i\}$ есть множество элементов $[0, 1]$. Основные законы бесконечнозначной логики совпадают с аналогичными законами в двузначном случае [14]. Далее, рассмотрим произвольную квазиматрицу вида

$$A_n = \left\| \begin{array}{cccc} a_{11} & \dots & a_{1m} & 1 \\ \dots & \dots & \dots & \dots \\ a_{n1} & \dots & a_{nm} & 1 \end{array} \right\| = \| a_{ij} \|, \quad M = \sum_{i=1}^n m_i,$$

элементы a_{ij} которой упорядочены в строках согласно $a_{i1} \leq a_{i2} \leq \dots \leq a_{im}$, $i=1, \dots, n$, в то время как отношения элементов раз-

личных строк неизвестны. Упорядочим множество элементов квазиматрицы A_n в порядке убывания их величин: $a^{(1)}, a^{(2)}, \dots, a^{(r)}, \dots, a^{(M)}$. Логический определитель r -го ранга от квазиматрицы A_n (см. [14]) определим как отображение $A_n = \{a_{11}, \dots, a_{1n_1}, \dots, a_{nn_n}\}$ в $a^{(r)}$. Обозначим логический определитель r -го ранга n -го порядка через

$$A_n^r = \left| \begin{array}{c} a_{11} \dots a_{1n_1} \\ \dots \dots \dots \\ a_{n1} \dots a_{nn_n} \end{array} \right|^{(r)} = |a_{ij}|^{(r)}, \quad r=1, 2, \dots, M.$$

Свойства логических определителей приведены в [14]. Для каждой квазиматрицы A_n имеется целое семейство определителей, получаемых при варьировании параметра r . Раскрыть логический определитель — значит указать формулу, выражающую его величину через величины элементов квазиматрицы. Для произвольного определителя n -го порядка ранга r такая формула имеет вид (см. [14]):

$$A_n^r = \bigvee_{\substack{\sum_{s=1}^n i_s = r + (n-1)}} a_{1i_1}^{n-1} a_{2i_2}^{n-2} \dots a_{ni_n}^{n-1},$$

где $a_{ki_k}^{n-k}$ означает, что элемент a_{ki_k} не входит в те конъюнкции, для которых из условия на сумму вторых индексов a_{ij} формально получается $i_k > n_k$. Аппарат логических определителей был впервые применен для блочного описания функционирования автоматов в непрерывном времени (см. [14]). Рассмотрим произвольный многомерный процесс

$$x(t) = (x_1(t), x_2(t), \dots, x_n(t)), \quad (I)$$

в котором компоненты $x_i(t)$ — обычные импульсные последовательности с постоянными начальным и конечным значениями 0 или 1 и между этими значениями последовательность мгновенных изменений ($0 \rightarrow 1$ и $1 \rightarrow 0$), совершаемых в произвольные моменты непрерывного времени.

Обозначим: I — постоянный сигнал 1; 0 — постоянный сигнал 0; I' — изменение сигнала $0 \rightarrow 1$; $0'$ — изменение сигнала $1 \rightarrow 0$; I'_a — I' в момент a ; $0'_a$ — $0'$ в момент a ; $1(a, b)$ — импульс $I'_a 0'_b$ в интервале (a, b) ; $0(a, b)$ — пауза $0'_a I'_b$ в интервале (a, b) . Любой процесс $x_i(t)$ можно записать в виде последовательности им —

пульсов и пауз, в этом случае начальное и конечное постоянные значения сигнала опускаются, а моменты промежуточных изменений сигнала указываются только один раз, например, запись $1(a,b)0(-,c)$ означает последовательность импульса $1(a,b)$ и паузы $0(b,c)$. Для определенности считаем, что $x_1(t)$ начинаются и заканчиваются импульсами

$$\begin{aligned} x_1(t) &= 1(a_{11}, b_{11})0(-, -)1(a_{12}, b_{12}) \dots 1(a_{1m_1}, b_{1m_1}) \\ x_n(t) &= 1(a_{n1}, b_{n1})0(-, -)1(a_{n2}, b_{n2}) \dots 1(a_{nm_n}, b_{nm_n}) \end{aligned} \quad (2)$$

Будем считать процесс $x_1(t)$ входным воздействием некоторой безынерционной схемы, реализующей симметрическую булеву функцию $f(x_1, \dots, x_n)$. Тогда указанный процесс можно трактовать просто как совокупность m импульсов ($m = \sum_{i=1}^n m_i$) из (2), действие которых не зависит от входов, по которым они подаются. В результате эквивалентных преобразований многомерного переключательного процесса (2) получаем эквивалентную воздействию (1), (2) совокупность из m импульсов $1(a^{(r)}, b^{(r)})$, $r = 1, \dots, m$, не привязанных к определенным входам и упорядоченных согласно $a^{(1)} \leq a^{(2)} \leq \dots \leq a^{(m)}$, $b^{(1)} \leq b^{(2)} \leq \dots \leq b^{(m)}$, где

$$a^{(r)} = \begin{vmatrix} a_{11} & \dots & a_{1m_1} \\ \dots & \dots & \dots \\ a_{n1} & \dots & a_{nm_n} \end{vmatrix}^{(r)}, \quad b^{(r)} = \begin{vmatrix} b_{11} & \dots & b_{1m_1} \\ \dots & \dots & \dots \\ b_{n1} & \dots & b_{nm_n} \end{vmatrix}^{(r)}.$$

После полного упорядочения квазиматриц, содержащих моменты начал и окончаний импульсов, можно отыскать выходной процесс в логических элементах по соответствующим формулам из [14]. Так, например, реакция n -входового элемента "И" на воздействия (2) выразится через

$$\begin{aligned} y(t) &= 1(b^{(1)}, a^{(1)}, b^{(1)})0(-, -)1(b^{(2)}, a^{(2)}, b^{(2)}) \dots \\ &\dots 1(b^{(M+1-n)}, a^{(M)}, b^{(M+1-n)}), \end{aligned} \quad (3)$$

реакция n -входового элемента "ИЛИ" на воздействия (2) через

$$\begin{aligned} y(t) &= 1(a^{(1)}, b^{(1)}, a^{(2)})0(-, -)1(a^{(2)}, b^{(2)}, a^{(3)}) \dots \\ &\dots 1(a^{(M-1)}, b^{(M-1)}, a^{(M)})0(-, -)1(a^{(M)}, b^{(M)}). \end{aligned} \quad (4)$$

Реакция n -входного элемента, реализующего фундаментальную симметрическую функцию f_n^p индекса p от n переменных ($f_n^p = 1$, если точно p из n переменных (любых) равны 1) [16] выразится через

$$f_n^p(t) = 1(b^{(1)}_a(p), b^{(1)}_a(p+1))0(-,-)1(b^{(1)}_b(p+1), b^{(2)}_a(p+2)) \dots 1(b^{(M-p+1)}_b(M-p), a^{(M-1)}_a(M-p), a^{(M)}_a(M))0(-,-) \times \\ \times 1(b^{(M-p)}_b(M-p+1), a^{(M)}_a(M-p+1)). \quad (5)$$

3. Комбинаторная геометрия восприятия

Комбинаторная геометрия восприятия – особое направление в области искусственного интеллекта, в котором изучаются пространственные взаимодействия дискретных объектов с точки зрения искусственного наблюдателя (анализ модели внешней среды), а также методы синтеза модели внешней среды. При решении задач рассматриваемого класса с помощью изложенного выше математического аппарата приходится сводить исходную трехмерную задачу к ряду плоских задач, каждая из которых относится к одному из сечений трехмерной внешней среды некоторой плоскостью. Подобное "разложение" широко применяется в области машинного зрения. Восстановление трехмерной структуры по результатам решения плоских задач производится с помощью сплайн-интерполяции [16] или "триангуляционными" методами [17]. Далее, отношения между геометрическими объектами на плоскости сводятся к взаимодействию импульсов в некотором условном многомерном переключающем процессе, являющемся реакцией некоторого автомата, реализованного программно или в виде электронной моделирующей сети. Работу такого автомата легко анализировать с помощью аппарата логических определителей. Информацию о совокупности геометрических объектов на плоскости представляют в виде логических матриц и определителей, делая задачу обозримой и способствуя тем самым оптимизации ее решения.

4. "Проблема скрытых поверхностей"

Проиллюстрируем все вышесказанное на примере одной из основных задач машинного зрения – "проблемы скрытых поверхностей". Последняя заключается в элиминировании элементов изображения, невидимых в данный момент из точки наблюдения [18]. При этом получаем полную информацию о взаимном расположении объектов, что позволяет предсказать возможные изменения картины, открывающейся перед ис-

кусственным наблюдателем при перемещении его во внешней среде, понять различные компромиссы, существующие в пространственном распределении объектов внешней среды. Пусть задано конечное множество $M = \{M_1, \dots, M_n\}$ произвольных трехмерных объектов. Построение наглядного изображения этой совокупности трехмерных объектов сводится 1) к определению для каждого $M_i \in M$ множества точек, принадлежащих M_i и невидимых из точки наблюдения, 2) к получению перспективного изображения совокупности трехмерных объектов на картинной плоскости с учетом невидимости определенных линий.

Покажем, как изложенная задача сводится к анализу переключательных процессов в моделирующем асинхронном автомате, построенном на многовыходовых логических элементах, реализующих фундаментальные симметрические функции x_n^P , и одном многовыходовом логическом элементе "ИЛИ". Алгоритм включает в себя 7 операционных блоков и один логический блок.

Блок 1. Начало алгоритма.

Блок 2. Рассечение исходной трехмерной сцены производится при помощи специально выбираемого множества плоскостей $\{l\}$, $l = 1, 2, \dots, P$, проходящих через одну и ту же прямую L . Цель рассечения — представить заданную трехмерную сцену набором из P плоских сцен. В качестве L выбирается одна из осей координат сцены Ox, Oy или Oz . Угол между соседними плоскостями рассечения выбирают достаточно малым, исходя из требуемой точности представления трехмерной сцены набором P плоских сцен. Варьируя этот угол, можно путем некоторого уменьшения точности понизить сложность выполнения блока 2.

Блок 3. Это блок логического условия, обеспечивающий повторение процедуры анализа плоской сцены для всех плоских сцен, полученных в результате выполнения блока 2. Процедура анализа отдельной плоской сцены задается блоками 4-8.

Блок 4. Аппроксимирует выпуклыми многоугольниками все плоские фигуры рассматриваемой плоской сцены с целью облегчить выполнение операций, предусмотренных последующими блоками. Такая аппроксимация возможна, если указанные фигуры выпуклые. Если же фигура аппроксимируется вогнутым многоугольником, то он разбивается на несколько выпуклых многоугольников. Число ребер и вершины аппроксимирующего многоугольника выбираются в соответствии с требуемой точностью аппроксимации плоской фигуры. Снижая требования к точности, можно уменьшить сложность выполнения блока 4. В результате

выполнения блока 4 получается множество $\{j\}$, $j = 1, \dots, n$, выпуклых многоугольников, заданных координатами своих вершин $(x_{j,1}, y_{j,1}), \dots, (x_{j,m_j}, y_{j,m_j})$, упорядоченных по направлению (или против направления) часовой стрелки.

Блок 5. Положение каждого многоугольника относительно всех других полностью определяется углами наклона двух лучей, выходящих из начала координат и стягивающих данный многоугольник. Точки касания многоугольника со стягивающими его лучами делят стороны многоугольника на две части: "переднюю" и "заднюю". Передняя часть является потенциально видимой — она видна наблюдателю в той мере, в которой не заслонена другими многоугольниками, расположенными ближе к точке наблюдения. Задняя не видна ни при каких обстоятельствах (см. рис. I). Вычисление для j -го ($j=1, 2, \dots, n$) многоуголь-

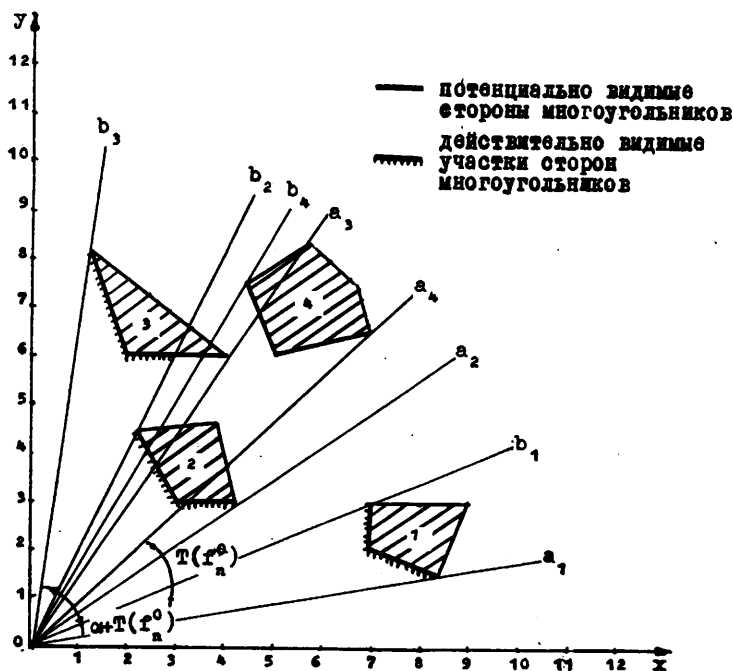


Рис. I

ника углов наклона a_j и b_j лучей, стягивающих этот многоугольник соответственно снизу и сверху, производится таким образом. Для всех вершин k многоугольника ($k=1, 2, \dots, m_j$) вычисляются углы наклона α_{jk} прямых, соединяющих эти вершины с началом координат (точкой наблюдения). Вычисление ведется по формуле:

$$\alpha_{jk} = \arctg \frac{y_{jk}}{x_{jk}}, \quad k = 1, \dots, m_j. \quad (6)$$

Далее искомые углы a_j и b_j определяются соответственно как минимум и максимум углов в (6):

$$a_j = \bigwedge_{k=1}^{m_j} \alpha_{jk}, \quad b_j = \bigvee_{k=1}^{m_j} \alpha_{jk}.$$

Попутно фиксируются сами крайние вершины K_{j*} и K_j^* , которые видны наблюдателю под минимальным и максимальным для данного многоугольника углами

$$K_{j*}: \alpha_{jK_{j*}} = \bigwedge_{k=1}^{m_j} \alpha_{jk}, \quad K_j^*: \alpha_{jK_j^*} = \bigvee_{k=1}^{m_j} \alpha_{jk}.$$

Итак, каждый j -й многоугольник представляется импульсом $1(a_j, b_j)$ (см. рис. 2).

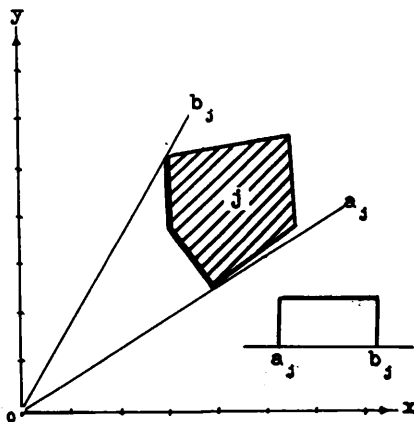


Рис. 2

Блок 6. Выделяет все потенциально видимые стороны j -го многоугольника ($j = 1, 2, \dots, n$) следующим образом. Вначале отыскиваются все потенциально видимые вершины многоугольника. Для этого выберем контрольную вершину $\tilde{K}_j$, отличную от крайних вершин K_{j*} и K_j^* , проведем через $\tilde{K}_j$ и точку наблюдения контрольную прямую, уравнение которой

$$y = \frac{y_{\tilde{K}_j}}{x_{\tilde{K}_j}} \cdot x. \quad (7)$$

Отыскиваем точки пересечения $K_n = (x_n, y_n)$ прямой (7) с различными сторонами многоугольника, не выходящими из взятой вершины $\tilde{K}_j$. При

этом сторона многоугольника (K_p, K_q) , соединяющая вершины $K_p = (x_p, y_p)$ и $K_q = (x_q, y_q)$, описывается следующей системой уравнения:

$$\frac{y - y_p}{x - x_p} = \frac{y_q - y_p}{x_q - x_p}, \quad x_p \leq x \leq x_q, \quad y_p \leq y \leq y_q. \quad (8)$$

В силу выпуклости многоугольника получим единственную точку $K_n = (x_n, y_n)$, в которой прямая (7) пересекается с одной из сторон многоугольника (8). Подсчитываем расстояния точек $\tilde{K}_j$ и K_n от точки наблюдения по формуле

$$\rho_{K_p} = \sqrt{x_p^2 + y_p^2}. \quad (9)$$

При этом если $\rho_{\tilde{K}_j} < \rho_{K_n}$, то контрольная вершина $\tilde{K}_j$ является потенциально видимой. Если же $\rho_{\tilde{K}_j} > \rho_{K_n}$, то вершина $\tilde{K}_j$ не является таковой. В упорядоченном множестве вершин многоугольника $A = \{1, 2, \dots, K_1, \dots, K_2, \dots, n_j\}$, где K_1 и K_2 - крайние вершины, выделяем следующие два упорядоченных подмножества: $A' = \{K_2, K_1 + 1, \dots, K_2\}$ и $A'' = \{K_2, K_2 + 1, \dots, n_j, 1, 2, \dots, K_1\}$. Одно из них есть множество в всех потенциально видимых вершин многоугольника. Это множество выбирается по следующему правилу: если контрольная вершина $\tilde{K}_j$ потенциально видимая, то выбирается подмножество, содержащее $\tilde{K}_j$; если же $\tilde{K}_j$ не потенциально видимая, то выбирается подмножество, не содержащее $\tilde{K}_j$. Сторона многоугольника, соединяющая две потенциально видимые его вершины, является потенциально видимой. Указанное правило позволяет выделить все потенциально видимые стороны многоугольника, т.е. всю его потенциально видимую часть.

Блок 7. Рассматривает множество $\{a_1, \dots, a_n\}$ углов наклона лучей, стягивающих соответствующие многоугольники $j = 1, 2, \dots, n$ снизу, и множество $\{b_1, \dots, b_n\}$ углов наклона лучей, стягивающих эти многоугольники сверху. Производится совместное упорядочение обоих множеств с учетом имеющейся частичной упорядоченности $a_1 < b_1, a_2 < b_2, \dots, a_n < b_n$. Строит панораму обзора плоской сцены 1

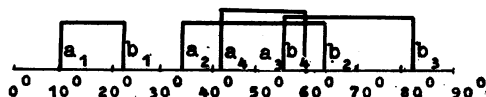


Рис. 3

(панорама на рис.3 соответствует сцене, изображенной на рис.1). Панорама представляет собой систему рассматриваемых совместно секторов обзора всех многоугольников, расположенных

на сцене (т.е. производится угловая сортировка объектов сцены). Панорама естественным образом распадается на последовательные участки, характеризующиеся постоянством числа действующих импульсов. Каждый участок задает соответствующий сектор обзора (a, b) , в пределах которого любой луч, выходящий из точки наблюдения, встречает на своем пути одно и то же число объектов. Эти участки находят с помощью реакций логических элементов, реализующих фундаментальные симметрические функции f_n^i , $i = 0, 1, \dots, n$, по формуле (5). Процесс построения панорамы продолжается до тех пор, пока не будет выполнено равенство

$$\alpha = \sum_{i=1}^k T(f_n^i), \quad k \leq n,$$

где α - суммарная длительность импульсов реакции n -входного элемента "ИЛИ", найденная по формуле (4) (суммарный угол обзора всех многоугольников сцены); $T(f_n^i)$ - суммарная длительность импульсов реакции логического элемента, реализующего функцию f_n^i , $i = 0, 1, 2, \dots, n$ (рис.1). Панораму можно построить и просто в результате последовательного просмотра массива, содержащего упорядоченные раннее моменты начал и окончаний всех n импульсов. Однако использование элементов, реализующих симметрические булевы функции, делает процедуру выделения участков "одинаковой плотности" более обобщимой и, что еще важнее, параллельной, тем самым существенно сокращается время построения панорамы.

Блок 8. Последовательно анализирует обозреваемые участки панорамы обзора сцены и выделяет на каждом из них единственный видимый многоугольник следующим образом. Пусть (a, b) - некоторый участок панорамы, содержащий $M_{(a,b)}$ импульсов многоугольника. Если $M_{(a,b)} = 0$, то участок (a, b) представляет собой просвет в сцене, не содержащий вообще многоугольников. Если $M_{(a,b)} = 1$, то на участке (a, b) имеется всего один многоугольник, который и является видимым. В общем случае, когда $M_{(a,b)} \geq 2$, анализ сводится к проведению через точку наблюдения под некоторым углом δ , где $a \leq \delta \leq b$, контрольной прямой (уравнение прямой $y = x \operatorname{tg} \delta$) и к нахождению точки пересечения этой прямой с различными сторонами многоугольников, находящимися на участке (a, b) . В силу выпуклости многоугольников, контрольная прямая будет пересекаться с каждым многоугольником не более чем в двух точках. Процедура отыскания точек пересечения K_n заданной прямой со сторонами заданного

многоугольника описана в блоке 6. Пусть $K'_{aj} = (x'_{aj}, y'_{aj})$, $K''_{aj} = (x''_{aj}, y''_{aj})$ - пара точек пересечения контрольной прямой с j -м многоугольником участка (a,b) ; $j=1,2,\dots,M(a,b)$. Вычислим по (9) расстояния $\rho_{K'_{aj}}$, $\rho_{K''_{aj}}$ от точки наблюдения. Выделяем минималь-

ное из расстояний $\rho_{K_{na}} = \min_{1 \leq j \leq M(a,b)} (\rho_{K'_{aj}}, \rho_{K''_{aj}})$. Находящийся на этом расстоянии от наблюдателя и выделенный теперь a -й многоугольник и есть единственный многоугольник, видимый на участке (a,b) и закрывающий от наблюдателя все остальные многоугольники на этом участке.

При помощи вышеизложенного алгоритма оценим сложность анализа сцены. Она равна $N = P \cdot N_{пл}$, где P - число плоских сцен, на которые разбивается анализируемая пространственная сцена, а $N_{пл}$ - сложность анализа одной плоской сцены. Оценим $N_{пл}$ в случае большого числа n объектов сцены. Сложность блока 2 не зависит от n , т.е. $N_{пл,2} = C_2$, а блоки 4,5,6 выполняются для различных объектов независимо, так что $N_{пл,4} + N_{пл,5} + N_{пл,6} = O_{4,5,6} \cdot n$. Сложность блока 7 есть сложность совместного упорядочения двух n -элементных множеств, равная $N_{пл,7} = C_7 n \log_2 n$. Сложность блока 8 равна $N_{пл,8} = C_8 \cdot R$, где R - суммарное количество многоугольников по всем участкам панорамы обзора сцены (если некоторый многоугольник проходит через несколько участков, то он учитывается соответствующее число раз). Наибольший интерес представляет оценка для $N_{пл,9}$, получаемая в ситуации "наблюдения в среднем". Пусть n рассматриваемых многоугольников расположены в пределах некоторой части Q сцены фиксированной формы с постоянной линейной плотностью σ . Через σ^2 обозначаем число многоугольников, приходящихся на единицу площади. Очевидно, что площадь $S_Q = C_Q \cdot h^2$, где C_Q - коэффициент формы участка Q , а h - его средняя глубина по множеству всех возможных направлений наблюдения. Тогда $n = C_Q \cdot h^2 \cdot \sigma^2$. Отсюда находим среднее число многоугольников вдоль одного направления наблюдения (усреднение проводим по множеству всех возможных направлений):

$$m = h\sigma = \frac{\sqrt{n}}{\sqrt{C_Q}}.$$

Это число показывает также среднее число многоугольников на одном участке панорамы обзора сцены. Поэтому суммарное количество R многоугольников по всем участкам панорамы обзора равно произведению m на число L участков панорамы (если начала и окончания различных импульсов $L = 2n-1$ не совпадают): $N_{пл,9} = C_9 \cdot R \approx C_9 (2n-1) \sqrt{n}$.

Таким образом, сложность анализа сцены в среднем определяется сложностью выполнения блока 8 и будет $N = O^* P n \sqrt{n}$.

Но можно существенно снизить сложность анализа сцены, дополнительно сортируя объекты сцены по их расстоянию до точки наблюдения, а также используя, кроме вышеупомянутых, логические элементы "И". При этом первые шесть блоков алгоритма остаются без изменений. В блоке 7, помимо сортировки многоугольников по углам, производится их сортировка по расстоянию до точки наблюдения. Соответствующая процедура такова. Опшем для каждого из n много-

угольников значение функции $L_j = \bigwedge_{k=1}^{m_j} (x_{jk} + y_{jk})$, $j = 1, 2, \dots, n$,

здесь m_j - число вершин j -го многоугольника; (x_{jk}, y_{jk}) - координаты k -й вершины j -го многоугольника. Пронумеровав объекты

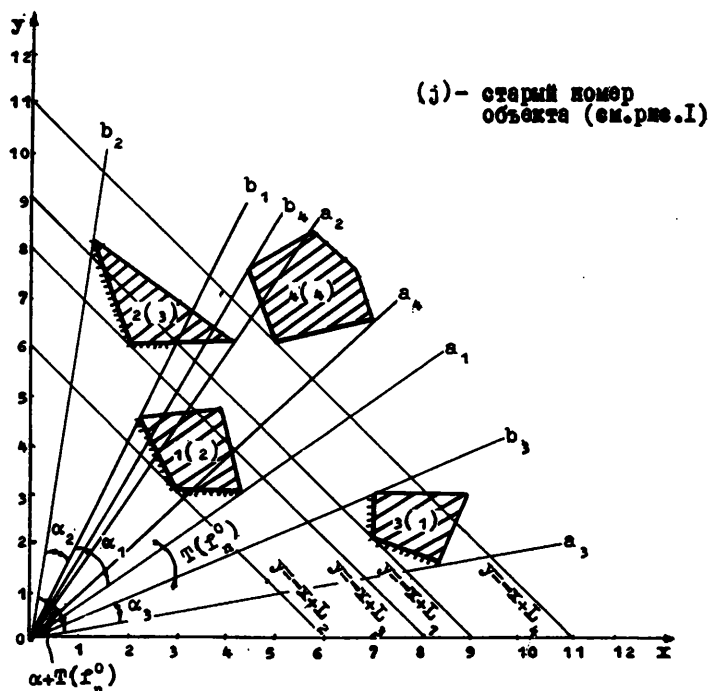


Рис. 4

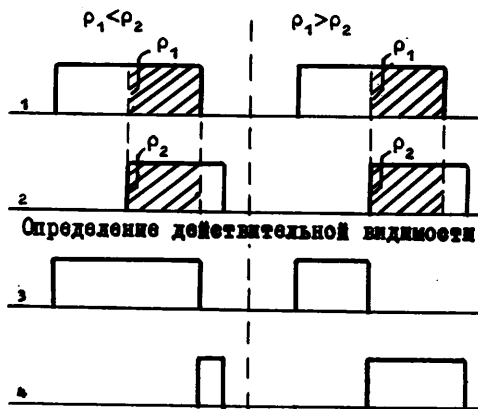


Рис. 5

гоугольниками сцены, чьи моменты начал или окончаний соответствующих им импульсов находятся в пределах импульса j -го многоугольника, определяется путем вычисления реакций двухходовых элементов "И" по формуле (3) при $n=2$. Такой метод справедлив потому, что углы обзора многоугольников, "конкурирующих" за видимость из начала координат, обязательно имеют общую часть. Для перекрывающихся потенциально видимых участков многоугольников действительная видимость определяется в ходе выполнения блока 8, как изложено выше. Чтобы определить действительную видимость, необходимо скорректировать длительности импульсов многоугольников (рис.5). Для этого сначала вычисляется взаимодействие первого многоугольника, затем - второго (при условии, что после предыдущего шага второй многоугольник не закрыт полностью первым), т.е. импульс, ему соответствующий, имеет ненулевую длительность и т.д. до тех пор,

пока $\alpha > \sum_{j=1}^k \alpha_j$. Здесь α_j - угол обзора j -го многоугольника, соответствующий его действительно видимому участку. К моменту дости-

жения равенства $\alpha = \sum_{j=1}^k \alpha_j$ вычислены взаимодействия каждого из k первых многоугольников с другими многоугольниками сцены (см. рис.5). Согласно изложенной выше модели, оценки сложности анализа сцены будут $k \cdot n \sim n$. Таким образом, сложность блока 8 есть $M_{пл,8} = C''_8 \cdot n$, а сложность всего анализа сцены будет определяться

в порядке возрастания их L_j (рис.4), получим овертюрку многоугольников по их удаленности от начала координат (точки наблюдения). Эта процедура позволяет сократить число рассматриваемых участков панорамы обзора сцены с $2n-1$ до $\sqrt{n}$ (теперь каждому участку соответствует импульс определенного многоугольника без дробления). Взаимодействие j -го многоугольника с другими мно-

теперь сложностью блока 7 с учетом дополнительных затрат на сортировку объектов по их расстоянию от точки наблюдения

$$N = O_1 P n \log_2 n. \quad (10)$$

Как видно из (10), алгоритм дает возможность эффективно проводить анализ сцен, содержащих большое число объектов.

Л и т е р а т у р а

1. ДЖАЙН Р., АГТАРВАЛ Д. Машинный анализ сцен, состоящих из объектов криволинейных очертаний. - Труды Ин-та радиоинженеров, 1979, т.67, №6, с.121-130.
2. ПРИНС М. Машинная графика и автоматизация проектирования. - М.: Сов. радио, 1975.-230 с.
3. НЬЮМЕН У., СПРУЛЛ Р. Основы интерактивной машинной графики. - М.: Мир, 1975.-573 с.
4. ОХОЦИМСКИЙ Д.Е., ПЛАТОНОВ А.К., ПРЯНИЧНИКОВ В.Е. Методика моделирования робота, перемещающегося в пространственной среде. - Изв. АН СССР. Техн. кибернетика, 1980, №1, с.46-54.
5. Интегральные роботы. /Под ред. Г.Е.Поздняка. Вып. 1. - М.: Мир, 1973.-421 с.
6. Интегральные роботы /Под ред. Г.Е.Поздняка. Вып. 2. - М.: Мир, 1975.-526 с.
7. ЧУМАКОВА Л.А. Методы сокращения перебора при решении некоторых задач автоматизации обработки графической информации. - Изв. АН СССР. Техн. кибернетика, 1975, № 2, с.62-73.
8. GOLDSTEIN R., NAGEL R. Three-dimensional visual simulation.- Simulation, 1971, v.16, N 1, p.25-30.
9. RAJKE R. Compu-Scene-modular approach to day-night computer image simulation.- AIAA Visual and Motion Simul. Conf.Proc., Dayton, Ohio, 1976, p.101-119.
10. NITZAN D., AGIN G. Fast methods for finding object outlines.- Computer. graphics and image processing, 1979, N 9, p.22-39.
11. IBM Systems Journal, 1968, v.7, N 3,4. Special issue: Interactive graphics in data processing, with numerous references.
12. FRANKLIN R., LEWIS H. Three-dimensional graphic display of discrete spatial data by prism maps. - Computer graphics, 1979, v.12, N 3, p.70-75.
13. LEE M., DI MARCO R. Computer graphics with hidden-surface-GRAPHS.- Computer and graphics, 1978, v.3, N 2, p.85-91.
14. ЛЕВИН В.И. Логические определители и автоматы с непрерывным временем. I, II, III. - Изв. АН СССР, Техн. кибернетика, 1977, № 3, с.113-122; №4, с.124-134; №5, с.134-142.
15. ЛЕВИН В.И. Динамические процессы в схемах с симметрическими функциями. - Автоматика и вычислительная техника, 1977, №6, с.8-11.

16. WU Sheng-Chuan, ABEL I., GREENBERG D. An interactive computer graphics approach to surface representation. - Commun ACM, 1977, v.20, N 10, p.703-712.

17. FUCHS R., KEDEN Z., UZELTON S. Optimal surface reconstruction from planar contours. - Commun ACM, 1977, v.20, N 10, p. 693-702.

18. SUTHERLAND I., SPROULL R., SCHUMACKER R. A characterization of ten hidden-surface algorithms.- Computing Surveys, 1974, v.6, N 1, p.1-55.

Поступила в ред.-изд. отд.

5 мая 1980 года