МИНОБРНАУКИ РОССИИ

Федеральное государственное бюджетное образовательное учреждение высшего образования
«Юго-Западный государственный университет»
(ЮЗГУ)

Кафедра программной инженерии

УТВЕРЖДАЮ:
Проректор по учебной работе
__________ О.Г. Локтионова
«28» 10 2020 г.

МЕТОД ДИНАМИЧЕСКОГО ПРОГРАММИРОВАНИЯ

Методические указания для выполнения лабораторной работы по дисциплине «Дискретная математика» для студентов направления подготовки 09.03.04 Программная инженерия

Курск - 2020

УДК 519.8

Составитель: Р.А. Томакова

Рецензент
Кандидат технических наук, доцент *А.В. Малышев*

Метод динамического программирования: методические указания для выполнения лабораторной работы по дисциплине «Дискретная математика» для студентов направления подготовки 09.03.04 Программная инженерия/ Юго-Зап. гос. ун-т; сост. Р.А. Томакова. Курск, 2020. 19 с.

Составлены в соответствии с федеральным государственным образовательным стандартом высшего образования направления подготовки 09.03.04 Программная инженерия и на основании учебного плана направления подготовки 09.03.04 Программная инженерия.

В методических указаниях представлены основные понятия и свойства метода динамического программирования, необходимые для выполнения лабораторной работы по дисциплине «Дискретная математика». Сформулированы требования для ее выполнения, разобраны примеры выполнения заданий, приведены вопросы к защите.

Предназначены для студентов, обучающихся направления подготовки 09.03.04 Программная инженерия (профиль «Разработка программно-информационных систем») всех форм обучения.

Текст печатается в авторской редакции

Подписано в печать 28.10.20 . Формат 60×84 1/16.
Усл. печ. л. 2,2 . Уч.- изд. л. 2,0. Тираж 25 экз. Заказ. 1390. Бесплатно.
Юго-Западный государственный университет.
305040, г. Курск, ул. 50 лет Октября, 94.

ЛАБОРАТОРНАЯ РАБОТА
МЕТОД ДИНАМИЧЕСКОГО ПРОГРАММИРОВАНИЯ

Цель работы:

1) приобретение навыков построения математических моделей для задач динамического программирования;

2) изучение метода динамического программирования, применяемого к нахождению оптимальных процессов управления;

3) освоение эффективной вычислительной процедуры нахождения оптимального решения;

4) оценка возможностей практического применения в области автоматических систем управления.

ЗАДАНИЕ

1. В соответствии с вариантом задания составить таблицу изменения параметров управления на плоскости.

Дискретные состояния объекта управления представить узлами сетки, пронумерованными по схеме, изображённой на рисунке 1.

В каждом узле сетки возможное значение изменения двух управлений заполняется студентом самостоятельно, учитывая, что n - порядковый номер студента в списке журнала группы, N - номер студенческой группы.

2. Провести расчеты оптимального управления на каждом шаге вычислений.

3. Рассчитать оптимальное управление для всего процесса.

4. Представить оптимальное управление процесса в виде графа.

5. Оценить достоинства и недостатки метода динамического программирования. Указать области применения данного метода для решения технических задач.

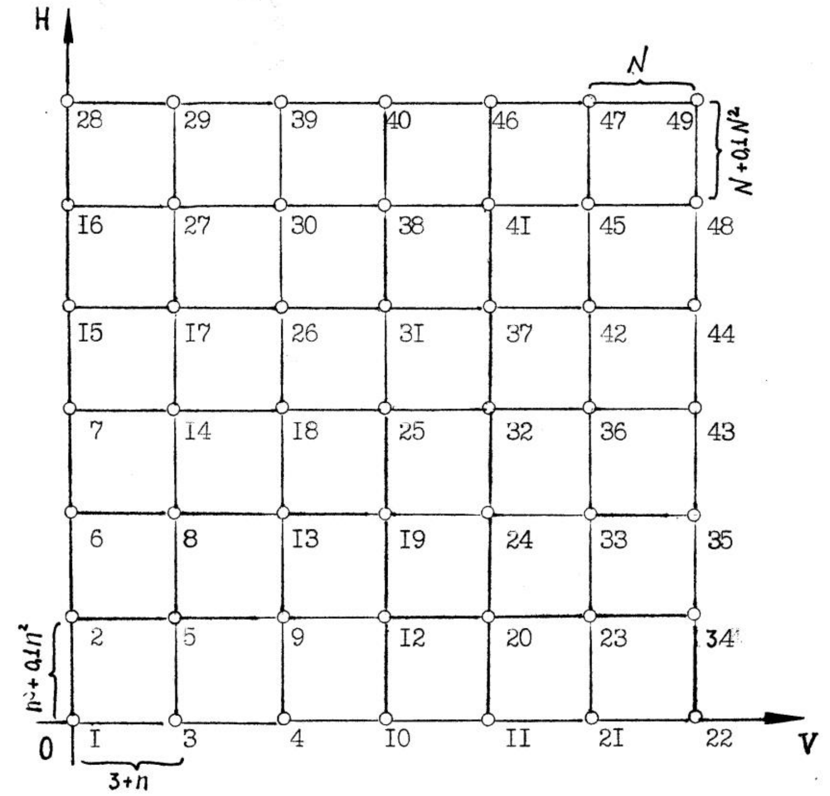

Рисунок 1. Таблица изменения параметров управления на плоскости в соответствии с вариантом задания

Таблица 1

x	u	x' = T(x,u)	Q(x,u)	f (x')	F (x,u)	f (x)	u ⋆
	u	T(x ,u)	Q(x ,u)	f (T(x,u)	F (x ,u)	f (x)	u
x	. ..	. . . .	. . .	. . .	. . .		
	u	T(x ,u)	Q(x ,u)	f (T(x,u)	F (x ,u)		
x	. .	. . . .	. . . .	. . . . . .	. . . . . .	. . .	. . .

ОСНОВНЫЕ ТЕОРЕТИЧЕСКИЕ ПОЛОЖЕНИЯ МЕТОДА ДИНАМИЧЕСКОГО ПРОГРАММИРОВАНИЯ

Динамическое программирование представляет математический метод оптимизации решений, приспособленный к многошаговым (или многоэтапным) операциям. Некоторые операции разделяются на шаги естественно. Например, при планировании хозяйственной деятельности группы предприятий естественным шагом является хозяйственный ход. В других операциях разделение на шаги приходится вводить искусственно. Например, процесс вывода ракеты на космическую орбиту условно можно разбить на этапы, каждый из которых занимает временной отрезок Δt.

Процессы, которые мы будем рассматривать, являются управляемыми, т.е. на каждом шаге принимается какое-то решение, от которого зависит успех данного шага и операции в целом.

ОБЩАЯ ПОСТАНОВКА ЗАДАЧИ ДИНАМИЧЕСКОГО ПРОГРАММИРОВАНИЯ

Пусть имеется некоторая физическая система *S*, которая с течением времени меняет свое состояние, т. е. в системе *S* происходит какой-то процесс. Мы можем управлять этим процессом, т.е. тем или другим способом влиять на состояние системы. Такую систему *S* будем называть ***управляемой системой,*** а способ нашего воздействия на нее **-** ***управляющим воздействием*** или ***управлением*** *U*.

Под управлением и будем понимать совокупность величин, векторов или функций, характеризующих управление. Мы хотим так управлять процессом, чтобы «выигрыш» *W* от управления был максимальным:

$$W = w(u) \rightarrow \max . \qquad (1)$$

Требуется найти такое управление (оптимальное) *U*=*u*, при котором выигрыш максимален:

$$W_{\max} = \max_{u}\{w(u)\} \ \} , \qquad (2)$$

Таким образом, поставлена общая задача оптимизации управления физической системы.

Для того, чтобы общая задача оптимизации была поставлена полностью, необходимо указать начальное состояние системы S_0 и конечное S_w.

Общая задача оптимального управления формулируется так:

***Из множества возможных условий U найти такое оптимальное управление U , которое переводит данную физическую систему S* из *начального состояния* So, *принадлежащее S, в конечное* S$_w$*, принадлежащее S, чтобы при этом выигрыш W обращался в* max.**

Дадим процессу управления геометрическую: интерпретацию. Введем понятие *фазового пространства*.

Состояние системы *S* всегда можно описать с помощью численных параметров, таких как:

- координаты тела и его скорость;
- количество средств, вложенных в отрасль производства; и др.

Эти параметры мы будем называть фазовыми координатами системы *S*, а состояние системы изображать точкой с этими координатами в некотором условном фазовом пространстве.

Если состояние системы характеризуется одним параметром *E*, то фазовое пространство будет одномерным и представляет собой участок оси абсцисс

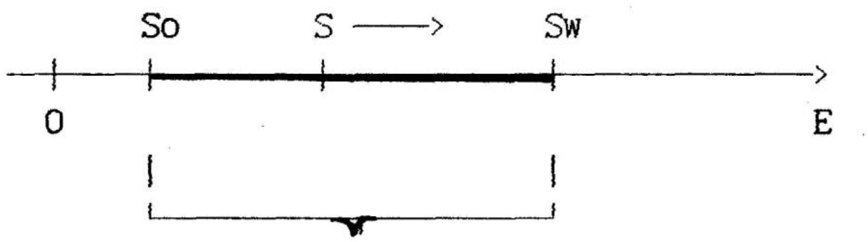

Рисунок 2 . Область возможных состояний системы, фазовое пространство которой характеризуется одним параметром

Если состояние системы характеризуется двумя параметрами E_1, E_2, то фазовое пространство будет двумерным, а процесс изображается перемещением точки S из $S_0 \in S$ в $S_w \in S$ по определенной траектории на фазовой плоскости.

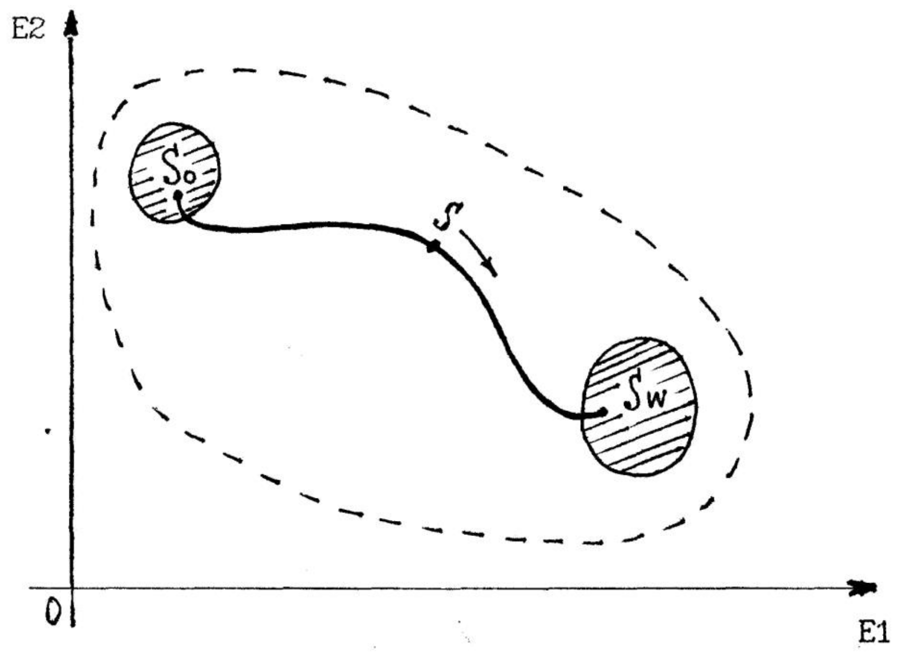

Рисунок 3. Область возможных состояний системы, фазовое пространство которой характеризуется двумя параметрами

Если состояние системы характеризуется тремя координатами E_1, E_2, E_3, то фазовым пространством будет трехмерным пространством, а управляемый процесс демонстрируется как перемещение точки вдоль пространственной кривой.

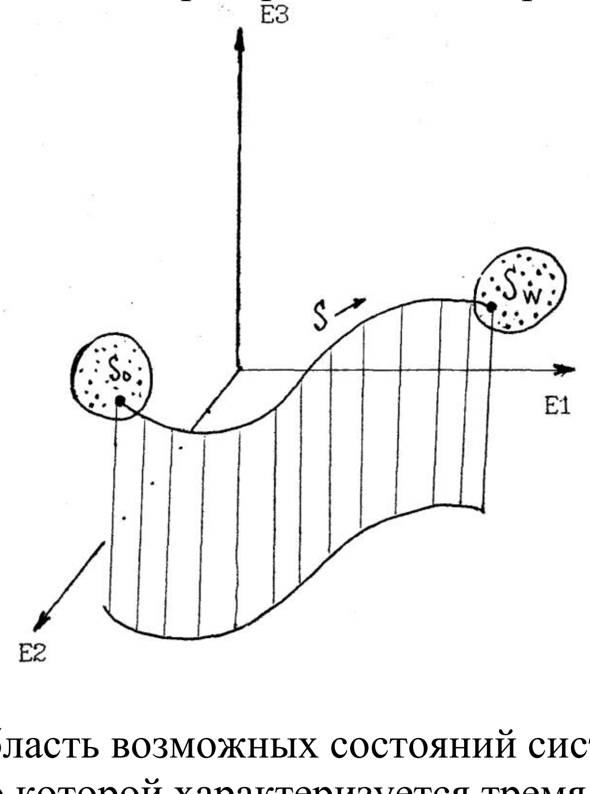

Рисунок 4. Область возможных состояний системы, фазовое пространство которой характеризуется тремя параметрами

Если состояние системы S описывается п параметрами $E_1, E_2, ..., E_n$, мы будем говорить о точке S в n-мерном пространстве и о ее перемещении из области $S_0 \in S$,в область $S_w \in S$. Поставленную задачу оптимального управления будем решать методом динамического программирования.

Процедура построения оптимального управления методом динамического программирования разделяется на две стадии: ***предварительную и окончательную.***

На ***предварительной стадии*** оптимизация производится по шагам в обратном порядке: от последнего шага к первому.

Для каждого ***шага определяется условное оптимальное управление,*** зависящее от состояния системы S (достигнутого в результате предыдущих шагов) и ***условным оптимальный выигрыш*** на всех оставшихся шагах, начиная с данного, также зависящий от состояния *S*.

На ***окончательной стадии*** определяется для каждого шага ***безусловное оптимальное управление.*** Окончательная оптимизация производиться по шагам, но в естественном порядке.

В основе поэтапной процедуры лежит ***принцип оптимальности***, заключающийся в том, что:

каково бы ни было состояние S системы в результате какого-то числа шагов, мы должны выбрать управление на ближайшем шаге так, чтобы оно, в совокупности с оптимальным управлением на всех последующих шагах, приводило к max выигрышу на всех оставшихся шагах, включая данный.

Обозначим $W_i(S)$ условный оптимальный выигрыш на всех последующих шагах, начиная с i-го и до конца; он достигается при оптимальном управлении на всех этих шагах и равен max выигрышу, который можно получить на всех этих шагах вместе, если вначале система в состоянии S.

$W_i(S)$- ***условный оптимальный выигрыш.***

Обозначим через $U_i(S)$ ***условное оптимальное управление*** на i-m шаге.

Рассмотрим i-й шаг процесса управления.

На i-м шаге выигрыш W_i зависит как от состояния системы S, так и от применяемого управления U_i.

$$W_i = W_i(S, U_i). \qquad (3)$$

Под влиянием управления U_i на i-м шаге система из состояния S (в котором она была перед этим шагом) перейдет в новое состояние S'. Это новое состояние будет зависеть от прежнего состояния S и примененного управления U_i:

$$S' = Y_i(S,U_i). \qquad (4)$$

Запишем выигрыш, который мы получим на всех шагах, начиная с i-го:

$$W_i(S,U_i) = W_i(S,U_i) + W_{i+1}(S') \qquad (5)$$

или

$$W_i(S,U_i) = W_i(S,U_i) + W_{i+1}(Y_i(S,U_i)) \qquad (6)$$

В соответствии с принципом оптимальности, мы должны выбрать такое управление $\tilde{U}_i = u_i$, при котором величина $W_i \to \max$.

$$\tilde{W}_i(S) = \max_{U_i}\{W_i(S,U_i) + W_{i+1}(\varphi_i(S,U_i))\} \qquad (7)$$

То управление, при котором этот max достигается, называется ***условное оптимальное управление*** на i-м шаге, а сама величина $W_i(S)$ - условный оптимальный выигрыш на всех шагах, начиная с i-го.

Формула (7) представляет собой ***основное функциональное уравнение динамического программирования*** (*уравнение Белмана*).

Она позволяет определить функцию $W_i(S)$, если известна $W_{i+1}(S)$. Условный оптимальный выигрыш на последнем шаге $W_m(S)$ определяется так;

$$W_m(S) = \max_{U_m}\{W_m(S,U_m)\}. \qquad (8)$$

Максимум берется не по всем возможным управлениям U_m на m-шаге, а только по тем, которые приводят систему в область конечных состояний S_w

$Y_m(S,U_m) \in \tilde{S}_w$.

Построим цепочку условных оптимальных управлений.

Зная $W_m(S)$, можно по общей формуле, полагая i+l=m, найти функцию $W_{m-1}(S)$ и соответствующее оптимальное управление $U_{m-1}(S)$, затем $W_{m-2}(S)$ и $U_{m-2}(S)$ и т. д., вплоть до 1-го шага $W_1(S)$, $U_1(S)$.

Предварительная оптимизация закончена, на этом этапе определены условный оптимальный выигрыш и условное оптимальное для каждого шага.

2 стадия. Найдем ***безусловное оптимальное управление*** $U = (U_1, U_2, \ldots, U_m)$.

Пусть исходное состояние S_0 нам полностью известно. Найдем $W_{\max} = W_1(S_0)$, тогда $U = U_1(S_0)$.

Далее, зная состояние системы S_0 и управление U_1, можем найти состояние системы S_1^* после 1-го шага $S_1 = Y_1(S_0, U_1)$

$$S_0 \to U_1(S_0) \to S_1^* \to U_2(S_1^*) \to \cdots \to S_{m-1}^* \to U_m(S_{m-1}^*) \to S_m$$

Определим оптимальные управления на каждом шаге и найдем, состоящие из них оптимальные управления $U = (U_1, U_2, \ldots, U_m)$.

При решении любой задачи динамического программирования следует придерживаться установленного порядка действий:

1. Выбрать способ описания процесса, т.е. параметры, характеризующие состояние системы, фазовое пространство и способ деления операции на шаги.

2. Записать выигрыш на i-м шаге, в зависимости от состояния системы S вначале этого шага и управления U_i :

$$W_i = W_i(S, U_i).$$

3. Записать для i-го шага функцию, выражающую измененное состояние системы от $S \to S'$ под влиянием U_i :

$$S' = Y_i(S, U_i) \ .$$

4. Записать основное функциональное уравнение (7), выражающее $W_i(S)$ через $W_{i+1}(S)$:

$$W_i(S, U_i) = \max_{U_i}\{W_i(S, U_i) + W_{i+1}(Y_i(S, U_i))\}.$$

5. Найти функцию $W_m(S)$ (условный оптимальный выигрыш.) для последнего шага

$$W_m(S) = \max_{U_m}\{W_m(S, U_m)\}$$

6. Зная $W_m(S)$, и пользуясь уравнением (7) при конкретном виде функций $W_i = W_i(S, U_i)$ и $Y_i(S, U_i)$ найти $W_{m-1}(S)$, $W_{m-2}(S), \ldots, W_1(S)$ и соответствующие им условные оптимальные управления $U_{m-1}(S)$, $U_{m-2}(S), \ldots, U_1(S)$.

7. Если начальное состояние S_0 известно, то найти max выигрыш $W_{\max} = W_1(S_0)$ и далее безусловные оптимальные управления

$$S_0 \to U_1(S_0) \to S_1^* \to U_2(S_1^*) \to \cdots \to S_{m-1}^* \to U_m(S_{m-1}^*) \to S_m.$$

ПРИМЕР РЕАЛИЗАЦИИ МЕТОДА ДИНАМИЧЕСКОГО ПРОГРАММИРОВАНИЯ ДЛЯ РЕШЕНИЯ ЗАДАЧИ О НАБОРЕ ВЫСОТЫ И СКОРОСТИ ЛЕТАТЕЛЬНЫМ АППАРАТОМ

Пусть самолет (или другой летательный аппарат), находящийся на высоте H_0 и имеющий скорость V_0, должен быть поднят на заданную высоту H_w, а скорость его доведена до заданного значения V_w (буквой w мы будем отмечать конец процесса). Известен расход горючего, необходимый для подъема аппарата с любой высоты H_i на любую другую H_{i+1},при неизменной скорости V; известен также расход горючего, потребный для увеличения скорости от любого значения V_i до V_{i+1}, при неизменной высоте Н.

Требуется найти оптимальный режим набора высоты и скорости, при котором общий расход горючего будет минимальным.

Решение будем строить следующим образом. Предположим, что весь процесс набора высоты и скорости разделен на ряд последовательных шагов (этапов) и за каждый шаг самолет увеличивает только высоту или только скорость. Будем изображать состояние самолета точкой S на плоскости VOH (рис. 5), где абсцисса-скорость самолета, а ордината - его высота.

Очевидно, существует множество возможных управлений - множество траекторий, по которым можно перевести точку S из S_0 в S_w.

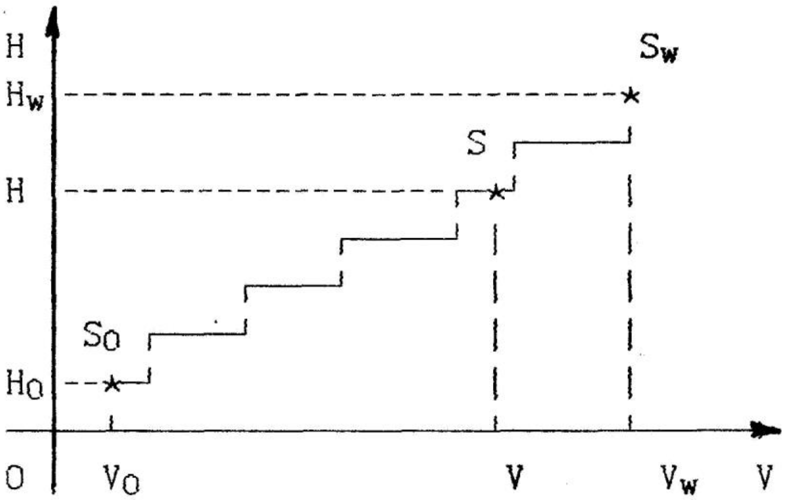

Рисунок 5. Траектория движения

Из всех этих траекторий нужно выбрать ту, на которой расход горючего будет минимальным. Будем решать задачу методом динамического программирования. Для этого разделим интервал скоростей $V_w - V_0$ на n_1 равных частей:

$$V = \frac{V_m - V_0}{n_1},$$

а интервал высот $H_w - H_0$ - на n_2 равных частей:

$\Delta H = \dfrac{H_m - H_0}{n_2}$ (рис.6).

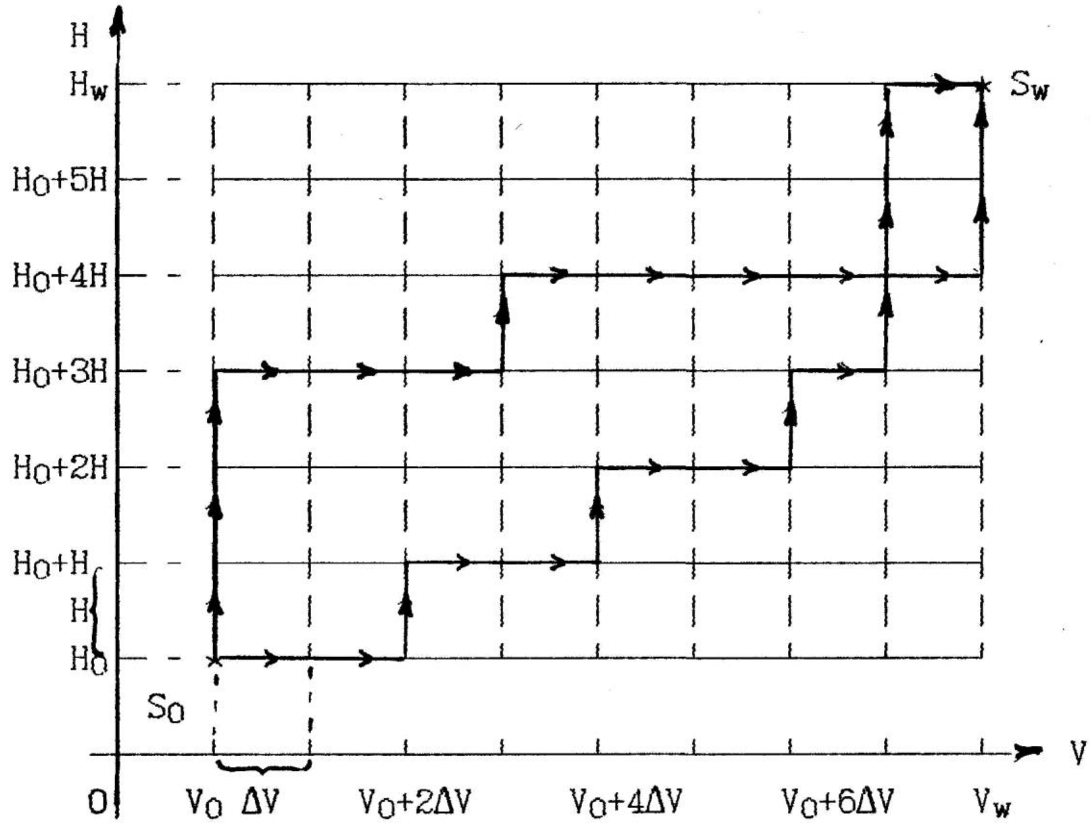

Рисунок 6. Деление интервала скоростей и высоты

Число частей n_1, n_2 принципиального значения не имеет и может быть выбрано, исходя из требований точности решения задачи. Так как за каждый шаг мы можем менять только высоту или только скорость, то общее число m шагов будет:

$$m = n_1 + n_2.$$

Например, для случая, изображенного на рис.6, n_1 = 8, n_2 = 6, m =14. Любая траектория, переводящая точку S из S_0 в S_w, состоит из 14 шагов или этапов.

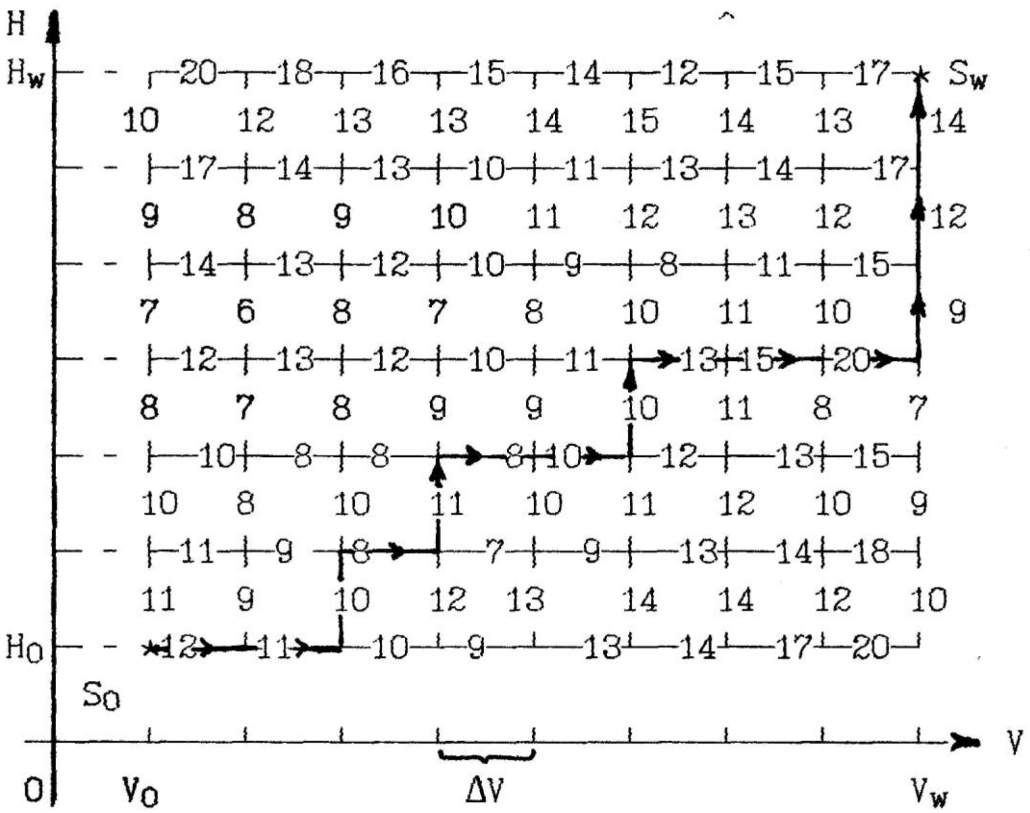

Рис. 7. Траектория управления из состояния S_0 в состояние S_w

Чтобы оптимизировать управление процессом набора высоты и скорости (т.е. выбрать ту траекторию, на которой расход горючего минимален), надо знать расход на каждом шаге (горизонтальном или вертикальном участке траектории). Предположим, что эти расходы заданы (см. рис.7). На каждом отрезке записан расход горючего в условных единицах. Любой траектории, переводящей S из S_0 в S_w, соответствует вполне определенный расход горючего, равный сумме чисел, написанных на отрезках. Например, траектория, помеченная стрелками на рис.7, дает расход горючего:

W=12+11+10+8+11+8+10+10+13+15+20+9+12+14=163.

Нужно из всех траекторий выбрать ту, для которой расход горючего минимален. Решим задачу методом динамического программирования. Процесс состоит из 4 шагов; будем оптимизировать каждый шаг, начиная с последнего. Конечное состояние самолета (точка S_w) нам задано; 14-й шаг непременно должен привести нас в эту точку. Посмотрим, откуда мы можем переместиться в точку S_w за один шаг, т.е. каковы возможные состояния самолета после последнего 13-го шага?

Рассмотрим отдельно правый верхний угол прямоугольной сетки (рис. 8) с конечной точкой S_w. В эту точку можно за один шаг переместиться из двух соседних точек: B_1, B_2, причем из каждой - только одним способом, так что выбора условного управления на последнем шаге нет - оно единственно. Если предпоследний шаг привел в точку B_1, мы должны двигаться по горизонтали (набрать скорость) и тратить 17 единиц горючего; если в точку B_2 - идти по вертикали (набрать высоту) и тратить 14 единиц. Запишем эти минимальные (в данном "случае"- неизбежные) расходы в специальных кружках, которые поставим в точках B_1 и B_2 (рис. *8*). Запись 17 в кружке у B_1 означает: если мы пришли в B_1, то минимальный расход горючего, переводящий нас в точку S_w, равен 17 единицам. Аналогичный смысл имеет запись 14 в кружке у точки B_2.. Оптимальное управление, приводящее к этому расходу, помечено в каждом случае стрелкой, выходящей из кружка. Стрелка указывает направление, по которому мы должны двигаться из данной точки, если в результате предыдущей деятельности оказались в ней.

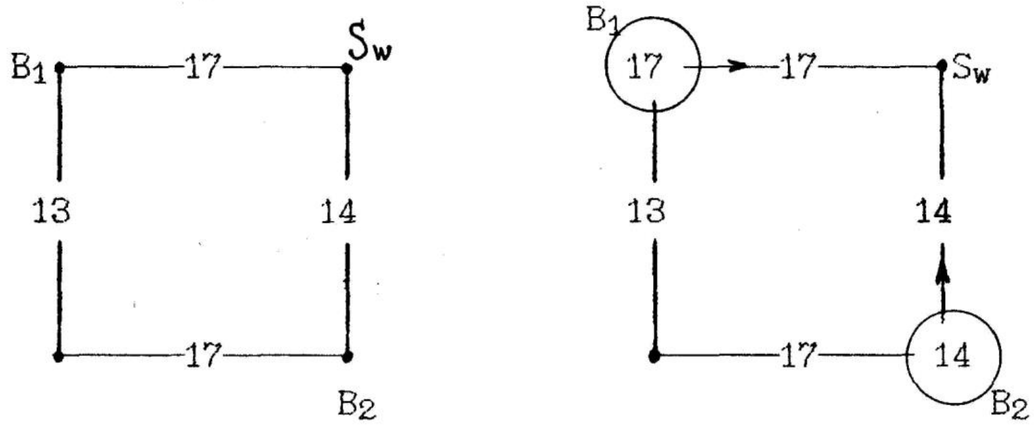

Рис. 8. Возможные способы перемещения в S_w

Таким образом, условное оптимальное управление на последнем, четырнадцатом шаге найдено для любого (B_1 или B_2) исхода 13 шага. Для каждого из этих исходов найден условный минимальный расход горючего, за счет которого можно переместиться из данной точки в S_w.

Перейдем к планированию предпоследнего, 13-го шага. Для этого рассмотрим все возможные результаты пред-предпоследнего, 12-го шага.После этого шага можем оказаться только в одной из точек C_1, C_2, C_3 (рис. 9). Из каждой такой точки должны найти оптимальный путь в S_w и соответствующий этому пути минимальный расход горючего.

Если оказались в точке C_1, то выбора нет: мы должны перемещаться по горизонтали и тратить 15+17=32 единицы горючего. Этот расход запишем в кружке при точке C_1, а оптимальное (в данном случае единственное.) управление из точки C_1 снова пометим стрелкой.

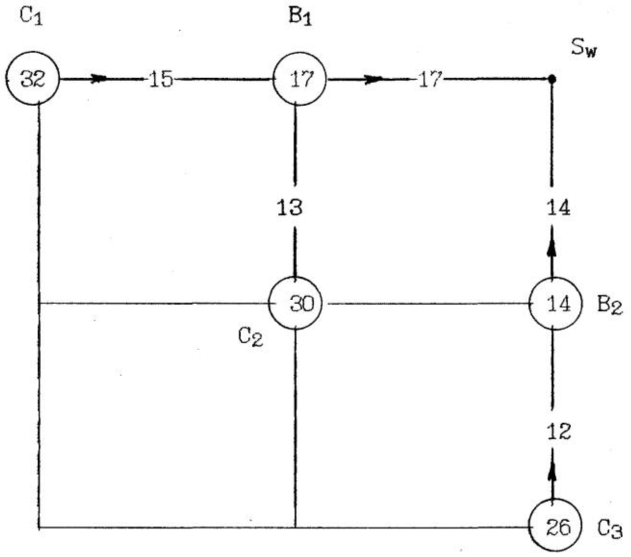

Рис.9. Условное оптимальное управление для 13 шага

Для точки C_2 выбор есть: из нее можно идти в S_w либо через B_1, либо через B_2. В первом случае израсходуем 13+17=30 единиц горючего, во втором 17+14=31 единицу. Значит, оптимальный путь из C_2 в S_w начинается вертикальным участком (отметим это вертикальной стрелкой), а минимальный расход горючего равен 30 (это число запишем в кружке при точке C_2.).

Наконец, для точки C_3 путь в S_w опять единственный: по вертикали. Обходится он в 12+14=26 единиц; эту величину (26) мы и запишем в кружке при точке C_3 , а стрелкой пометим оптимальное управление.

Таким образом, переходя от точки к точке справа налево и сверху вниз (от конца процесса к его началу), можно для каждой узловой точки рис. 9 выбрать условное оптимальное управление на следующем шаге, т.е. направление, ведущее из данной точки в точку S_w с минимальным расходом горючего, и записать в кружке у данной точки этот минимальный расход.

Чтобы найти в узловой точке оптимальное управление, нужно посмотреть два возможных пути из этой точки: направо и вверх, и для каждого из них найти сумму расходов горючего на этом шаге и минимального расхода горючего на оптимальном продолжении пути, уже построенном для следующей точки, куда ведет данный путь. Из двух путей (вправо и вверх) выбирается тот, для которого эта сумма меньше (если суммы равны, выбирается любой путь). В результате выполнения такой процедуры, из каждой узловой точки (см. рис. 10) проводится стрелка, указывающая условное оптимальное управление, а в кружке записывается минимальная стоимость перехода из этой точки в S_w (условная минимальная стоимость). Рано или поздно процесс заканчивается, дойдя до исходной точки S_0. Из этой точки, как и из любой другой, идет стрелка, указывающая, куда надо из нее перемещаться, а в кружке записан минимальный расход горючего. На этом этап условной оптимизации управления заканчивается.

Начинается завершающий этап безусловной оптимизации - *построение оптимального управления на каждом шаге от первого до последнего.* При этом мы строим оптимальную траекторию точки S, перемещаясь по стрелкам из S_0 в S_w.

На рис. 10 показан окончательный результат такой процедуры - оптимальная траектория отмечена жирными кружками и дополнительными стрелками. Число 139, стоящее у точки S_0, означает минимальный расход горючего $V_{\min}$ меньше которого нельзя получить ни на какой траектории. Таким образом, поставленная задача решена, оптимальное управление процессом найдено.

Оно состоит в следующем:

- на первом шаге увеличивать только скорость, сохраняя неизменной высоту Hо, и довести скорость до $V_0 + \Delta V$;

- на втором и третьем шагах увеличить высоту до $H_0 + 2\Delta H$, сохраняя скорость неизмененной;

- на четвертом, пятом и шестом шагах снова набирать скорость, пока она не станет равной $V_0 + 4\Delta V$;

- на седьмом и восьмом шагах набирать высоту и довести ее до $H_0 + 4\Delta H$;

- на девятом, десятом, одиннадцатом и двенадцатом шагах снова набирать скорость и довести ее до заданного конечного значения V_w;

- на последних двух шагах (тринадцатом и четырнадцатом) набирать высоту до заданного значения H_w.

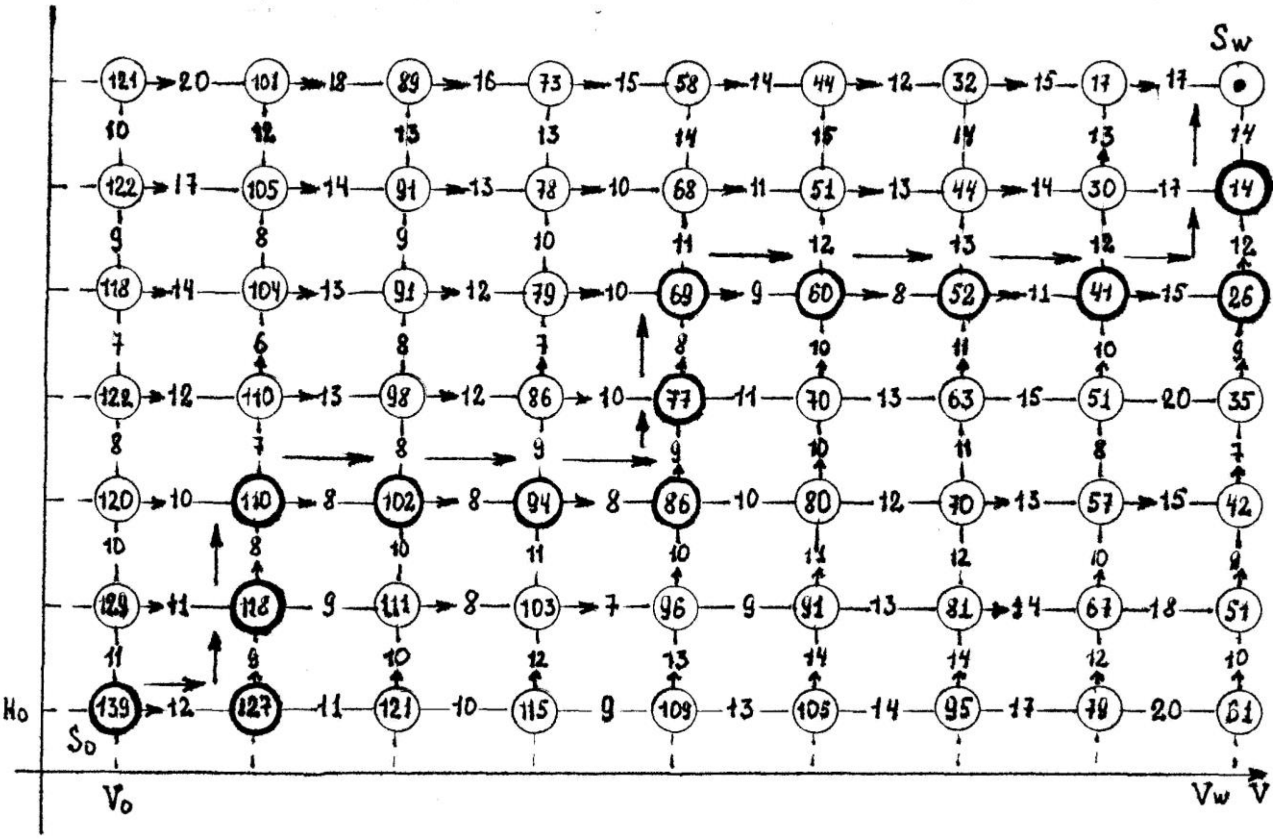

Рисунок 10. Оптимальная траектория управления

Нетрудно на ряде примеров убедиться, что найденное управление действительно является оптимальным и на любой другой траектории расход горючего будет больше (или не меньше).

КОНТРОЛЬНЫЕ ВОПРОСЫ

1. Привести математическую формулировку задачи оптимального управления.
2. Что такое критерий качества управления и его физический смысл?
3. Какое управление называется оптимальным, допустимым управлением?
4. Изложить суть метода динамического программирования.
5. В чем состоят особенности оптимизации управления n-шагового процесса?
6. Записать уравнение Белмана. Что позволяет определить это уравнение?
7. В чем состоят особенности метода динамического программирования?