27. Динамическое программирование

Словосочетание динамическое программирование впервые было использовано в 1940-х годах Р. Беллманом для описания процесса нахождения решения задачи, где ответ на одну задачу может быть получен только после решения задачи, «предшествующей» ей. Вклад Беллмана в динамическое программирование был увековечен в названии уравнения Беллмана, центрального результата теории динамического программирования.

В упрощенной формулировке динамическое программирование представляет собой направленный последовательный перебор вариантов, который обязательно приводит к глобальному максимуму (глобальному минимуму).

Слово «программирование» в словосочетании «динамическое программирование» в действительности к традиционному программированию (написанию кода) почти никакого отношения не имеет и происходит от словосочетания «математическое программирование», которое является синонимом слова «оптимизация». Поэтому слово «программа» в данном контексте скорее означает оптимальную последовательность действий для получения решения задачи. К примеру, определенное расписание событий на выставке иногда называют программой. Программирование в данном случае понимается как допустимая последовательность событий.

Динамическое программирование — раздел математического программирования, совокупность приемов, позволяющих находить оптимальные решения, основанные на вычислении последствий каждого решения и выработке оптимальной стратегии для последующих решений.

Процессы принятия решений, которые строятся по такому принципу, называются многошаговыми процессами. Математически оптимизационная задача строится с помощью таких соотношений, которые последовательно связаны между собой: например, полученный результат для одного года вводится в уравнение для следующего (или, наоборот, для предыдущего) и т. д. Таким образом, можно получить на вычислительной машине результаты решения задачи для любого избранного момента времени и «следовать» дальше. Динамическое программирование применяется не обязательно для задач, связанных с течением времени. Многошаговым может быть и процесс решения вполне статической задачи. Таковы, например, некоторые задачи распределения ресурсов.

Динамическое программирование представляет собой отдельное программирование, основным направлением которого является исследование экстремальных задач. Последние предполагают использование условий целочисленности относительно искомых переменных, а область допустимых решений является конечной.

Поэтому в данном случае применяют модель общей задачи математического программирования с ограничением $x_1, x_2, ..., x_n$ (целочисленны).

Множество реальных задач характеризуются как дискретные. Причиной этому служит физическая неделимость некоторых факторов и объектов расчета. Так, на практике невозможно построить примерно 2,3 завода или принять на работу 1/5 землекопа. Все отраслевые задачи составляются на основе точного количества предприятий или проектных вариантов. Что касается планирования, то здесь используются типовые размеры предприятий, типовые мощности агрегатов. Такие элементы, в свою очередь вносят дискретность в реализацию расчетов.

Существуют так называемые плановые показатели — это годовые, месячные или суточные периоды, которые являются дискретными, раздельными. Каждый из элементов предполагает свое начало и конец.

Дискретное программирование также называют целочисленным. По этому поводу среди ученых возникают разногласия. С одной стороны, как видно, из примеров, в таком программировании действительно используются целые числа, с другой — в целом дискретное программирование не является исключительно целочисленным.

Идеи динамического программирования.

Планируя многошаговый процесс, необходимо выбирать управляющее воздействие на каждом шаге с учетом его будущих последствий на еще предстоящих шагах. Однако, из этого правила есть исключение. Среди всех шагов существует один, который может планироваться без "заглядывания в будущее". Какой это шаг? Очевидно, последний — после него других шагов нет. Этот шаг, единственный из всех, можно планировать так, чтобы он как таковой принес наибольшую выгоду. Спланировав оптимально этот последний шаг, можно к нему пристраивать предпоследний, к предпоследнему — предпредпоследний и т.д.

Поэтому процесс динамического программирования на 1-м этапе разворачивается от конца к началу, то есть раньше всех планируется последний, N-й шаг. А как его спланировать, если мы не знаем, чем кончился предпоследний? Очевидно, нужно сделать все возможные предположения о том, чем кончился предпоследний, $(N$-1)-й шаг, и для каждого из них найти такое управление, при котором выигрыш (доход) на последнем шаге был бы максимален. Решив эту задачу, мы найдем условно оптимальное управление на N-м шаге, т.е. управление, которое надо применить, если (N-1)-й шаг закончился определенным образом.

Предположим, что эта процедура выполнена, то есть для каждого исхода $(N$-1)-го шага мы знаем условно оптимальное управление на N-м шаге и соответствующий ему условно оптимальный выигрыш. Теперь мы можем оптимизировать управление на предпоследнем, (N-1)-м шаге. Сделаем все возможные предположения о том, чем кончился предпредпоследний, то есть $(N$-2)-й шаг, и для каждого из этих предположений найдем такое управление на $(N$-1)-м шаге, чтобы выигрыш за последние два шага (из которых последний уже оптимизирован) был максимален. Далее оптимизируется управление на $(N$-2)-м шаге, и т.д.

Одним словом, на каждом шаге ищется такое управление, которое обеспечивает оптимальное продолжение процесса относительно достигнутого в данный момент состояния. Этот принцип выбора управления, называется принципом оптимальности. Само управление, обеспечивающее оптимальное продолжение процесса относительно заданного состояния, называется условно оптимальное управление на данном шаге.

Теперь предположим, что условно оптимальное управление на каждом шаге нам известно: мы знаем, что делать дальше, в каком бы состоянии ни был процесс к началу каждого шага. Тогда мы можем найти уже не "условное", а действительно оптимальное управление на каждом шаге.

Действительно, пусть нам известно начальное состояние процесса. Теперь мы уже знаем, что делать на первом шаге: надо применить условно оптимальное управление, найденное для первого шага и начального состояния. В результате этого управления после первого шага система перейдет в другое состояние; но для этого состояния мы знаем условно оптимальное управление и т. д. Таким образом, мы найдем оптимальное управление процессом, приводящее к максимально возможному выигрышу.

Таким образом, в процессе оптимизации управления методом динамического программирования многошаговый процесс "проходится" дважды:

— первый раз — от конца к началу, в результате чего находятся условно оптимальное управление на каждом шаге и оптимальный выигрыш (тоже условный) на всех шагах, начиная с данного и до конца процесса
— второй раз — от начала к концу, в результате чего находятся оптимальные управления на всех шагах процесса.

Можно сказать, что процедура построения оптимального управления методом динамического программирования распадается на две стадии: предварительную и окончательную. На предварительной стадии для каждого шага определяется условно оптимальное управление, зависящее от состояния системы (достигнутого в результате предыдущих шагов), и условно оптимальный выигрыш на всех оставшихся шагах, начиная с данного, также зависящий от состояния. На окончательной стадии определяется (безусловное) оптимальное управление для каждого шага. Предварительная (условная) оптимизация производится по шагам в обратном порядке: от последнего шага к первому; окончательная (безусловная) оптимизация — также по шагам, но в естественном порядке: от первого шага к последнему. Из двух стадий оптимизации несравненно более важной и трудоемкой является первая. После окончания первой стадии выполнение второй трудности не представляет: остается только "прочесть" рекомендации, уже заготовленные на первой стадии.

Основные необходимые свойства задач, к которым возможно применить эти идеи:

1. Задача должна допускать интерпретацию как $\boldsymbol{n}$-шаговый процесс принятия решений.
2. Задача должна быть определена для любого числа шагов и иметь структуру, не зависящую от их числа.
3. При рассмотрении $\boldsymbol{k}$-шаговой задачи должно быть задано некоторое множество параметров, описывающих состояние системы, от которых зависят оптимальные значения переменных. Причем это множество не должно изменяться при увеличении числа шагов.
4. Выбор решения (управления) на $\boldsymbol{k}$-м шаге не должен оказывать влияния на предыдущие решения, кроме необходимого пересчета переменных.

Задача о выборе траектории, задача последовательного принятия решения, задача об использовании рабочей силы, задача управления запасами — классические задачи динамического программирования.

Постановка задачи динамического программирования.

Постановку задачи динамического программирования рассмотрим на примере инвестирования, связанного с распределением средств между предприятиями. В результате управления инвестициями система последовательно переводится из начального состояния - S_0 в конечное - S_n. Предположим, что управление можно разбить на n шагов и решение принимается последовательно на каждом шаге, а управление представляет собой совокупность n пошаговых управлений. На каждом шаге необходимо определить два типа переменных - переменную состояния системы S_k и переменную управления x_k . Переменная S_k определяет, в каких состояниях может оказаться система на рассматриваемом k-м шаге. В зависимости от состояния S на этом шаге можно применить некоторые управления, характеризующиеся переменной x_k, которые удовлетворяют определенным ограничениям и называются допустимыми. Допустим $\vec{X} = (x_1, x_2, ..., x_n)$ - управление, переводящее систему на состояния S_0 в состояние S_n , а S_k - есть состояние системы на k-м шаге управления. Применение управляющего воздействия x_k на каждом шаге переводит систему в новое состояние $S_k (S, x_k)$ и приносит некоторый результат $W_k (S, x_k)$. Для каждого возможного состояния на каждом шаге среди всех возможных управлений выбирается оптимальное управление x_k^*, такое, чтобы результат, который достигается за шаги с k-го по последний n-ый, оказался бы оптимальным. Числовая характеристика этого результата называется **функцией Беллмана** $F_k (S)$ и зависит от номера шага k и состояния системы S.

Задача динамического программирования формулируется следующим образом: требуется определить такое управление $\vec{X}$, переводящее систему из начального состояния S_0 в конечное состояние S_n , при котором целевая функция принимает наибольшее (наименьшее) значение $F(S_0, \vec{X}) \Rightarrow ext$.

Рассмотрим более подробно особенности математической модели динамического программирования:

1) задача оптимизации формулируется как конечный многошаговый процесс управления;
2) целевая функция (выигрыш) является аддитивной и равна сумме целевых функций каждого шага: $F = \sum_{k=1}^{n} F_k\left(S_{k-1}, x_k\right) \Rightarrow ext$;
3) выбор управления $\boldsymbol{x_k}$ на каждом шаге зависит только от состояния системы $\boldsymbol{k}$ этому шагу $\boldsymbol{S_{k-1}}$, и не влияет на предшествующие шаги (нет обратной связи);
4) состояние системы $\boldsymbol{S_k}$ после каждого шага управления зависит только от предшествующего состояния системы $\boldsymbol{S_{k-1}}$ и этого управляющего воздействия $\boldsymbol{x_h}$ (отсутствие последействия) и может быть записано в виде уравнения состояния: $\boldsymbol{S_k = f_k(S_{k-1}, x_k), k = 1,\dots, n}$;
5) на каждом шаге управление $\boldsymbol{x_k}$ зависит от конечного числа управляющих переменных, а состояние системы зависит $\boldsymbol{S_k}$ – от конечного числа параметров;
6) оптимальное управление представляет собой вектор , определяемый последовательностью оптимальных пошаговых управлений: $\vec{X}^* = \left(x_1^*, x_2^*, \dots, x_n^*\right)$, число которых и определяет количество шагов задачи.

Достоинства динамического программирования.

1. Идея и метод динамического программирования наиболее приспособлены к дискретным задачам, каковыми являются задачи из экономики.
2. Метод динамического программирования применим при любом способе задания F_i и любом допустимом множестве состояний и управлений. Этого

преимущества лишены классические методы оптимизации и другие вычислительные методы математического программирования.

3. Вычислительные схемы метода динамического программирования в дискретном случае связаны с перебором оптимальных значений показателя эффективности и управления на i-м шаге для всех возможных значений переменной состояния, но объем расчетов по этому методу значительно меньше, чем при прямом переборе вариантов. Это связано с тем, что на этапе условной оптимизации неудачные варианты сразу отбрасываются, а сохраняются лишь условно оптимальные на данном шаге.

4. Метод динамического программирования дает возможность анализа чувствительности к изменению исходных данных S_i и n. Фактически здесь решается не одна задача, а множество однотипных задач для различных состояний S_i и различных i $(1 \leq i \leq n)$ на каждом шаге. Поэтому при изменении исходных данных можно не решать задачу заново, а сделать лишь несложные добавления к уже выполненным расчетам, т.е. продолжить уже решенную задачу за счет увеличения числа шагов n или числа значений S_i.

28. Функции Беллмана. Уравнения Беллмана. Условно-оптимальные управления

В основе метода динамического программирования лежит принцип оптимальности впервые сформулированный в 1953г. американским математиком Р.Э.Беллманом: каково бы ни было состояние системы в результате какого-либо числа шагов, на ближайшем шаге нужно выбирать управление так, чтобы оно в совокупности с оптимальным управлением на всех последующих шагах приводило к оптимальному выигрышу на всех оставшихся шагах, включая выигрыш на данном шаге.

При решении задачи на каждом шаге выбирается управление, которое должно привести к оптимальному выигрышу. Если считать все шаги независимыми, тогда оптимальным управлением будет то управление, которое обеспечит максимальный выигрыш именно на данном шаге. Однако, например, при покупке новой техники взамен устаревшей на ее приобретение затрачиваются определенные средства, поэтому доход от ее эксплуатации в начале может быть небольшой, а в следующие годы новая техника будет приносить больший доход. И наоборот, если принято решение оставить старую технику для получения дохода в текущем году, то в дальнейшем это приведет к значительным убыткам. Этот пример демонстрирует следующий факт: в многошаговых процессах управление на каждом конкретном шаге надо выбирать с учетом его будущих воздействий на весь процесс. Кроме того, при выборе управления на данном шаге следует учитывать возможные варианты состояния предыдущего шага. Например, при определении количества средств, вкладываемых в предприятие в i-м году, необходимо знать, сколько средств осталось в наличия к i-тому году и какой доход получен в предыдущем $(i - 1)$-м году.

Таким образом, при выборе шагового управления необходимо учитывать следующие требования:

1) возможные исходы предыдущего шага S_{k-1};

2) влияние управления x_k на все оставшиеся до конца процесса шаги ***(n-k)***.

В задачах динамического программирования первое требование учитывают, делая на каждом шаге условные предположения о возможных вариантах окончания предыдущего шага и проводя для каждого из вариантов условную оптимизацию.

Выполнение второго требования обеспечивается тем, что в этих задачах условная оптимизация проводится от конца процесса к началу.

Условная оптимизация.

На первом этапе решения задачи, называемом условной оптимизацией, определяются **функция Беллмана** и **оптимальные управления** для всех возможных состояний на каждом шаге, начиная с последнего, в соответствии с алгоритмом обратной прогонки.

На последнем, ***n***-м шаге оптимальное управление - x_n^* определяется функцией Беллмана:

$$F(S) = \max\left\{W_n\left(S, x_n\right)\right\}, \tag{28.1}$$

в соответствие с которой максимум выбирается из всех возможных значений x_n, причем $x_n \in X$.

Дальнейшие вычисления производятся согласно рекуррентному соотношению, связывающему функцию Беллмана на каждом шаге с этой же функцией, но вычисленной на предыдущем шаге.

В общем виде это уравнение имеет вид:

$$F_k(S) = \max_{x_k \in X}\left\{W_k\left(S, x_k\right) + F_{k+1}\left(S^n\left(S, x_k\right)\right)\right\}. \tag{28.2}$$

Этот максимум (или минимум) определяется по всем возможным для ***k*** и ***S*** значениям переменной управления X.

Безусловная оптимизация.

После того, как функция Беллмана и соответствующие оптимальные управления найдены для всех шагов с ***n***-го по первый, осуществляется второй этап решения задачи, называемый безусловной оптимизацией. Пользуясь тем, что на первом шаге (***k=1***) состояние системы известно - это ее начальное состояние S_0, можно найти оптимальный результат за все ***n*** шагов и оптимальное управление на первом шаге x_1, которое этот результат доставляет. После применения этого управления система перейдет в другое состояние $S^1\left(S, x_1^*\right)$, зная которое, можно, пользуясь результатами условной оптимизации, найти оптимальное управление на втором шаге x_2^*, и так далее до последнего ***n***-го шага. Вычислительную схему динамического программирования можно строить на сетевых моделях, а также по алгоритмам прямой прогонки (от начала) и обратной прогонки (от конца к началу). Рассмотрим примеры решения различных по своей природе задач, содержание которых требует выбора переменных состояния и управления.

29. Принцип Беллмана для оптимальных путей

Математический аппарат динамического программирования, основанный на методологии пошаговой оптимизации, может быть использован при нахождении кратчайших расстояний, например, на географической карте, представленной в виде сети. Решение задачи по определению кратчайших расстояний между пунктами отправления и пунктами получения продукции по существующей транспортной сети является исходным этапом при решении таких экономических задач, как оптимальное прикрепление потребителей за поставщиками, повышение производительности транспорта за счет сокращения непроизводительного пробега и др.

Пусть транспортная сеть состоит из 10 узлов, часть из которых соединены магистралями. На рисунке показана сеть дорог и стоимости перевозки единицы груза между отдельными пунктами сети, которые проставлены у соответствующих ребер. Необходимо определить маршрут доставки груза из пункта 1 в пункт 10, обеспечивающий наименьшие транспортные расходы.

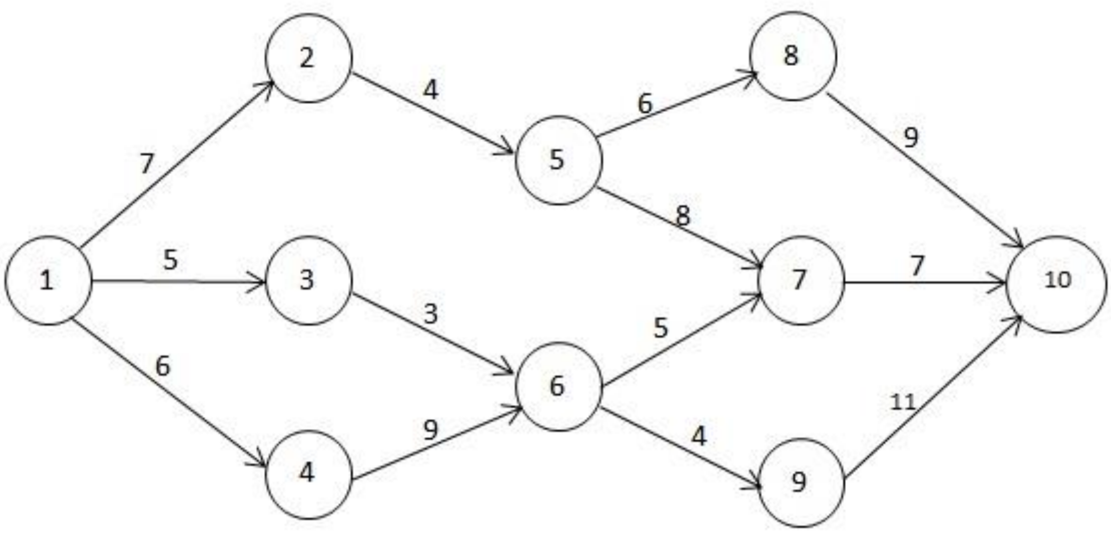

рис. 29.1

В задаче имеется ограничение - двигаться по изображенным на схеме маршрутам можно только слева на право, т.е. попав, например, в пункт **7**, мы имеем право переместиться только в пункт 10 и не можем возвратиться обратно в 5-й или 6-й. Эта особенность транспортной сети дает право отнести каждый из десяти пунктов к одному из поясов. Будем считать, что пункт принадлежит ***k***-му поясу, если из него попасть в конечный пункт ровно за ***k*** шагов, т.е. с заездом ровно в (***k***-**1**)-й промежуточный пункт. Таким образом, пункты 7, 8 и 9 принадлежат к первому поясу**,** 5 и 6 - ко второму, 2, 3 и 4 - к третьему и 1 - к четвертому. Тогда на ***k***-м шаге будем находить оптимальные маршруты перевозки груза из пунктов ***k***-го пояса до конечного пункта. Оптимизацию будем производить с конца процесса, и потому, дойдя до ***k***-го шага, неизвестно, в каком из пунктов ***k***-го пояса окажется груз, перевозимый из первого пункта.

Введем обозначения:

k - номер шага ($k=1,2,3,4$);

i - пункт, из которого осуществляются перевозки ($i =1,2,..., 9$);

j - пункт, в который доставляется груз ($j=2,3,.., 10$);

$C_{i,j}$ - стоимость перевозки груза из пункта ***i*** в пункт ***j***;

$F_k(i)$ - минимальные затраты на перевозку груза на ***k***-м шаге решения задачи из пункта ***i*** до конечного пункта.

Очевидно, что минимум затрат на перевозку груза из пунктов ***k***-го пояса до пункта 10 будет зависеть от того, в каком пункте этого пояса мы оказались. Номер ***i*** пункта, принадлежащего ***k***-му поясу, будет являться переменной состояния системы на ***k***-м шаге. Поскольку оптимизация осуществляется с конца процесса, то, находясь в некотором пункте ***i*** ***k***-го пояса, принимается решение о перемещении груза в один из пунктов (***k***–**1**)-го пояса, а направление дальнейшего движения известно из предыдущих шагов. Номер ***j*** пункта (***k*-1**)-го пояса будет переменной управления на ***k***-м шаге.

Для первого шага управления (***k*-1**) функция Беллмана представляет собой минимальные затраты на перевозку груза из пунктов 1-го пояса в конечный пункт, т.е. $F_1(i) = C_{i,10}$. Для последующих шагов затраты складываются из двух слагаемых - стоимости перевозки груза $C_{i,j}$ из пункта ***i*** ***k***-го пояса в пункт ***j*** (***k*-1**)-го пояса и минимально возможных затрат на перевозку из пункта ***j*** до конечного пункта, т.е. - $F_{k-1}(i)$. Таким образом, функциональное уравнение Беллмана будет иметь вид

$$F_k(i) = \min_j \left\{ C_{ij} + F_{k-1}(j) \right\} \tag{29.1}$$

Минимум затрат достигается на некотором значении ***j****, которое является оптимальным направлением движения из пункта ***i*** в конечный пункт. На четвертом шаге попадаем на 4-й пояс и состояние системы становится определенным ***i=1***. Функция $F_4(1)$ представляет собой минимально возможные затраты по перемещению груза из 1-го пункта в 10-й. Оптимальный маршрут определяется в результате анализа всех шагов в обратном порядке, а выбор некоторого управления ***j*** на ***k***-м шаге приводит к тому, что состояние системы на (***k*** **1**)-м шаге становится определенным.

Задача 29.1. Решим сформулированную выше задачу, исходные данные которой приведены на рисунке 29.1.

I этап. ***Условная оптимизация.***

1-й шаг. $k = 1$.

$$F_1(i) = C_{i,10}$$

На первом шаге в пункт 10 груз может быть доставлен из пунктов 7,8 или 9.

Таблица 29.1

j / i	10	$F_1(i)$	j^*
7	7	7	10
8	9	9	10
9	11	11	10

2-й шаг. ***k = 2*** .

Функциональное уравнение на втором шаге принимает вид:

$$F_2(i) = \min_j \{C_{ij} + F_1(j)\}.$$

Все возможные перемещения груза на втором шаге и результаты расчета приведены в следующей таблице 29.2.

Таблица 29.2

j / i	7	8	9	$F_2(i)$	j^*
5	8 + 7	6 +9	-	15	7;8
6	5 +7	-	4 + 11	12	7

3-й шаг. ***k = 3***.

$$F_3(i) = \min_j \{C_{ij} + F_2(j)\}$$

Таблица 29.3

j / i	5	6	$F_3(i)$	j^*
2	4 + 15	-	19	5
3	-	3 + 12	15	6
4	-	9 + 12	21	6

4-й шаг. ***k = 4***.

$$F_{4-}(i) = \min_j \{C_{ij} + F_3(j)\}$$

Таблица 29.4

j / i	2	3	4	$F_4(i)$	j^*
1	7 + 19	5 + 15	6 + 21	20	3

II этап. *Безусловная оптимизация.*

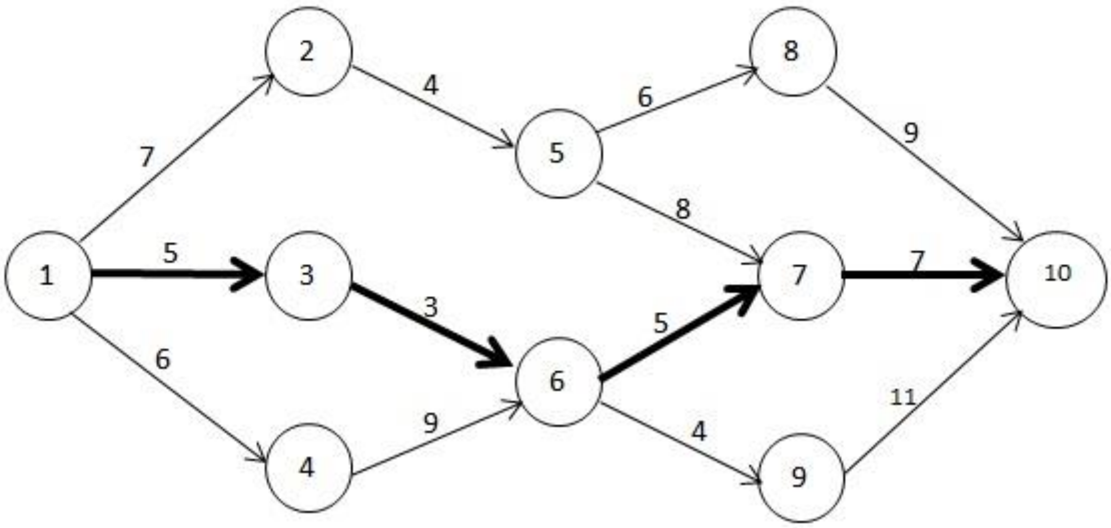

рис. 29.2

На этапе условной оптимизации получено, что минимальные затраты на перевозку груза из пункта 1 в пункт 10 составляют $F_4(1)=20$. Данный результат достигается при движении груза из 1-го пункта в 3-й. По данным табл.29.3, из пункта 3 необходимо двигаться в пункт 6, затем - в пункт 7 (см. табл.29.2) и из него - в конечный пункт (см. табл.29.1). Таким образом, оптимальный маршрут доставки груза: 1 => 3 => 6 => 7 => 10. На рисунке 29.2 он показан жирными стрелками.

30. Оптимальное распределение инвестиций как задача динамического программирования

Инвестор выделяет средства в размере ***D*** условных единиц, которые должны быть распределены между ***n***-предприятиями. Каждое ***k***-тое предприятие при инвестировании в него средств ***x*** приносит прибыль W_k ***(S,*** x_k***)*** условных единиц, ***k=1...n***. Нужно выбрать оптимальное распределение инвестиций между предприятиями, обеспечивающее максимальную прибыль.

Выигрышем ***F*** в данной задаче является прибыль, приносимая ***n***-предприятиями.

Построение математической модели.

1. Определение числа шагов. Число шагов ***n*** равно числу предприятий, в которые осуществляется инвестирование.

2. Определение состояний системы. Состояние системы на каждом шаге характеризуется количеством средств S_k, имеющихся в наличии перед данным шагом, $S \le D$.

3. Выбор шаговых управлений. Управление на ***k***-м шаге $x_i, i = 1,...,n$ является количество средств, инвестируемых в ***k***-тое предприятие.

4. Функция выигрыша на ***k***-м шаге:

$$W_k\left(S, x_k\right). \tag{30.1}$$

Это прибыль, которую приносит ***k***-тое предприятие при инвестировании в него средств x_k.

$$F = \sum_{k=1}^{n} W_k\left(S, x_k\right), \tag{30.2}$$

следовательно, данная задача может быть решена методом динамического программирования.

5. Определение функции перехода в новое состояние.

$$S^n\left(S, x_k\right) = S - x. \qquad (30.3)$$

Таким образом, если на ***k***-м шаге система находилась в состоянии S, а выбрано управление x, то на ***(k+1)***-м шаге система будет находиться в состоянии $S - x$. Другими словами, если в наличии имеются средства в размере S у.е., и в ***k***-тое предприятие инвестируется x у.е., то для дальнейшего инвестирования остается ($S - x$) у.е.

6. Составление функционального уравнения для ***k=n***:

$$F_n(S) = W_n\left(S, x_n\right), \qquad (30.4)$$

$$x_n(S) = S. \qquad (30.5)$$

На последнем шаге, т.е. перед инвестированием средств в последнее предприятие, условное оптимальное управление соответствует количеству средств, имеющихся в наличии; т.е. сколько средств осталось, столько и надо вложить в последнее предприятие. Условный оптимальный выигрыш равен доходу, приносимому последним предприятием.

7. Составление основного функционального уравнения.

Подставив в формулу (28.2) выражение (30.3), получим следующее функциональное уравнение:

$$F_k(S) = \max_{x_k \in x}\left\{W_k\left(S, x_k\right) + F_{k+1}(S - x)\right\} \qquad (30.6)$$

Поясним данное уравнение. Пусть перед ***k***-м шагом у инвестора остались средства в размере S у.е. Тогда x у.е. он может вложить в ***k***-тое предприятие, при этом оно принесет доход $W_k\left(S, x_k\right)$, а оставшиеся $(S - x)$ у.е. - в остальные предприятия с ***k+1***-го до ***n***-го. Условный оптимальный выигрыш от такого вложения $F_{k+1}(S - x)$. Оптимальным будет то условное управление x, при котором сумма $W_k\left(S, x_k\right)$ и $F_{k+1}(S - x)$ максимальна.

Задача 30.1. Какова оптимальная прибыль, приносимая тремя предприятиями при инвестировании в них 5000 у.е. (***D***=5000, ***n***=3). Значения $W_k\left(S, x_k\right)$, $k=1,2,3$ заданы в таблице 30.1.

Таблица 30.1

x_k тыс. усл. ед.	$W_1\left(S, x_k\right)$ тыс. усл. ед.	$W_2\left(S, x_k\right)$ тыс. усл.ед.	$W_3\left(S, x_k\right)$ тыс. усл.ед.
1	1,5	2	1,7
2	2	2,1	2,4
3	2,5	2,3	2,7
4	3	3,5	3,2
5	3,6	4	3,5

Для $x_1 > x_2 \quad W_k\left(S, x_1\right) \geq W_k\left(S, x_2\right), \quad i=1,2,3,4,5.$ (30.7)

Решение. Для простоты в задаче сделано предположение, что вкладываются только тысячи условных единиц.

Проведем **условную оптимизацию**.

По ее результатам заполняется таблица 30.2.

Таблица 30.2

S	***i=3***		***i=2***		***i=1***	
	$x_k(S)$	$F_k(S)$	$x_k(S)$	$F_k(S)$	$x_k(S)$	$F_k(S)$
1	1	1,7	0	2		
2	2	2,4	1	3,7		
3	3	2,7	1	4,4		
4	4	3,2	1	4,7		
5	5	3,5	1/4	5,2	2	6,4

В первой колонке таблицы записываются возможные состояния системы $S=1,\ldots,5$, в верхней строке - номера шагов $i=1,2,3$. На каждом шаге определяются условные оптимальные управления $x_k(S)$ и условные оптимальные выигрыши $F_k(S)$.

- Проведение условной оптимизации для последнего шага ***i***=3. Функциональное уравнение на последнем шаге имеет вид:

$$F_3(S)=W_3(S,x_3), \tag{30.8}$$

$$x_3(S)=S$$

поэтому два столбца таблицы 30.2, соответствующие ***i***=3, заполняются автоматически по таблице 30.1 исходных данных.

- Условная оптимизация для ***i***=2. Функциональное уравнение

$$F_2(S)=\max_{x\le S}\left\{W_2(S,x_2)+F_3(S-x)\right\}. \tag{30.9}$$

Для проведения условной оптимизации заполним ряд вспомогательных таблиц (таблицы 30.3—30.8), соответствующих различным значениям ***S***, т.е. различным исходам окончания предыдущего шага.

1) S=1

Таблица 30.3

x	$1-x$	$W_2(S,x_k)$	$F_3(1-x)$	$W_2(S,x_k)+F_3(1-x)$
0	1	0	1,7	1,7
1	0	2	0	2

$\max\limits_{x\le 1}\{1,7;\,2\}=2$, следовательно, $F_2(1)=2$, $x_2(1)=1$.

2) S=2

Таблица 30.4

x	$2-x$	$W_2(S,x_k)$	$F_3(2-x)$	$W_2(S,x_k)+F_3(2-x)$
0	2	0	2,4	2,4
1	1	2	1,7	3,7
2	0	2,1	0	2,1

$\max\limits_{x\le 2}\{2,4;\,3,7;\,2,1\}=3,7$, следовательно, $F_2(2)=3,7$; $x_2(2)=1$

3) S=3

Таблица 30.5

x	$3-x$	$W_2(S,x_k)$	$F_3(3-x)$	$W_2(S,x_k)+F_3(3-x)$
0	3	0	2,7	2,7
1	2	2	2,4	4,4
2	1	2,1	1,7	3,8
3	0	2,3	0	2,3

$\max\limits_{x\le 3}\{2,7;\ 4,4;\ 3,8;\ 2,3\}=4,4$, следовательно, $F_2(3)=4,4$; $x_2(3)=1$

4) S=4

Таблица 30.6

x	$4-x$	$W_2(S,x_k)$	$F_3(4-x)$	$W_2(S,x_k)+F_3(4-x)$
0	4	0	3,2	3,2
1	3	2	2,7	4,7
2	2	2,1	2,4	4,5
3	1	2,3	1,7	4
4	0	3,5	0	3,5

$\max\limits_{x\le 4}\{3,2;\ 4,7;\ 4,5;\ 3,5\}=4,7$, следовательно, $F_2(4)=4,7$; $x_2(4)=1$.

5) S=5

Таблица 30.7

x	$5-x$	$W_2(S,x_k)$	$F_3(5-x)$	$W_2(S,x_k)+F_3(5-x)$
0	5	0	3,5	3,5
1	4	2	3,2	5,2
2	3	2,1	2,7	4,8
3	2	2,3	2,4	4,7
4	1	3,5	1,7	5,2
5	0	4	0	4

$$\max_{x \le 5} \{3,5;\ 5,2;\ 4,8;\ 5,2;\ 4\} = 5,2$$

Для ***S***=5 $F_2(5) = 5,2$ возможны два условных варианта управления: $x_2(5) = 1$ и $x_2(5) = 4$.

- Условная оптимизация для ***i***=1.

Перед первым шагом состояние системы известно.

S=***D***=5 тыс. у.е., и условную оптимизацию следует проводить только для этого значения ***S***=5.

Таблица 30.8

x	$5-x$	$W_1(S, x_k)$	$F_2(5-x)$	$W_1(S, x_k) + F_2(5-x)$
0	5	0	5,2	5,2
1	4	1,5	4,7	6,2
2	3	2	4,4	6,4
3	2	2,5	3,7	6,2
4	1	3	2	5
5	0	3,6	0	3,6

$\max_{x \le 5} \{5,1;\ 6,2;\ 6,4;\ 6,2; 5; 3,6\} = 6,4$, следовательно, $F_1(5) = 6,4$, $x_1(5) = 2$.

Оптимальная прибыль, приносимая тремя предприятиями при инвестировании в них 5000 у.е., равна 6,4 тыс. у.е.

$F = F_1(5) = 6,4$.

Проведем **безусловную оптимизацию**.

Ее результаты отмечены в таблице 39.2.

Для ***i***=1 $S_1 = 5$ $F_1(5) = 6,4$; $x_1^* = x_1(5) = 2$.

Для ***i***=2 по формуле (39.3) $S_2 = S_1 - x_1 = 5 - 2 = 3$.

$F_2(3) = 4,4$; $x_2^* = x_2(3) = 1$.

Для ***i***=3 $S_3 = S_2 - x_2 = 3 - 1 = 2$.

$F_3(2) = 2,4$; $x_3^* = x_3(2) = 2$.

$x^* = (2;1;2)$.

Следует понимать, что полученное решение есть лишь некоторое приближение к оптимальному решению. Его можно улучшить, т.е. приблизить к оптимальному, взяв более мелкий шаг оптимизации, например, вкладывать в предприятия средства, кратные 500 у.е.

31.Задача о замене оборудования

Чем дольше механизм эксплуатируется, тем выше затраты на его обслуживание и ниже его производительность. Когда срок эксплуатации механизма достигает определенного уровня, может оказаться более выгодной его замена. Задача о замене оборудования, таким образом, сводится к определению оптимального срока эксплуатации механизма.

Предположим, что мы занимаемся заменой механизмов на протяжении n лет. В начале каждого года принимается решение либо об эксплуатации механизма еще год, либо о его замене новым.

Обозначим через $r(t)$ и $c(t)$ прибыль от эксплуатации t–летнего механизма на протяжении года и затраты на его обслуживание за тот же период. Далее пусть $s(t)$ - стоимость продажи механизма, который эксплуатировался t лет. Стоимость приобретения нового механизма остается неизменной на протяжении всех лет и равна I.

Элементы модели динамического программирования таковы:

1. Этап i представляется порядковым номером года i, $i=1,2,\dots n$.
2. Вариантами решения на i–м этапе (т. е. для i–го года) являются альтернативы: продолжить эксплуатацию или заменить механизм в начале i–го года.
3. Состоянием на i–ом этапе является срок эксплуатации (возраст) механизма к началу i–го года.

Пусть $f_i(t)$– максимальная прибыль, получаемая за годы от i до n при условии, что в начале i–го года имеется механизм t– летнего возраста.

Рекуррентное уравнение имеет следующий вид:

$$f_i(t) = \max\begin{cases} \mathrm{r}(t) - c(\mathrm{t}) + \mathrm{f}_{\mathrm{i+1}}(\mathrm{t}+1) & (1) \\ r(0) + s(t) - I - c(0) + f_{i+1}(1) & (2) \end{cases}$$

(1)– если эксплуатировать механизм

(2)– если заменить механизм.

Постановка задачи в общем виде.

Рассмотрим задачу в общей постановке.

Пусть в течение периода, состоящего из *n* этапов, предприятие использует оборудование, которое вместе с установкой имеет стоимость I. Со временем оборудование изнашивается.

При этом:

1) суммарная производительность оборудования r(t) снижается, так как увеличивается время, необходимое для его профилактики и ремонта;
2) возрастают затраты на обслуживание и ремонт c(t) данного оборудования.

Требуется определить моменты (их для длительного периода эксплуатации оборудования может быть несколько), когда следует заменять старое оборудование на новое, чтобы суммарная прибыль за весь период эксплуатации оборудования была максимальной. Под прибылью имеется в виду чистый доход от продажи выпущенной продукции за вычетом стоимости обслуживания и ремонта или замены оборудования. Предполагается, что заменяемое оборудование имеет нулевую стоимость, а доход от его реализации по остаточной стоимости не учитывается. Это допущение не является принципиальным и введено для упрощения анализа.

Обозначим моменты начала каждого этапа через $T_0, T_1, \dots, T_{n-1}$; момент завершения всего периода — через T_n. Тогда на каждом этапе $T_0, T_1, \dots, T_{n-1}$ мы можем выбрать одно из двух возможных управлений:

U_1 — сохранить установленное оборудование и продолжать его эксплуатацию;

U_2 — приобрести новое оборудование и заменить старое.

Здесь следует подчеркнуть, что управленческие решения могут применяться только в моменты $T_1, T_2, \dots, T_{n-1}$.

Оптимальной инвестиционной политикой (стратегией управления) *U** называется совокупность таких решений $U_1^*, U_2^*, \dots, U_{n-1}^*$, в результате реализации которых рассматриваемое производство за n шагов переходит из начального S_0 состояния в конечное S_n и при этом суммарная прибыль от эксплуатации оборудования достигает максимального значения. Состояние процесса на S_i этапе характеризуется возрастом оборудования. Максимально возможный возраст оборудования в момент t_i равен i, а минимальный равен 1, если на предыдущем этапе произошла замена оборудования.

Прибыль f_i, получаемая на i-м этапе, зависит от состояния оборудования и сделанного нами выбора (управления) U_i:

$$f_i = f_i(t) = \begin{cases} r(t) - c(t) + f_{i+1}(t+1) & U_i = U_1, \text{если продорлжать экспуатаию оборудования} \\ r(0) - c(0) + s(t) - I + f_{i+1}(1), & U_i = U_2, \text{если приобрести новое оборудование} \end{cases}$$

(31.1)

где $r(0)$ — стоимость продукции, выпускаемой на новом оборудовании; $s(t)$ – стоимость продажи оборудования, который эксплуатировался t лет; I — полная стоимость нового оборудования.

Очевидно, что суммарная прибыль F, которую требуется максимизировать для разработки оптимальной инвестиционной политики, равна сумме прибылей на каждом из этапов производства. Таким образом, суммарная прибыль будет определяться по формуле:

$$F(t) = \sum_{k=0}^{n-1} f_i \qquad (31.2)$$

В выражении (31.2) критерий является аддитивным. Для оптимизации подобных критериев возможно использование методов динамического программирования.

Таким образом, суть разработки оптимальной инвестиционной политики — это получение максимального значения суммарной прибыли F. Она зависит от результатов управления процессом эксплуатации оборудования на каждом шаге.

Особенностью динамического программирования является рассмотрение процесса от конца к началу, то есть сначала анализируется n-1

этап, ищется здесь оптимальное решение, затем то же делается на *n-2* этапе и т.д.

Таким образом, основой математического аппарата для нахождения оптимального решения является принцип оптимальности Беллмана:

$$F_i^* = f_i(t) + f_{i+1}^* = \max_{U_i}\{[f(t) + F_{i+1}(S_{i+1})]\}. \qquad (31.3)$$

При вычислениях с каждым из этапов ассоциирована одна управляемая переменная. Набор рекуррентных вычислительных процедур, связывающих различные этапы, обеспечивает получение оптимального решения задачи в целом при достижении последнего этапа.

Управление U_i^* выбирают следующим образом:

1) для всех возможных управлений U_i определяются значения сумм

$$F_i(S_n) = f_i(S_i, U_i) + F_{n+1}(S_{n+1}), \qquad (31.4)$$

которые сравниваются между собой;

2) в качестве условно оптимального управления выбирается то управление, которое соответствует наибольшему значению $F_i(S_i)$ (при решении задачи максимизации).

Если наибольшее значение $F_i(S_i)$ получается при применении нескольких различных управлений, то все они являются условно оптимальными. В этом случае U_i^* не единственный.

При выборе управления на предпоследнем этапе (при *i*=*n*-1) никаких дальнейших шагов по эксплуатации оборудования не планируется, так что естественно считать $F_{n+1}(S_{i+1}) = F_n(S_n) = 0$. Тогда в сумме (31.4) остается лишь первое слагаемое.

Таким образом, процесс продвигается от конечного этапа к начальному и на всех промежуточных шагах находятся значения:

– условно оптимальных управлений U_i^* для каждого из возможных состояний S_i;

– оптимальное значение $F_i^*(S_i)$.

Вектор оптимального управления *U** всего процесса будет получен при объединении найденных в ходе вычислений $F_i^*(S_i)$ условно оптимальных управлений *U**: $U^* = (U_0^*, U_2^*, \ldots, U_{n-1}^*)$.

Задача 31.1. Пусть требуется выработать оптимальную стратегию для эксплуатации оборудования на 6-ти летнем периоде (таблица 31.1).

Таблица 31.1

Возраст оборудования – *t*(лет)	0	1	2	3	4	5
Производительность оборудования – *r(t)* (млн. руб.)	9	8	7	6	5	5
Затраты на ремонт – *c(t)* (млн. руб.)	1	1	2	2	3	4

Стоимость нового комплекта оборудования *I*=6 млн. руб.

Решение. Будем предполагать, что за пределами срока, представленного в таблице, эксплуатация оборудования нас не интересует.

Обозначения:

f_i (t) – максимальный доход на временном отрезке *i*,…,*n*, при возрасте оборудования t лет в начале *i*–го года.

I – начальные затраты на приобретение и установку нового оборудования (цена нового оборудования).

Составляем уравнение Беллмана.

$$f_i(t) = \max\begin{cases} \mathrm{r}(t) - c(\mathrm{t}) + \mathrm{f}_{\mathrm{i+1}}(\mathrm{t}+1), & \text{если эксплуатировать оборудование} \\ r(0) - c(0) - I + f_{i+1}(1), & \text{если заменить оборудование} \end{cases}$$

Приведем расчеты.

После пятого этапа эксплуатация оборудования не предусматривается, поэтому будем иметь:

Этап 5.

$$f_5(t) = \max\begin{cases} \mathrm{r}(t) - c(\mathrm{t}), & \text{если эксплуатировать оборудование} \\ r(0) - c(0) - I, & \text{если заменить оборудование} \end{cases}$$

Таблица 31.2

	$r(t) - c(t)$	$r(0) - c(0) - I$	Оптимальное решение	
t	старое	новое	$f_5\ (t)$	решение
1	8-1=7	9-6-1=2	7	с
2	7-2=5	9-6-1=2	5	с
3	6-2=4	9-6-1=2	4	с
4	5-3=2	9-6-1=2	2	с,н
5	5-4=1	9-1-6=2	2	н

Переходим к анализу процесса эксплуатации оборудования на четвертом этапе.

Этап 4.

$$f_4\ (t)\ =\ \ max\begin{cases} r(t) - c(t) + f_5(t+1)\ ,\text{если эксплуатировать оборудование} \\ r(0) - c(0) - I + f_5(1)\ ,\ \text{если заменить оборудование} \end{cases}$$

Таблица 31.3

	$r(t) - c(t) + f_5(t+1)$	$r(0) - c(0) - I + f_5(1)$	Оптимальное решение	
t	старое	новое	$f_4\ (t\ )$	решение
1	8-1+5=12	9-1-6+7=9	12	с
2	7-2+4=9	9-1-6+7=9	9	с,н
3	6-2+2=6	9-1-6+7=9	9	н
4	5-3+2=4	9-1-6+7=9	9	н

Теперь приступим к этапу 3. На этом этапе возраст оборудования может принимать значения 1,2 и 3.

Этап 3.

$$f_3\ (t\ )\ =\ \ max\begin{cases} r(t) - c(t) + f_4(t+1)\ ,\ \text{если эксплуатировать оборудование} \\ r(0) - c(0) - I + f_4(1)\ \ ,\ \text{если заменить оборудование} \end{cases}$$

Таблица 31.4

	$r(t) - c(t) + f_4(t+1)$	$r(0) - c(0) - I + f_4(1)$	Оптимальное решение	
t	старое	новое	$f_3\ (t\ )$	решение
1	8-1+6=16	9-1-6+12=14	16	с
2	7-2+9=14	9-1-6+12=14	14	с, н
3	6-2+9=13	9-1-6+12=14	14	н

Этап 2.

$$f_2\ (t\ )\ =\ \ max\begin{cases} r(t) - c(t) + f_3(t+1)\ ,\ \text{еслиэ ксплуатировать оборудование} \\ r(0) - c(0) - I + f_3(1)\ \ ,\ \text{если заменить оборудование} \end{cases}$$

Таблица 31.5

	$r(t) - c(t) + f_3(t+1)$	$r(0) - c(0) - I + f_3(1)$	Оптимальное решение	
t	старое	новое	$f_2(t)$	решение
1	8-1+14=21	9-1-6+16=18	21	с
2	7-2+14=19	9-1-6+16=18	19	с

Этап 1.

$$f_1(t) = \max\begin{cases} r(t) - c(t) + f_2(t+1), & \text{если эксплуатировать оборудование} \\ r(0) - c(0) - I + f_2(1), & \text{если заменить оборудование} \end{cases}$$

Таблица 31.6

	$r(t) - c(t) + f_2(t+1)$	$r(0) - c(0) - I + f_2(1)$	Оптимальное решение	
t	старое	новое	$f_1(t)$	решение
1	8-1+19=26	9-1-6+21=23	26	с

Оптимальное решение: $f_1(t) = 26, (\text{с, с, н, с, с.})$

Лучше учесть и доходы от нового оборудования. Они равны $r(0) - c(0) = 8$.

Окончательно получаем оптимальное решение в более привычном виде: $f_0(t) = 34, (\text{н, с, с, н, с, с.})$.

Значение $f_0(t) = 34$ — наибольшая прибыль, которую можно получить при эксплуатации оборудования в данном примере.

По результатам рассмотренного примера можно сделать следующий вывод. Оборудование нужно заменить один раз по истечении трех лет эксплуатации. Максимальная прибыль при этом составит 34 млн. руб.

Задача 31.2. Компания планирует определить оптимальную политику замены используемого в настоящее время трехлетнего механизма на протяжении следующих 4 лет (n = 4), т.е. вплоть до начала пятого года. Таблица 31.7 содержит относящиеся к задаче данные. Компания требует обязательной замены механизма, который находится в эксплуатации 6 лет. Стоимость нового механизма равна 100 000 долл.

Таблица 31.7

Возраст t без учета первых 3-х лет (года)	Возраст t^* с учетом 3-х лет (года)	Прибыль $r(t)$ (долл.)	Стоимость обслуживания $c(t)$ (долл.)	Остаточная стоимость $s(t)$ (долл.)
0		20000	200	-
1	3	19000	600	80000
2	4	18500	1200	60000
3	5	17200	1500	50000
4	6	15500	1700	30000
5	7	14000	1800	10000
6	8	12200	2200	5000

Решение. Определение допустимых значений возраста механизма на каждом этапе является нетривиальной задачей. На рис. 31.1 представлена рассматриваемая задача замены оборудования в виде сети. В начале первого года имеется механизм, эксплуатирующийся 3 года (на графике рис. 31.1 по оси **Y** откладывается возраст механизма). Мы можем либо заменить его, купить новое ***(З)*** или ***(Н)***, либо эксплуатировать старое ***(С)*** на протяжении следующего года. Если механизм заменили, то в начале второго года его возраст будет равен одному году, в противном случае его возраст будет 4 года. Такой же подход используется в начале каждого года, начиная со второго по четвертый.

Если однолетний механизм заменяется в начале второго или третьего года, то заменивший его механизм к началу следующего года также будет однолетним. К тому же, в начале 4-го года 6-летний механизм обязательно должен быть заменен, если он еще эксплуатируется; в конце 4-го года все механизмы продаются ***(П)*** в обязательном порядке. На схеме сети также видно, что в начале второго года возможны только механизмы со сроком эксплуатации 1 или 4 года. В начале третьего года механизм может иметь возраст 1, 2 или 5 лет, а в начале четвертого — 1, 2, 3 или 6 лет.

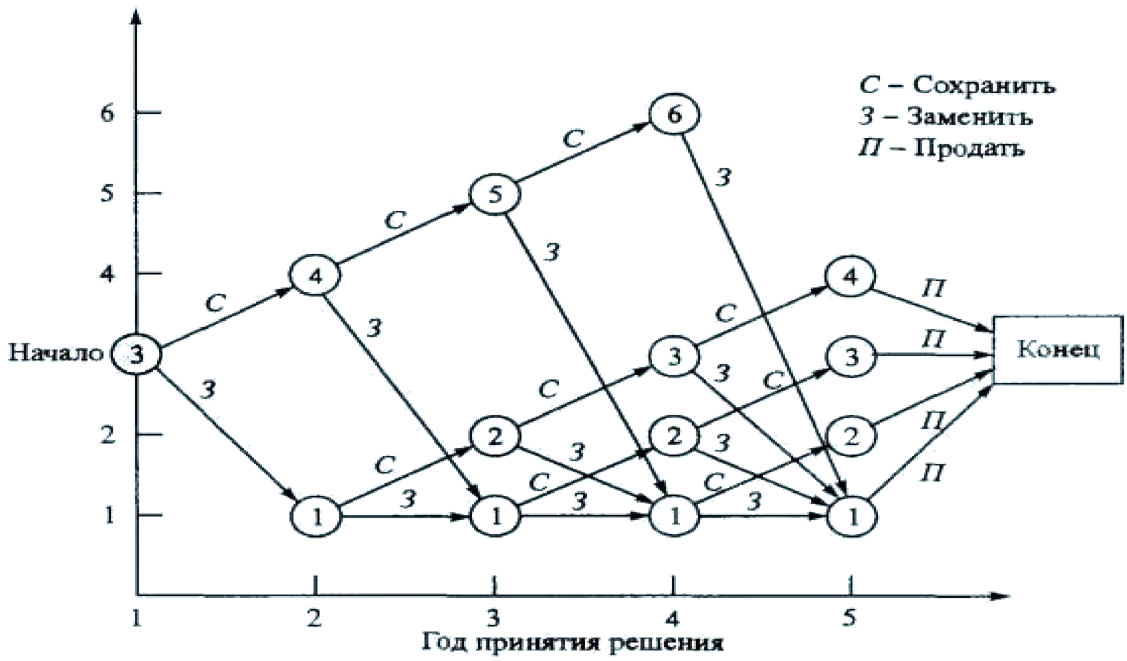

рис. 31.1

Решение данной задачи эквивалентно поиску маршрута максимальной длины (т.е. приносящего максимальную прибыль) от начала первого года к концу четвертого в сети, показанной на рис. 31.1. При решении таких задач часто используют табличную форму записи. (Числовые данные в таблице кратны тысячам долларов.)

Этап 4.

Таблица 31.8

	С	Н	Оптимальное решение	
t	*r(t) + s(t + 1)-c(t)*	r*(0)+s(t)+s(1) - c(0) - I*	$f_4(t)$	Решение
1	19,0 + 60 - 0,6 = 78,4	20 + 80 + 80 - 0,2 - 100 = 79,8	79,8	Н
2	18,5 + 50 - 1,2 = 67,3	20 + 60 + 80 - 0,2 - 100 = 59,8	67,3	С
3	17,2 + 30 - 1,5 = 45,7	20 + 50 + 80 - 0,2 - 100 = 49,8	49,8	Н
4	Необходима замена	20 + 5 + 80 - 0,2 - 100 = 4,8	4,8	Н

Этап 3.

Таблица 31.9

	С	Н	Оптимальное решение	
t	*r(t) - c(t) +* f_4*(t + 1)*	*r(0) + s(t) - c(0) - I +* f_4*(1)*	$f_3(t)$	Решение
1	19,0 - 0,6 + 67,3 = 85,7	20+80- 0,2 - 100 + 79,8 = 79,6	85,7	С
2	18,5 - 1,2 + 49,8 = 67,1	20+60- 0,2 - 100 + 79,8 = 59,6	67,1	С
5	14,0 - 1,8 + 4,8 = 17,0	20+ 10 - 0,2 - 100 + 79,8 = 9,6	17,0	Н

Этап 2.

Таблица 31.10

	С	Н	Оптимальное решение	
t	*r(t) - c(t) +* f_3*(t + 1)*	*r(0) + s(t) - c(0) - I +* f_3*(1)*	$f_2(t)$	Решение
1	19,0 - 0,6 + 67,1 = 85,5	20+80- 0,2 - 100 + 85,7 = 85,5	85,5	С или Н
4	15,5 - 1,7 + 19,6 = 33,4	20+30- 0,2 - 100 + 85,7 = 35,5	35,5	Н

Этап 1.

Таблица 31.11

	С	Н	Оптимум	
t	*r(t) - c(t) +* f_2*(t + 1)*	*r(0) + s(t) - c(0) - I +* f_2*(1)*	$f_1(t)$	Решение
3	17,2 - 1,5 + 35,5 = 51,2	20+50-0,2 - 100 + 85,5 = 55,3	55,3	Н

На рис. 31.2 показана последовательность получения оптимального решения. В начале первого года оптимальным решением при ***t*=3** является замена механизма. Следовательно, новый механизм к началу второго года будет находиться в эксплуатации 1 год. При ***t*=1** в начале второго года оптимальным решением будет либо использование, либо замена механизма. Если он заменяется, то новый к началу третьего года будет находиться в эксплуатации 1 год, иначе механизм будет иметь возраст 2 года. Описанный процесс продолжается до тех пор, пока не будет определено оптимальное решение для четвертого года.

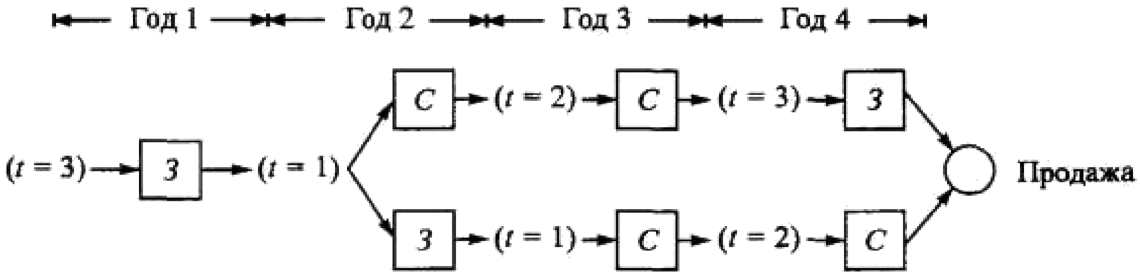

рис. 31.2

Следовательно, начиная с первого года эксплуатации механизма, альтернативными оптимальными стратегиями относительно замены механизма будут ***(Н, С, С, Н)*** и ***(Н, Н, С, С)***. Общая прибыль составит 55 300 долл.

32.Марковские модели принятия решений

Определение 32.1. **Марковский процесс** – это процесс, протекающий в некоторой системе, при котором для каждого момента времени поведение системы в будущем зависит только от состояния системы в данный момент и не зависит от того, каким образом система пришла в это состояние.

Определение 32.2. **Случайная цепь** – это случайный процесс $X(t)$ с дискретным временем и дискретным множеством значений.

Определение 32.3. **Простая марковская цепь** – это случайная цепь, для которой в каждый момент времени закон распределения $X(t_k)$ вполне определяется значением $X(t_{k-1})$ и не зависит от предыдущих значений.

Марковская задача принятия решений – это задача математического программирования к многошаговым задачам принятия решений в условиях риска, в которой процесс изменения состояния любой изучаемой системы является марковским процессом с конечным множеством возможных состояний и дискретным временем.

Математические модели, приводящие к таким задачам, называют марковскими моделями принятия решений.

В марковских моделях принятия решений поощрения (доход, потери) задают матрицей доходов.

Определение 32.4. **Матрица доходов** – это матрица, элементами которой являются доходы (положительные значения) или затраты (отрицательные значения), возникающие вследствие перехода системы из одного возможного состояния в другое.

Определение 32.5. **Матрица переходных вероятностей** – это матрица, каждый элемент p_{ij} которой является вероятностью перехода из состояния S_i в состояние S_j.

Матрицы переходных вероятностей и матрицы доходов зависят от стратегий, т.е. допустимых решений, которыми располагает лицо, принимающее решения.

Основная цель — определение оптимальной стратегии (оптимального решения), максимизирующей ожидаемый доход за конечное или бесконечное число этапов марковского процесса изменения состояния изучаемой системы.

Рассмотрим некоторую динамическую систему S с возможными дискретными состояниями Sj, $j = 1, 2, \ldots m$, которая в фиксированные последовательные моменты времени $t_1 < t_2 < \ldots$ случайным образом переходит скачком (мгновенно) из одного возможного состояния в другое или остается в прежнем. При этом будем предполагать, что сам процесс изменения состояния изучаемой системы S является марковским процессом, т.е. вероятность перехода системы S в любое возможное состояние в момент времени t_i определяется состоянием, достигнутым в момент времени t_{i-1}, и не зависит от того, когда и как она пришла в это состояние. Фиксированные моменты времени t_i принято называть шагами (или этапами) марковского скалярного процесса изменения состояния системы S.

Для удобства дальнейших рассуждений некоторый момент времени $t_0 < t_1$ будем считать начальным и введем случайное событие s^i , состоящее в том, что после i этапов исходная система S находится в состоянии S_k, $k = 1, 2, \ldots m$. При этом $i = 0$ соответствует начальному этапу, и при любом фиксированном i события s_k^i, $k = 1, 2, \ldots m$ образуют полную группу событий. Поэтому

$$\sum_{k=1}^{m} P\left[s_k^i\right] = 1,\ i = 1, 2, ..., N.$$

Кроме того, вектор

$$p(i) = \left(P\left[s_1^i\right], P\left[s_2^i\right], ..., P\left[s_m^i\right]\right)^T$$

является вектором вероятностей состояний системы S после i этапов при $i = 1, 2, \ldots, N$ и вектором начальных вероятностей состояний системы S при $i = 0$.

В соответствии с исходными допущениями условная вероятность реализации случайного события s_k^i при условии s_j^{i-1} зависит от принятого после (i–1)-го этапа решения $X_{n_{i-1}}$ из множества G допустимых решений.

Таким образом,

$$p_{jk}\left(i / X_{n_{i-1}}\right) = P\left[s_k^i / s_j^{i-1}; X_{n_{i-1}}\right]$$

И матрица переходных вероятностей

$$P\left(i / X_{n_{i-1}}\right) = p_{jk}\left(i / X_{n_{i-1}}\right).$$

При этом нетрудно убедиться в том, что верны следующие утверждения:

а) сумма элементов любой строки матрицы переходных вероятностей $P\left(i / X_{n_{i-1}}\right)$ равна единице;

б) вектор вероятностей состояний изучаемой системы S после i этапов равен произведению транспонированной матрицы переходных вероятностей $P\left(i / X_{n_{i-1}}\right)$ на вектор вероятностей состояний после (i -1)-го этапа, т.е.

$$p(i) = P^T\left(i / X_{n_{i-1}}\right) \cdot p(i-1).$$

Из приведенных рассуждений следует, что в общем случае вектор вероятностей состояний системы S после N этапов зависит от вектора начальных вероятностей ее состояний $p(0)$ и вектора $\left(X_{n_0} X_{n_1} ... X_{n_{N-1}}\right)^T \in G^N$, который называют вектором решений.

Действительно,

$$p(N) = P^T\left(N / X_{n_{N-1}}\right) \cdot p(N-1) = P^T\left(N / X_{n_{N-1}}\right) \cdot P^T\left(N-1 / X_{n_{N-2}}\right) \cdot p(N-2) = ...$$

$$... = P^T\left(N / X_{n_{N-1}}\right) \cdot P^T\left(N-1 / X_{n_{N-2}}\right) ... P^T\left(1 / X_{n_0}\right) \cdot p(0).$$

С переходом системы S из одного состояния в другое связана матрица дохода $R\left(i / X_{n_{i-1}}\right) = \left(r_{jk}\left(i / X_{n_{i-1}}\right)\right) \in M_m(R)$, в которой элемент $r_{jk}\left(i / X_{n_{i-1}}\right)$ есть доход (положительное значение) или убыток (отрицательное значение) за *i*-й этап. При этом доход или убыток за *i*-й этап связан лишь с переходом системы из состояния S_j, в котором она находилась после (*i*–1)-го этапа, в состояние S_k при принятии решения $X_{n_{i-1}} \in G$.

Величина

$$n_j\left(X_{n_{i-1}}\right) = \sum_{k=1}^{m} p_{jk}\left(i / X_{n_{i-1}}\right) r_{jk}\left(i / X_{n_{i-1}}\right) \qquad (32.1)$$

определяет ожидаемый доход за *i*-й этап, если после (*i*–1)-го этапа система находилась в состоянии S_j и было принято решение $X_{n_{i-1}} \in G$. Заметим, что ожидаемый доход связан лишь с переходами системы из одного возможного состояния в другое при фиксированном допустимом решении.

Далее в качестве принципа оптимальности, совпадающего в рассматриваемом случае с критерием оптимальности, использована максимизация ожидаемого дохода за N этапов. При этом специфика решения задачи прежде всего связана с тем, будет ли число этапов N

конечным или нет. В соответствии с этим рассматривают задачи принятия решений с конечным горизонтом планирования, когда $N < \infty$, или с бесконечным горизонтом планирования, когда $N = \infty$.

Необходимо отметить, что лицо, принимающее решения, может интересовать величина ожидаемого дохода при заранее определенной стратегии поведения в случае того или иного состояния системы. Так, например, лицо, принимающее решения, может считать, что если после (i–1)-го этапа система находится в состоянии S_j, то безотносительно к конкретному значению j всегда необходимо принимать решение $X^* \in G$. В этом случае говорят, что процесс принятия решений описывается стационарными стратегиями. При этом каждой стационарной стратегии будут соответствовать свои матрицы переходных вероятностей и доходов.

При конечном горизонте планирования $N < \infty$ марковскую задачу принятия решений можно представить как задачу динамического программирования.

Для доказательства введем следующее определение: **оптимальный ожидаемый доход $f_i(j)$** – это наилучший доход в смысле используемого принципа оптимальности за этапы с номерами i, i+1, ..., N при условии, что после (i –1)-го этапа изучаемая система S находится в состоянии S_j, где $j \in \{1, 2, ..., m\}$. Так как горизонт планирования конечен, то для оптимальных ожидаемых доходов должны быть выполнены естественные условия:

$$f_{N+1}(j) \equiv 0, \; j = 1, 2, ..., m.$$

Оптимальный ожидаемый доход $f_i(j)$ на этапах с номерами i, i+1,..., N складывается из двух составляющих. Первая составляющая — оптимальный ожидаемый доход на (i+1)-м этапе, обусловленный одним лишь переходом системы S из состояния S_j, в котором она находилась на i-

м этапе, в любое допустимое состояние S_k, k = 1, 2, …, m на (i + 1)-м этапе (рисунок 32.1).

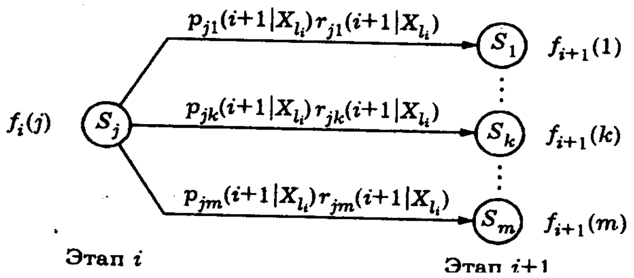

рис.32.1

Эта составляющая равна

$$\max_{X_{m_i} \in G} v_j\left(X_{n_i}\right),$$

$$v_j\left(X_{n_{i-1}}\right)=\sum_{k=1}^{m} p_{jk}\left(i+1/X_{n_i}\right) r_{jk}\left(i+1/X_{n_i}\right),$$

где $p_{jk}\left(i+1/X_{n_i}\right)$ - условная вероятность того, что после (i + 1)-го этапа система S будет находиться в состоянии S_k, если после i этапов она находилась в состоянии S_j и было принято допустимое решение Xn_i; $\kappa_{jk}\left(i+1/X_{n_i}\right)$ - доход или убыток, связанный лишь с переходом системы из состояния S_j, в котором она находилась после i этапов, в состояние S_k на (i +1)-м этапе в результате принятия решения Xn_i из множества допустимых решений G.

Вторая составляющая оптимального ожидаемого дохода $f_i(j)$ определяется совокупностью оптимальных ожидаемых доходов $f_{i+1}(k)$, k=1,2,…,m с учетом переходных вероятностей $p_{jk}\left(i+1/X_{n_i}\right), k=1,2,...,m$

$$\max_{X_{m_i} \in G} \sum_{k=1}^{m} p_{jk}\left(i+1/\,X_{n_i}\right) f_{i+1}(k).$$

В результате проведенных рассуждений мы приходим к рекуррентному уравнению динамического программирования с конечным числом этапов, связывающему оптимальные ожидаемые доходы $f_i(j)$, j = 1, 2, …, m и $f_{i+1}(k)$, k = 1, 2, …, m:

$$f_i(j) = \max_{X_{m_i} \in G} \left\{ v_j\left(X_{n_i}\right) + \sum_{k=1}^{m} p_{jk}\left(i+1/\,X_{n_i}\right) f_{i+1}(k) \right\},$$

i = 1, 2, …, N–1, j = 1, 2, …, m.

При этом напомним, что $f_{N+1}(k) \equiv 0$, k = 1, 2, …, m и

$$v_j\left(X_{n_{i-1}}\right) = \sum_{k=1}^{m} p_{jk}\left(i+1/\,X_{n_i}\right) r_{jk}\left(i+1/\,X_{n_i}\right), \; j = 1, 2, \ldots, m.$$

Рассмотрим задачу, которая может быть полезна при анализе актуальных прикладных задач (управление запасами, замена оборудования, контроль и регулирование емкости водохранилища и т.д.).

Задача 32.1. Каждый год в начале сезона садовник проводит химический анализ почвы на своем участке и по его результатам оценивает продуктивность сада на новый сезон как хорошую, удовлетворительную или плохую (таблица 32.1). Оценка состояния сада позволяет садовнику принять одно из допустимых решений: обойтись без дополнительной обработки или провести обработку, включающую, например, внесение удобрений. Это требует дополнительных затрат, а цель садовника — получить максимальный доход. Он планирует прекратить занятие садоводством через три года и за этот период хочет получить максимальный доход. Какова его оптимальная стратегия поведения?

Решение. По результатам многолетних наблюдений садовник установил, что продуктивность сада в текущем году можно считать зависящей лишь от состояния почвы в предыдущем году. Таким образом, процесс изменения состояния почвы представляет собой марковский процесс с тремя возможными состояниями и дискретным временем.

Таблица 32.1

Продуктивность почвы	Состояние системы
Хорошая	$S1$
Удовлетворительная	$S2$
Плохая	$S3$

Множество допустимых решений G={X_1, X_2}, где X_1 — решение не вносить удобрений, а X_2 — решение внести удобрения. Таким образом, для i=1, 2,…, N матрица переходных вероятностей равна

$$P\left(i / X_{n_{i-1}}\right) = \begin{cases} P_1, X_{n_{i-1}} = X_1 \\ P_2, X_{n_{i-1}} = X_2 \end{cases}.$$

Пусть в рассматриваемой задаче матрица переходных вероятностей является постоянной:

$$P_1 = \begin{pmatrix} 0,2 & 0,5 & 0,3 \\ 0 & 0,5 & 0,5 \\ 0 & 0 & 1 \end{pmatrix},$$

где номера строк и столбцов — это номера состояний системы S_1, S_2, S_3 (см. таблицу 32.1) в текущем и последующем годах, соответственно. Так, например, если в текущем (i–1)-м году состояние почвы хорошее, то вероятность ее перехода в плохое состояние в последующем i-м году равна $p_{13}\left(i\right) = P\left[s_3^i / s_1^{i-1}\right] = 0,3$.

Садовник может принять решение о применении удобрений с целью повышения продуктивности сада, т.е. с целью улучшения состояния почвы, так как в противном случае переходные вероятности не изменятся. Если же садовник принял решение о применении удобрений, то переходные вероятности задаются матрицей:

$$P_2 = \begin{pmatrix} 0,3 & 0,6 & 0,1 \\ 0,1 & 0,6 & 0,3 \\ 0,05 & 0,4 & 0,55 \end{pmatrix}$$

наглядно иллюстрирующей улучшение состояния почвы в последующие годы. А матрица дохода имеет вид

$$R\left(i / X_{n_{i-1}}\right) = \begin{cases} R_1, X_{n_{i-1}} = X_1 \\ R_2, X_{n_{i-1}} = X_2 \end{cases}, \text{ где } \quad R_1 = \begin{pmatrix} 7 & 6 & 3 \\ 0 & 5 & 1 \\ 0 & 0 & -1 \end{pmatrix}, \quad R_2 = \begin{pmatrix} 6 & 5 & -1 \\ 7 & 4 & 0 \\ 6 & 3 & -2 \end{pmatrix}.$$

В рассматриваемом случае горизонт планирования $N = 3$, а элементы матриц доходов и переходных вероятностей не зависят от номера этапа.

Воспользовавшись матрицами P_1, P_2, R_1, R_2 и их независимостью от номера этапа, вычислим ожидаемые доходы (32.1), обусловленные одним лишь переходом изучаемой системы S из одного возможного состояния в другое, при различных вариантах допустимых решений из множества G:

$$v_1\left(X_1\right) = 0,2 \cdot 7 + 0,5 \cdot 6 + 0,3 \cdot 3 = 5,3,$$

$$v_2\left(X_1\right) = 0 \cdot 0 + 0,5 \cdot 5 + 0,5 \cdot 1 = 3,$$

$$v_3\left(X_1\right) = 0 \cdot 0 + 0 \cdot 0 + 1 \cdot (-1) = -1,$$

$$v_1\left(X_2\right) = 0,3 \cdot 6 + 0,6 \cdot 5 + 0,1 \cdot (-1) = 4,7,$$

$$v_2\left(X_2\right) = 0,1 \cdot 7 + 0,6 \cdot 4 + 0,3 \cdot 0 = 3,1,$$

$$v_3\left(X_2\right) = 0,05 \cdot 6 + 0,4 \cdot 3 + 0,55 \cdot (-2) = 0,4.$$

Для наглядности воспользуемся табличным алгоритмом решения рассматриваемой задачи (таблица 32.2 - этап 3, соответствующий $f_3(j)$, таблица 32.3 - этап 2, соответствующий $f_2(j)$, таблица 32.4 - этап 1, соответствующий $f_1(j)$). При этом нумерация этапов „с конца” обусловлена определением оптимального ожидаемого дохода $f_i(j)$.

Этап 3.

Таблица 32.2

j	$v_j(X_k)$		Оптимальный ожидаемый доход $f_3(j)$	Оптимальное решение
	k=1	k=2		
1	5,3	4,7	5,3	X_1
2	3	3,1	3,1	X_2
3	-1	0,4	0,4	X_3

Этап 2.

Таблица 32.3

j	$v_j(X_k)+\sum_{l=1}^{3} p_{jl}(i+1/X_k)\cdot f_3(l)$		$f_2(j)$	Оптимальное решение
	k=1	k=2		
1	$5{,}3+0{,}2\cdot 5{,}3+0{,}5\cdot 3{,}1+$ $+0{,}3\cdot 0{,}4=8{,}03$	$4{,}7+0{,}3\cdot 5{,}3+0{,}6\cdot 3{,}1+$ $+0{,}1\cdot 0{,}4=8{,}19$	8,19	X_2
2	$3+0\cdot 5{,}3+0{,}5\cdot 3{,}1+$ $+0{,}5\cdot 0{,}4=4{,}75$	$3{,}1+0{,}1\cdot 5{,}3+0{,}6\cdot 3{,}1+$ $+0{,}3\cdot 0{,}4=5{,}61$	5,61	X_2
3	$-1+0\cdot 5{,}3+0\cdot 3{,}1+$ $+1\cdot 0{,}4=-0{,}6$	$0{,}4+0{,}05\cdot 5{,}3+0{,}4\cdot 3{,}1+$ $+0{,}55\cdot 0{,}4=2{,}125$	2,125	X_2

Этап 1.

Таблица 32.4

j	$v_j(X_k)+\sum_{l=1}^{3} p_{jl}(i+1/X_k)\cdot f_2(l)$		$f_1(j)$	Оптимальное решение
	k=1	k=2		
1	$5{,}3+0{,}2\cdot 8{,}19+0{,}5\cdot 5{,}61+$ $+0{,}3\cdot 2{,}125=10{,}38$	$4{,}7+0{,}3\cdot 8{,}19+0{,}6\cdot 5{,}61+$ $+0{,}1\cdot 2{,}125=10{,}74$	10,74	X_2
2	$3+0\cdot 8{,}19+0{,}5\cdot 5{,}61+$ $+0{,}5\cdot 2{,}125=6{,}87$	$3{,}1+0{,}1\cdot 8{,}19+0{,}6\cdot 5{,}61+$ $+0{,}3\cdot 2{,}125=7{,}92$	7,92	X_2
3	$-1+0\cdot 8{,}19+0\cdot 5{,}61+$ $+1\cdot 2{,}125=1{,}13$	$0{,}4+0{,}05\cdot 8{,}19+0{,}4\cdot 5{,}61+$ $+0{,}55\cdot 2{,}125=4{,}23$	4,23	X_2

Из оптимального решения следует, что первые два года садовник должен применять удобрения при любом состоянии системы, т.е. вне зависимости от результатов химического анализа почвы (первые два года оптимальным является допустимое решение X_2 для всех возможных состояний), но на последнем этапе (третий год) ему следует применять удобрения лишь при удовлетворительном (S_2) и плохом (S_3) состояниях почвы. В этом случае суммарный ожидаемый доход составит 10,74 при хорошем состоянии почвы в начале первого года, 7,92 — при удовлетворительном и 4,23 — при плохом состоянии почвы в начале первого года.

Список литературы.

1. *Абрамов Л.М.* Математическое программирование / Л.М. Абрамов, В.Ф. Капустин. - Л., Изд-Ленингр. ун-та, 1976.
2. *Акулич И.Л.* Математическое программирование в примерах и задачах / И.Л. Акулич. - М.: Высшая школа, 1993.
3. *Афанасьев М.Ю.* Исследование операций в экономике: модели, задачи, решения: Учебное пособие / М.Ю. Афанасьев, Б.П. Суворов. - М.: ИНФРА-М, 2003.
4. *Ашманов С.А.* Линейное программирование / С.А. Ашманов. - М.: Наука, 1981.
5. Баканов М.И. Теория экономического анализа: Учебник. -4-е изд., доп. и перераб / М.И. Баканов, А.Д. Шеремет. - М.: Финансы и статистика, 2000.
6. Баканов М.И. Экономический анализ: ситуации, тесты, примеры, задачи, выбор оптимальных решений, финансовое прогнозирование: Учеб. Пособие / М.И. Баканов, А.Д. Шеремет. - М.: Финансы и статистика, 1999.
7. *Банди Б.* Основы линейного программирования: Пер. с англ. / Б. Банди. - М.: Радио и связь, 1989.
8. *Беллман Р.* Прикладные задачи динамического программирования / Р. Беллман, С. Дрейфус; перевод с англ. Н. М. Митрофановой. — М.: Наука, 1965.
9. *Вентцель Е.С.* Элементы динамического программирования / Е.С. Вентцель.- М.: Наука,1964.
10. *Гасс С.* Линейное программирование / С. Гасс. - М.: Физматгиз, 1961.
11. *Габасов Р.* Методы линейного программирования. Ч.1. Общие задачи / Р. Габасов, Ф.М. Кириллова. – Минск: Изд-во БГУ им. В.И. Ленина, 1977.

12.Габасов Р. Методы линейного программирования. Ч.2. Транспортные задачи / Р. Габасов, Ф.М. Кириллова. – Минск: Изд-во БГУ им. В.И. Ленина, 1977.
13.*Глухов В.В.* Математические методы и модели для менеджмента / В.В. Глухов, М.Д. Медников, С.Б. Коробко - СПб.: Издательство “Лань”, 2000.
14.*Гольштейн Е.Г.* Линейное программирование, теория, методы и приложения / Е.Г. Гольштейн, Д.Б. Юдин. - М.: Наука, 1969.
15.*Грешилов А.А.* Прикладные задачи математического программирования: Учеб. Пособие/ А.А. Грешилов. – 2-е изд. – М.: Логос, 2006.
16.*Грин, Д.* Математические методы анализа алгоритмов / Д. Грин. - Д.~Кнут. — М.:Мир, 1987.
17.*Дудорин В.И.* Моделирование в задачах управления производством/ В.И. Дудорин.-М.:Статистика, 1980.
18.Исследования операций в экономике: учебное пособие для ВУЗов / под ред. Кремера Н.Ш. –М.: Банки и Биржи , ЮНИТИ, 1997.
19.Жолобов Д. А. Введение в математическое программирование: Учебное пособие/ Д. А. Жолобов. – М.: МИФИ, 2008.
20.*Карасев А.И.* Математические методы и модели в планировании / А.И. Карасев, Н.Ш. Кремер, Т.И. Савельева.-М.: Экономика, 1987.
21.*Карманов В.Г.* Математическое программирование/ В.Г. Карманов. –М.: Наука, 1986.
22.*Ковалев М.М.* Дискретная оптимизация. Целочисленное программирование/ М.М. Ковалев. – М.:"Едиториал УРСС", 2003.
23.*Колемаев В.А.* Математическая экономика/ В.А. Колемаев. - М.: Юнити,1998.
24.*Красс М.С.* Математические методы и модели для магистрантов экономики: Учебное пособие / М.С. Красс, Б.П. Чупрынов. – СПБ.: Питер, 2006.

25.*Кузнецов А.В.* Высшая математика. Математическое программирование / А.В. Кузнецов, В.А. Сакович, Н.И. Холод. - М.: Выш. шк., 1994. – 286 с.: ил.

26.*Кузнецов Ю.Н.* Математическое программирование: Учеб. Пособие / Ю.Н. Кузнецов, В.И. Кузубов, А.Б. Волощенко / 2-е изд., перераб. и доп. – М.: Высш. школа, 1980.

27.*Лотов А.В.* Введение в экономико-математическое моделирование/ А.В. Лотов.-М.: Наука, 1984.

28.*Ляшенко И.Н.* Линейное и нелинейное программирование / И.Н, Ляшенко, Е.А, Карагодова, Н.В. Черникова, Н.З. Шор. – Киев.: Издательское объединение “Вища школа”, 1975.

29.*Морозов В.В.* Исследование операций в задачах и упражнениях / В.В. Морозов, А.Г. Сухарев, В.В. Федоров. - М.,1986.

30.*Пантелеев А.В.* Методы оптимизации в примерах и задачах: Учебное пособие / А.В. Пантелеев, Т.А. Летова. - М.: Высшая Школа, 2005.

31.*Пападимитриу Х.* Комбинаторная оптимизация. Алгоритмы и сложность. / Х. Пападимитриу, К. Стайглиц. — М.: Мир, 1985.

32.*Ромакин М.И.* Оптимизация планирования производства: экономико-математические модели и методы/ М.И. Ромакин.-М.: Финансы и статистика, 1981.

33.*Солодовников А.С.* Введение в линейную алгебру и линейное программирование/ А.С. Солодовников. - М.: Изд. “Просвещение”, 1966.

34.*Схрейвер А.* Теория линейного и целочисленного программирования: В 2-х т. Т.1: Пер с англ. / А. Схрейвер. - М.: Мир, 1991.

35.*Таха Х.А.* Введение в исследование операций. Кн.1 и 2/ Х.А. Таха. - М.:Мир, 1985.

36.Терехов Л.Л. Экономико-математические методы.- М.: Статистика, 1972.

37. *Тынкевич М.А.* Экономико-математические методы (исследование операций). Изд. 2, испр. и доп. / М.А. Тынкевич. - Кемерово, 2000.
38. *Хедли Дж.* Нелинейное и динамическое программирование/ Дж. Хедли. - М.: Мир, 1967.
39. *Черноруцкий И.Г.* Методы оптимизации в теории управления: Учеб. Пособие / И.Г. Черноруцкий. – СПб.: Питер, 2004.
40. Y(x).ru Сервис онлайн построения графиков. Предназначен для онлайн построения графиков функций (обычных и параметрических) и графиков по точкам (графиков по значениям). Режим доступа: http://www.yotx.ru Дата обращения 12.08.2016.
41. Математическое бюро МатБюро[Электронный ресурс]/ Полезные материалы: учебники, программы, сайты, формулы – Режим доступа: http://www.matburo.ru/useful.php. Дата обращения 12.08.2016.
42. Экономико-математические методы. Электронный учебник. Режим доступа: http://www.math.mrsu.ru/text/method/index.htm Дата обращения 12.08.2016.

Приложение. Великий русский математик, академик Императорской Санкт-Петербургской Академии наук А.А. Марков

Андрей Андреевич Марков (2 июня 1856 – 20 июля 1922) – русский математик, академик, внёсший большой вклад в теорию вероятностей, математический анализ и теорию чисел.

Марков, родился в Рязани. Он был сыном чиновника Андрея Григорьевича Маркова, служившего в Лесном департаменте в чине коллежского советника,

В 1874 году Марков поступил в Императорский Санкт-Петербургский университет, где слушал лекции Чебышева, влияние которого отразилось на всей научной деятельности. Слушал он также лекции профессоров Коркина и Золотарева, которые помимо лекций вели кружковые занятия с лучшими студентами. 31 мая 1878 года Марков окончил Императорский Санкт-Петербургский университет по математическому разряду физико-математического факультета со степенью кандидата. В том же году был награжден золотой медалью за сочинение на предложенную факультетом тему "Об интегрировании дифференциальных уравнений при помощи непрерывных дробей" и был оставлен при Императорском Санкт-Петербургском университете «для приготовления к профессорскому званию».

В 1880 году Марков защитил знаменитую магистерскую диссертацию "О бинарных квадратичных формах положительного определителя", сразу выдвинувшую его в первые ряды русских математиков. В 1881 году он защитил докторскую диссертацию "О некоторых приложениях алгебраических непрерывных дробей".

В 1880 году началась преподавательская деятельность Маркова в Императорском Санкт-Петербургском университете в качестве приват-доцента. В учебных 1880/81 и 1881/82 годах он читал повторительный курс дифференциального и интегрального исчислений. В 1883 году ему был передан курс "Введения в анализ", до того читавшийся профессорами

Сохоцким и Поссе. В том же году из университета ушел Чебышев, и Марков первый раз читал курс теории вероятностей. С учебного 1885/86 года он читал этот курс непрерывно из года в год.

13 декабря 1886 года, по предложению Чебышева, Марков был избран адъюнктом Императорской Санкт-Петербургской Академии наук, 3 марта 1890 года – экстраординарным академиком, 2 марта 1896 года – ординарным академиком. В 1886 году он был назначен экстраординарным профессором Императорского Санкт-Петербургского университета, а в 1893 году – ординарным. В 1905 году Марков вышел в отставку со званием заслуженного профессора, но курс теории вероятностей продолжал читать практически до самой своей смерти.

Марков является первооткрывателем обширного класса стохастических процессов с дискретной и непрерывной временной компонентой, названных его именем. Марковские процессы обладают следующим (марковским) свойством: следующее состояние процесса зависит, вероятностно, только от текущего состояния. В то время, когда эта теория была построена, она считалась весьма абстрактной, однако в настоящее время практические применения данной теории чрезвычайно многочисленны. Теория цепей Маркова выросла в огромную и весьма важную область научных исследований – теорию марковских случайных процессов, которая в свою очередь представляет основу общей теории стохастических процессов. Марков существенно продвинул классические исследования предшественников, касающиеся закона больших чисел и центральной предельной теоремы теории вероятностей, а также распространил их и на цепи Маркова.

Марков своим открытием (как и затем Колмогоров, предложивший строгую теоретико-вероятностную формулировку на основе теории меры) сделал крупнейший вклад в теорию случайных процессов и теорию вероятностей вообще.

Как и все выдающиеся русские математики, Андрей Андреевич Марков занимался многими проблемами математического анализа. В общем списке его научных трудов работы по математическому анализу составляют более одной третьей части. Внимание Маркова привлекали теория непрерывных дробей, исчисление конечных разностей, теория интерполирования функций, экстремальные задачи в функциональных пространствах, проблема моментов, теория ортогональных многочленов, квадратурные формулы, дифференциальные уравнения, теория функций, наименее уклоняющихся от нуля, и другие вопросы. По многим разделам математического анализа Марков получил выдающиеся результаты, которые играют важную роль и в наши дни. Марков воспринял идеи своего учителя Чебышева и занимался решением многих задач, поставленных в его трудах. Классические работы Чебышева и Маркова о предельных величинах интегралов составили основы теории моментов и теории экстремальных задач в функциональных пространствах.

Работ по теории чисел у Маркова сравнительно немного – 15, но они имеют непреходящее значение для этой теории. Сюда относится, прежде всего, магистерская диссертация «О бинарных квадратичных формах положительного определителя». Она примыкала к исследованиям Коркина и Золотарева и была высоко оценена Чебышевым. Диссертация посвящена проблеме арифметических минимумов неопределенных бинарных квадратичных форм. В последующих статьях рассматривается проблема арифметических минимумов неопределенных тернарных и кватернарных квадратичных форм. Идеи и результаты Маркова оказали большое влияние на дальнейшее развитие теории чисел.

Имя Маркова носят следующие математические объекты:

- неравенство Маркова
- теорема Маркова
- цепь Маркова
- эргодическая теорема Маркова

- марковский процесс
- марковское свойство
- марковский процесс принятия решений.