

УДК: 65.012.26

DOI: 10.52531/1682-1696-2022-22-4-88-95

Научная статья

# ДИНАМИЧЕСКОЕ ПРОГРАММИРОВАНИЕ ПРИ УПРАВЛЕНИИ ПРОЕКТОМ

А. В. ТИХОМИРОВ<sup>1</sup>,  
А. В. МАРАХОВСКИЙ<sup>2</sup>,  
Н. П. ПЕЧАЛИН<sup>2</sup>

<sup>1</sup> АО «АВТОВАЗ»<sup>2</sup> АО «ЦЕНТРАЛЬНЫЙ НАУЧНО-  
ИССЛЕДОВАТЕЛЬСКИЙ РАДИОТЕХНИЧЕСКИЙ  
ИНСТИТУТ ИМ. АКАДЕМИКА А. И. БЕРГА»

В статье проводится обзор методов управления проектом и рассматриваются способы динамического программирования для оптимизации проекта.

**Ключевые слова:** управление проектом, диаграмма Ганта, динамическое программирование, календарный план

## АКТУАЛЬНОСТЬ

Современная трактовка управления проектами связана с совокупностью процессов, задаваемой пересечением групп процессов управления проектом (инициация, планирование, исполнение, контроль, завершение) и функциональных (предметных) областей управления проектами (управление интеграцией проекта, содержанием, сроками, стоимостью, качеством, человеческими ресурсами, коммуникациями, рисками, поставками и контрактами) [1, 2]. Данные базовые понятия могут быть расширены ключевыми терминами, к которым можно отнести: группа процессов управления проектом; жизненный цикл проекта; заинтересованная сторона; план управления проектом. В группы процессов управления проектами входят процессы: инициации, планирования, исполнения, мониторинга и управления, а также завершения. Группы процессов управления проектами не являются фазами проекта. Основные принципы проектного управления сформировались как результат обобщенного опыта и системного подхода к «механизмам управления». Как основные правила, принципы управления проектами вытекают из закономерностей, которые в свое время привели к успеху многочисленные решения. В состав основных принципов, как правило, включаются [2]:

Original article

## DYNAMIC PROGRAMMING IN PROJECT MANAGEMENT

A. V. TIKHOMIROV<sup>1</sup>,  
A. V. MARAKHOVSKY<sup>2</sup>, N. P. PECHALIN<sup>2</sup>

<sup>1</sup> JSC «AVTOVAZ»<sup>2</sup> JSC «CENTRAL RESEARCH INSTITUTE  
OF RADIO ENGINEERING NAMED AFTER  
ACADEMICIAN A. I. BERG»

The article provides an overview of project management methods and discusses ways of dynamic programming to optimize the project.

**KEY WORDS:** project management, Gantt chart, dynamic programming, calendar plan

1. Принцип дифференцированного подхода.
2. Принцип экономической целесообразности.
3. Принцип гибкости.
4. Принцип конкурентоспособности.
5. Принцип разделения полномочий.
6. Принцип открытости.
7. Принцип лучших практик.

Общепринятым на современном этапе подходом к управлению проектами является процессный подход. Процесс – это ряд взаимосвязанных действий и операций, выполняемых для достижения заранее определенных продуктов, результатов или услуг. Процессы управления проектом выполняются командой проекта и обычно бывают двух типов.

1. Процессы управления проектом, общие для большинства проектов, как правило, нацелены на выполнение общей задачи. Такой задачей может быть инициация, планирование, исполнение, мониторинг и управление, а затем и закрытие проекта. Эти процессы взаимодействуют между собой сложным образом, это нельзя полностью объяснить в документе или с помощью рисунков. Взаимодействие процессов может также затрагивать содержание, стоимость, расписание проекта и т.д. Данные элементы называются областями знаний.

2. Процессы, ориентированные на продукт, определяют и создают продукт проекта. Они обычно определяются через жизненный цикл проекта и меняются

в зависимости от области приложения. Например, содержание проекта не может быть определено без понимания основ того, как производить указанный продукт.

Областью управления проектами является подсистема управления временем проекта. Управление временем осуществляется на всех этапах жизненного цикла проекта, реализуясь в различных функциях проект-менеджмента. На этапе разработки проекта – это планирование времени проекта, на этапе реализации – контроль выполнения сетевого графика и внесение изменений по ходу осуществления проекта.

## МЕТОДЫ ИССЛЕДОВАНИЯ

Главной задачей управления временем на этапе планирования является разработка такого расписания работ, при котором целевая функция задачи при соблюдении всех условий достигала бы экстремального значения. Иными словами, главная задача календарного планирования интегрирует в себе достижение трех условий:

- минимизация продолжительности проекта в условиях ограниченности ресурсов;
- минимизация стоимости проекта;
- равномерное распределение ресурсов.

Итогом выполнения главной задачи планирования времени является обоснованный календарный план. Календарный план – это проектно-технологические документы, устанавливающие полный перечень работ проекта, их последовательность, взаимосвязь, сроки выполнения, продолжительность, исполнителей и ресурсы, необходимые для выполнения работ. Создание календарного плана предполагает ряд предварительных действий:

- 1) определение продолжительности работ;
- 2) установление взаимосвязи между работами;
- 3) определение времени доступности всех видов ресурсов.

Процесс определения продолжительности работ может осуществляться различными методами, в частности экспертными методами, с использованием баз данных, с помощью внутренних и внешних консультантов, существующих стандартов и т. д.

При этом методе предполагается возможным учесть основные факторы продолжительности работы: трудоемкость, количество исполнителей, чистое время задержки. Затраты проекта рассчитываются сразу после того, как только в расписание проекта введены основные работы, длительность которых зависит от количества назначаемых на них ресурсов. Рабочие часы проекта содержатся в его календарном плане, а система управления затратами рассчитывает все прямые издержки, дополнительные выплаты, накладные, общие и управленческие расходы и другие текущие затраты, возникающие в процессе выполнения

работ. Затем, элементы, отсутствующие в расписании, вносятся в систему управления затратами. Для проведения анализа состояния работ и затрат необходимо иметь следующие величины:

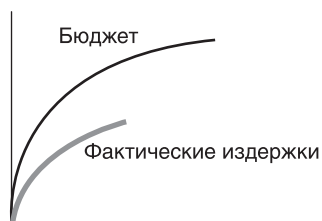
- плановые затраты (бюджет проекта),
- фактическая выработка,
- фактические затраты,
- прогнозные значения затрат.

Для получения четкой картины состояния проекта необходимо получить все эти четыре величины. Простой анализ плановых затрат, в сравнении с фактическими, зачастую может давать искаженную картину. Например, если фактические затраты составляют 90% от плановых, то может показаться, что выполнение контракта идет очень хорошо. В то время, как состояние проекта или его фактическая выработка, добавленные в анализ, могут показать, что выполнена только половина из запланированных на данное число работ. Таким образом, выполнение контракта отстает от календарного графика и, кроме того, затраты на выполненную часть работ превышают плановые (см. рис. 1).

Таким образом, существуют исходный и утвержденный планы затрат. Утвержденный план используется при расчетах фактической выработки и, соответственно, на его основе производится анализ состояния работ и затрат. Эти различные планы затрат обычно анализируются после завершения контракта, когда предпринимается попытка извлечения уроков из проекта.

Фактическая выработка позволяет нам количественно оценить прогресс проекта в денежном выражении. Теория управления проектами рекомендует использовать величину фактической выработки при анализе состояния работ и затрат крупных проектов.

Прогнозные значения затраты – это неотъемлемая часть анализа месячного цикла. В ходе управления затратами проекта менеджер может в любой момент оценить затраты на оставшуюся часть работ календарного плана. В системе управления затратами должна быть предусмотрена возможность получения и хранения набора прогнозов, учитывающих будущий рост стоимостей (рис. 2).



Эта схема показывает, что фактические затраты меньше, чем предусмотренные бюджетом. Без поступлений невозможно сказать, что фактические затраты просто меньше, потому что продвигающаяся работа находится на уровне ниже, чем запланированная

РИС. 1.

График выполнения контракта

Наличие множества прогнозов обеспечивает достоверное определение состояния проекта, выявление возможного перерасхода на ранних этапах и приводит к эффективному управлению проектом.

Фактические затраты необходимы, чтобы отчет о фактических затратах был составлен таким образом, что аналитик мог легко сопоставить фактические и плановые затраты. Планирование затрат и ввод фактических данных по затратам должен осуществляться для одних и тех же уровней иерархической структуры работ и ресурсов. Эта технология позволяет провести точный анализ фактических и плановых данных.

Бюджет, фактические затраты, фактическая выработка и прогнозы должны анализироваться в совокупности, чтобы получить действительную картину состояния проекта. Для быстрого получения информации о состоянии проекта можно использовать так называемую S-кривую (график каждой из вышеперечисленных характеристик рис. 3).

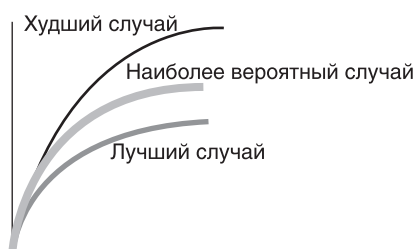
Контроль изменений – неотъемлемая часть управления проектом. Изменения должны вноситься в расписание оперативно с тем, чтобы руководитель проекта мог продолжать управлять выполнением работ.

Контроль изменений – это процесс, который необходимо планировать до начала реализации проекта.

Для того чтобы в будущем иметь возможность проследить ход изменений исходного плана, необходимо протоколировать все имеющиеся место модификации. В данный протокол должна вноситься следующая информация о проводимом изменении:

- номер счета затрат;
- изменяемая сумма;
- номера счетов, с и на которые происходит перенос затрат;
- имя пользователя, вносящего изменения в исходный план.

В случае утверждения изменений, они вносятся в исходный план. Сохранение нескольких исходных планов позволяет произвести глубокий анализ завершенного проекта и извлечь из него полезные уроки для будущих контрактов [4, 6, 8].



Эта схема показывает три совмещенных на одном графике прогноза

РИС. 2.

Совмещение проектов

Когда проект завершен, надо выполнить следующее:

- рассчитать фактическую выработку;
- внести фактические данные по накладным, общим, административным и пр. затратам;
- подготовить финальные отчеты;
- произвести анализ отчетов и извлечь из завершенного проекта полезную информацию на будущее;
- поместить проект в архив для использования его при выполнении будущих контрактов.

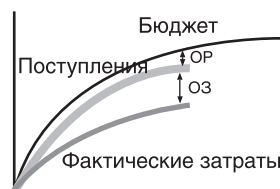
Таким образом, для того чтобы провести соответствующий анализ выполненных работ по контракту недостаточно одного лишь сравнения плановых затрат с их фактическими значениями. В частности, также надо учитывать и календарный график работ.

Интеграция календарного плана работ и затрат является одной из лучших на сегодняшний день методологий, обеспечивающей всей необходимой информацией для проведения анализа проекта. Интеграция систем управления затратами и расписанием избавляет от необходимости повторного ввода данных и обеспечивает возможность оперативного контроля для проверки отчетов. Этот метод даст менеджерам проекта инструмент для эффективного управления проектами как альтернативу простому написанию отчетов в конце отчетного периода.

Использование описанных управленческих процессов/методологий имеет множество положительных сторон. Вот некоторые из них:

- оценка стоимости проекта осуществляется исходя из набора работ, требуемых для выполнения контракта. при этом возможно внесение изменений в условия контракта с целью сведения затрат по проекту к установленной сумме;
- метод фактической выработки позволяет провести глубокий анализ проекта;
- поддержка исходного плана и процесс контроля изменений помогают в управлении проектом и проведении анализа результатов при завершении контракта.

Возможность составления прогнозов крайне важ-



Эта схема показывает отклонения от расписания (ОР), разницу между поступлениями и бюджетом; и отклонение затрат (ОЗ), разницу между поступлениями и фактическими затратами

РИС. 3.

S – кривая

на для эффективного управления проектом. Чтобы качественно планировать проекты и чётко следовать графику работ, можно использовать методический инструмент диаграмм Ганта [7], которые, по сути, являются отображениями сквозных планов при планировании. Диаграмма Ганта – это горизонтальная столбчатая диаграмма, которая иллюстрирует хронологию проекта и всех его задач, упрощая управление. Она даст возможность в наглядной форме просматривать график продвижения работы над проектами, предстоящие вехи и хронологию в целом. То есть диаграмма Ганта представляет собой визуальный способ отображения запланированных задач, что позволяет наглядно изобразить составляющие части проекта и разбить его на задания меньшего размера для удобства управления.

Получившиеся задания размещаются вдоль временной шкалы диаграммы Ганта, после чего на нее добавляются зависимости между заданиями, исполнителями и контрольные точки. Стоит отметить, что на графике размещают и другую информацию: например, устанавливают проценты выполнения, формируют бюджет, помечают дедлайны – вехи.

Таким образом график работ состоит из ряда отрезков, размещённых вдоль временной оси. Каждый из них соответствует отдельной задаче или подзадаче. Начало и конец отрезка соответствуют моменту начала и завершения работы по задаче. Длина отрезка – продолжительность работ. На каждый отрезок целесообразно назначить исполнителей задачи. Диаграммы Ганта можно использовать для самых разных проектов и программ, однако полезно будет понять, в каких случаях они чаще всего применяются и почему:

- Подготовка сложных проектов и управление ими. Чем крупнее проект, тем больше задач, которыми нужно управлять. С помощью диаграмм Ганта можно с лёгкостью визуализировать проекты и разбивать их на более мелкие задачи.
- Отслеживание зависимостей между задачами. Проекты могут идти не по плану. Диаграммы Ганта помогают автоматизировать зависимые элементы задач, благодаря чему следующий этап работы не начинается раньше, чем закончен предыдущий.
- Отслеживание хода работ по проекту. Диаграммы Ганта позволяют отслеживать прогресс и вехи, благодаря чему можно с лёгкостью скорректировать план проекта в случае необходимости.

При этом целесообразно выделить такие достоинства диаграммы Ганта, как визуализация хронологии проекта; возможность получить общее представление о хронологии проекта; простая организация сложных задач; помощь в управлении ресурсами; координация работы; оценка загрузки сотрудников; определение критического пути; быстрое нахождение связанных задач; онлайн-диаграмму легко обновить – добавить новый этап работы или изменить срок задачи, если

того требует ситуация. Таким образом, диаграмма Ганта является дорожной картой проекта и инструментом, помогающим отслеживать достижение всех вех, проводить оценку укладывается ли вы в эти сроки. Она позволяет упорядочить задачи – в том числе, если нужно привлечь несколько команд. Диаграмма Ганта отлично справляется там, где есть чёткие дедлайны, ограниченный бюджет и ресурсы. Поэтому её можно назвать основным инструментом управления проектами. При этом подразумевается, что диаграмма Ганта в контексте управления проектом обладает свойством иерархичности, т.е. основная задача управления проектами в результате декомпозиции может разбиваться на подэтапы, которые сами в свою очередь представляются в виде диаграммы Ганта. Такой подход позволяет сводить общую многокритериальную задачу к совокупности частных однокритериальных, решение которых осуществимо, например, методом динамического программирования, на каждом иерархическом уровне. Динамическое программирование – это метод нахождения оптимальных решений в задачах с многошаговой (многоэтапной) структурой. Многие процессы расчлняются на шаги естественным образом. Это все процессы планирования и управления, развиваемые во времени. Естественным шагом в них может быть год, квартал, месяц, декада, неделя, день и т. д. Однако метод динамического программирования может использоваться при решении задач, где время вообще не фигурирует, разделение на шаги в таких задачах вводится искусственно. Поэтому «динамика» задач динамического программирования заключается в методе решения.

## СОДЕРЖАНИЕ ИССЛЕДОВАНИЯ

Общая математическая постановка задачи определения функции синтеза управления проектом создания образцов может быть представлена как вектор-функция фазовых переменных  $X^*(t) \in R^n$  и вектор-функция управлений  $u \times(t) \in R^m$  при  $t \in [t_0, T]$ , доставляющих максимум функционалу:

$$\Phi = \Phi(X, u, t) \rightarrow \max \quad (1)$$

при дифференциальных связях  $\dot{X}(t) = f(X, u, t)$ , ограничениях вдоль траектории  $(X, u, t) \in G$  и краевых условиях

$$(X, t_0) \in E_0, (X, T) \in E_T \quad (2)$$

где  $G$  – некоторая область пространства,  $E_0, E_T$  – некоторые многообразия,  $[t_0, T]$  – интервал планирования,  $f(X, u, t) = \{f^1(X, u, t), \dots, f^n(X, u, t)\}$  – вектор-функция.

Таким образом, математической постановке задачи формирования решений по каждому проекту создания образцов, в данном случае, содержание и вид вектор-функций фазовых переменных  $X(t) \in R^n$ , управлений

$u(t) \in R^m$ , функционала  $\Phi = \Phi(X, u, t)$ , дифференциальных связей  $\dot{X}(t) = f(X, u, t)$ , ограниченный  $(X, u, t) \in G$  и краевых условий  $(X, t_0) \in E_0, (X, T) \in E_T$  определяются в зависимости от целевой направленности и условий реализации конкретных проектов. Для решения задачи определения функции синтеза управления проектом создания образцов в соответствии с выражениями (1–2) целесообразно определить вектор-функции управлений  $u \times (t) \in R^m$  при  $t \in [t_0, T]$ , доставляющих максимум функционалу  $\Phi = \Phi(X, u, t) \rightarrow \max$ , при  $\dot{x} = F(x, u, t)$  и ограничениях типа  $C(x_{opt}) \leq C_{дон}$ . Тогда начальными исходными данными для решения данной задачи являются

$$\begin{aligned} x &= (x_1, x_2, \dots, x_n) \\ u &= (u_1, u_2, \dots, u_m) \\ t &= (t_1, t_2, \dots, t_n) \\ x &\in X: IR^n \rightarrow R^l \\ u &\in U: IR^n \rightarrow R^l \\ t &\in T: IR^n \rightarrow R^l \\ K &\in IN \end{aligned}$$

Само решение в результате применения изложенного ниже методического материала определяется точкой  $B_{opt}$  за время  $t_k$ . Для достижения сформулированной цели – нахождения оптимального управления проектом создания образцов может решаться задача определения пошагового управления, например, может решаться задача динамического программирования. Динамическое программирование – подход, позволяющий решать задачи оптимизации, которые могут быть сформулированы как задачи многошагового оптимального управления некоторой системой. Как правило, такие задачи обладают аддитивной или мультипликативной целевой функцией, то есть

$$f(x) = \sum_{k=1}^n f_k(x_k) \text{ или } f(x) = \prod_{k=1}^n f_k(x_k), x = (x_1, \dots, x_n).$$

Обычно предполагается, что все функции  $f_k$  принимают только положительные значения, но никакие условия регулярности (непрерывность, дифференцируемость, выпуклость и т.п.) на них не накладываются. Суть метода заключается в сведении исходной задачи к упорядоченной совокупности задач меньшей размерности и последовательного их решения. При этом на каждом шаге при решении очередной задачи используются результаты решения предыдущих задач. Поскольку любая подзадача определяет некоторое множество допустимых решений, метод динамического программирования относится к комбинаторным методам оптимизации, то есть представляет собой метод упорядоченного перебора вариантов.

В обобщенном виде метод динамического программирования, для оптимизации управляемой системы, заключается в следующем [9]. Рассматривается некоторый объект (управляемая система), текущее состояние которого описывается с помощью набора параметров  $\xi$ . Множество всех возможных состояний  $\Xi$

конечно. Изменение состояний системы осуществляется посредством управляющих воздействий, которые представляют собой выбор элемента  $x$  (управления) из некоторого множества  $X$ , называемого множеством управлений. Без ограничения общности можно считать, что  $X \subseteq R^l$ . В состоянии  $\xi$  доступно конечное множество управлений  $X(\xi) \subseteq X$ . Предполагается, что в системе отсутствует последствие, то есть состояние  $\xi'$ , в которое переходит система после выбора управления, зависит только от этого управления и непосредственно предшествующего состояния:

$$\xi' = \varphi(\xi, x), x \in X(\xi). \quad (3)$$

Уравнения (3) называются уравнениями состояния. Траекторией системы называется конечная последовательность  $\xi = (\xi_1, \dots, \xi_k)$  состояний системы такая, что

$$\xi_{k+1} = \varphi(x, \xi_k), x \in X(\xi_k), k = 1, \dots, m-1.$$

Управление, посредством которого система переходит из состояния  $\xi_k$  в состояние  $\xi_{k+1}$ , обозначим через  $x_k$ . Стратегией управления для траектории  $\xi = (\xi_1, \dots, \xi_m)$  называется вектор допустимых управлений  $x = (x_1, \dots, x_{m-1})$ . Очевидно, траектория однозначно определяется своим начальным состоянием и выбором стратегии. Эффективность управления  $x \in X(\xi)$  оценивается с помощью неотрицательной функции стоимости  $f(x, \xi)$ , зависящей не только от применяемого управляющего воздействия, но и от текущего состояния системы. Стоимость траектории определяется как сумма стоимостей отдельных её этапов, то есть

$$f(\xi) = \sum_{k=1}^{m-1} f(x_k, \xi_k).$$

По определению траектории, её стоимость зависит только от её начального состояния и набора пошаговых управлений. Пусть заданы начальное состояние системы  $\xi_0$  и конечное состояние  $\xi_T$ . Связывающая их траектория  $(\xi_0, \dots, \xi_T)$  называется полной. Обычно требуется построить полную траекторию, имеющую наибольшую или наименьшую стоимость. Для определенности будем рассматривать задачу на минимум. Стратегия, определяющая оптимальную полную траекторию, называется оптимальной стратегией. Метод динамического программирования – это метод пошаговой оптимизации, поэтапного построения оптимального управления. В его основе лежит принцип оптимальности, впервые сформулированный Р. Беллманом в 1953 г. следующим образом – оптимальное поведение обладает тем свойством, что каковы бы ни были первоначальное состояние и решение в начальный момент, последующие решения должны составлять оптимальное поведение относительно состояния, получающегося в результате первого решения. Принцип может быть обоснован доказательством от противного. Пусть, далее,  $(\xi_0, \xi_1, \xi_2, \dots, \xi_T)$  – полная

оптимальная траектория. В соответствии с принципом оптимальности, траектория  $(\xi_1, \xi_2, \dots, \xi_T)$  будет иметь оптимальную стоимость среди всех траекторий, соединяющих  $\xi_1$  и  $\xi_T$ . Приняв  $\xi_1$  за начальное состояние и применяя принцип оптимальности получим, что траектория  $(\xi_2, \dots, \xi_T)$  – лучшая среди траекторий, соединяющих  $\xi_2$  и  $\xi_T$ , и так далее, если траектория оптимальна, то любой её заканчивающийся в  $\xi_T$  отрезок также оптимален. Аналогично обосновывается оптимальность любого отрезка оптимальной траектории, начинающегося с  $\xi_0$ . Пусть  $\xi$  – произвольное состояние системы, через  $F^T(\xi)$  обозначим стоимость оптимальной траектории, соединяющей  $\xi$  с финальным состоянием  $\xi_T$ . Очевидно,  $F^T(\xi_T)=0$ ,  $F^T(\xi_0)$  – стоимость оптимальной полной траектории. Из принципа оптимальности Беллмана вытекает, что

$$F^T(\xi) = \min_{x \in X(\xi)} \{f(x, \xi) + F^T(\phi(x, \xi))\}. \quad (4)$$

Обозначим теперь через  $F^0(\xi)$  стоимость оптимальной траектории, соединяющей начальное состояние  $\xi_0$  с состоянием  $\xi$ . Очевидно,  $F^0(\xi_0) = 0$ , а стоимость оптимальной полной траектории равна  $F^0(\xi_T)$ . Пусть  $\Gamma(\xi)$  – множество состояний, непосредственно предшествующих состоянию  $\xi$ , то есть таких состояний  $\xi'$ , что существует управление  $x(\xi', \xi) \in X(\xi')$ , переводящее систему из  $\xi'$  в  $\xi$ ,  $\xi = \phi(x(\xi', \xi), \xi')$ . В соответствии с принципом оптимальности,

$$F^0(\xi) = \min_{\xi' \in \Gamma(\xi)} \{f(x(\xi', \xi), \xi) + F^0(\xi')\}. \quad (5)$$

Функции  $F^T(\xi)$  и  $F^0(\xi)$  называются функциями Беллмана, а рекуррентные соотношения (4) и (5) – уравнениями Беллмана. В основе алгоритма динамического программирования могут лежать либо соотношения (4), тогда он называется методом обратной прогонки, либо соотношения (5), и алгоритм называют методом прямой прогонки.

Метод обратной прогонки (обратный метод Беллмана) осуществляется следующим образом. Последовательно вычисляются значения функции Беллмана:

$$F^T(\xi_T) = 0, F^T(\xi) = f(x, \xi, \xi_T, \xi) \text{ для } \xi \in \Gamma(\xi_T),$$

далее, с помощью соотношений (4) постепенно определяются  $F^T(\xi)$  для таких состояний  $\xi$ , что найдены значения функции Беллмана для всех состояний  $\phi(x, \xi)$ ,  $x \in X(\xi)$ , и, в конце концов, вычисляется  $F^T(\xi_0)$ . Пусть  $x(\xi) \in X(\xi)$  – управление, минимизирующее правую часть в (4), оно называется условным оптимальным управлением. Тогда оптимальная полная траектория  $(\xi_0, \xi_1, \xi_2, \dots, \xi_T)$  восстанавливается следующим образом:

$$\xi_0, \xi_1 = \phi(x(\xi_0), \xi_0), \xi_2 = \phi(x(\xi_1), \xi_1), \dots$$

При осуществлении метода прямой прогонки определяются значения функции  $F_0(\xi)$  в следующем порядке:

$$F^0(\xi_0) = 0, F^0(\xi) = f(x, \xi) \text{ для } x \in X(\xi_0), \xi = \phi(x, \xi_0),$$

далее, на основании соотношений (5), последовательно вычисляются  $F^0(\xi)$  для таких состояний  $\xi$ , что найдены значения функции Беллмана для всех состояний из  $\Gamma(\xi)$ , и, в конце концов, определяется  $F^0(\xi_T)$ .

Пусть  $\gamma(\xi) \in \Gamma(\xi)$  – состояние, минимизирующее правую часть в (5). Тогда условные оптимальные управления –  $x(\gamma(\xi), \xi)$ . Оптимальная траектория восстанавливается, начиная с финального состояния:

$$\xi_T, \xi_{T-1} = \gamma(\xi_T), \dots$$

Таким образом, во время реализации алгоритма динамического программирования дважды осуществляется многошаговый процесс: сначала находятся условные оптимальные управления и значения функции Беллмана от конечного состояния до начального или от начального состояния до конечного, затем восстанавливается оптимальная траектория и оптимальная стратегия в обратном направлении – от начала к концу или от конца к началу соответственно. Число элементарных операций, необходимых для реализации обратного и прямого методов Беллмана, квадратично зависит от числа состояний системы, причем большая часть вычислений приходится на этап условной оптимизации. Несмотря на одинаковую вычислительную сложность, алгоритм обратной прогонки использует чаще, чем алгоритм прямой прогонки.

Многие задачи динамического программирования формулируются таким образом, что система переводится из начального состояния в финальное за фиксированное число  $n$  шагов (этапов), то есть стратегия управления –  $n$ -мерный вектор  $x = (x_1, \dots, x_n)$ , где  $x_k$  – управление, выбираемое на  $k$ -м шаге,  $k = 1, \dots, n$ . Каждый этап определяется множеством состояний, в которых система может, находится перед выбором управления, либо в которые может перейти в результате управления, причем в начале первого этапа система находится в начальном состоянии  $\xi_0$ , а в конце последнего – в финальном состоянии  $\xi_T$ . Таким образом, все полные траектории включают  $n+1$  состояние  $(\xi_0, \dots, \xi_n \equiv \xi_T)$ . Можно считать, что функции состояния (3) и эффективность управления изменяются от этапа к этапу, то есть

$$\xi_k = \phi(x_k, \xi_{k-1}), k=1, \dots, n, \quad (6)$$

стоимость шага  $k = 1, \dots, n$  равна  $f_k(x_k, \xi_{k-1})$ , а стоимость стратегии определяется аддитивной функцией  $f(x) = \sum_{k=1}^n f_k(x_k, \xi_{k-1})$ .

Пусть в начале шага  $t$  система находится в состоянии  $\xi$ . Обозначая через  $F_t(\xi)$  минимальное значение  $\sum_{k=1}^n f_k$ , то есть  $F^T(\xi)$ , получаем, что соотношение (5) примет вид

$$F_t(\xi) = \min_{x_t} \{f(x_t, \xi) + F_{t+1}(\phi(\xi_t))\} \quad (7)$$

где минимум берется по всем допустимым на этапе  $t$  управлениям.

Решение задачи (7) – условное оптимальное управление на шаге  $t$  – обозначается  $\hat{x}_t(\xi)$ . Процедура обратной прогонки динамического программирования осуществляется следующим образом. Пусть  $F_{n+1} \equiv 0$ . Используя рекуррентные соотношения (7), последовательно определяем  $F_t(\xi)$  и  $\hat{x}_t(\xi)$  для всех состояний шага  $t = n, \dots, 1$ , причем  $F_n(\xi) = \min\{f_n(x_n, \xi)\}$ . Далее поэтапно восстанавливаются оптимальная траектория и оптимальная стратегия:

$$\xi_0, x^* = \hat{x}_1(\xi_0), \xi_1 = \varphi_1(x^*, \xi_0), \xi_t = \varphi(x_t^*, \xi_{t-1}), x_t^* = \hat{x}_t(\xi_{t-1}), t=2, \dots, n.$$

При использовании процедуры прямой прогонки через  $F_1(\xi)$  обозначается  $F^0(\xi)$ , то есть минимальное значение  $\sum_{k=1}^n f_k$  при условии, что в конце этапа  $t$  система находится в состоянии  $\xi$ . Условное оптимальное управление на этапе  $t$  выбирается в предположении, что предыдущие этапы пройдены оптимально, и рекуррентное соотношение (5) принимает вид

$$F_t(\xi) = \min_{x_t} \{f(x_t, \xi_{t-1}) + F_{t-1}(\xi_{t-1})\}, \xi_t = \varphi(x_t, \xi_{t-1}). \quad (8)$$

Полагая  $F_0 \equiv 0$ , имеем

$$F_t(\xi) = \min_{x_t} \{f_t(x_t, \xi)\} \quad \xi_t = \varphi(x_t, \xi_{t-1}).$$

Последовательно вычисляя  $F_t(\xi)$  и  $\hat{x}_t(\xi)$  для всех возможных в конце этапа  $t$  состояний,  $t = 1, \dots, n$ , на последнем шаге получим  $x_n^* = \hat{x}_n(\xi_0)$ . Пользуясь уравнениями состояния, восстановим в обратном порядке оптимальный план и оптимальную траекторию:

$$\xi_T, x_n^* = \hat{x}_n(\xi_T), \xi_t = \varphi(x_t^*, \xi_{t-1}), x_t^* = \hat{x}_t(\xi_t), t = n-1, \dots, 1.$$

## ВЫВОДЫ

В основе динамического программирования лежит идея решения поставленной задачи путем деления ее на отдельные части (подзадачи, этапы), решение этих подзадач и последующего объединения этих решений в одно общее решение. Динамическое программирование обычно придерживается двух подходов к решению задач:

– нисходящее динамическое программирование – задача разбивается на подзадачи меньшего размера, они решаются и затем комбинируются для решения исходной задачи. Используется запоминание для решений уже решенных подзадач;

– восходящее динамическое программирование – все подзадачи, которые впоследствии понадобятся для решения исходной задачи просчитываются заранее и затем используются для построения решения исходной задачи.

Способ восходящего динамического программирования выглядит эффективней нисходящего программирования в смысле размера необходимого стека и количества вызова функций, но иногда бывает не-

легко заранее выяснить, решение каких подзадач нам потребуется в дальнейшем. Но так как при традиционном подходе управления проектом возможно построение диаграммы Ганта, то данная проблема снимается автоматически.

## ЛИТЕРАТУРА

1. АНДРЕЕВ Г.И., ЗАМАРИН М.Е., СОЗИНОВ П.А., ТИХОМИРОВ В.А. Концептуальные основы управления проектом создания образцов РЭБ. Радиотехника. 2020. Т. 84. №10(20). С. 5–14.
2. АНДРЕЕВ Г.И., СОЗИНОВ П.А., ТИХОМИРОВ В.А. Управленческие решения при проектировании радиотехнических систем. М.: Радиотехника, 2018. 560 с.
3. АНДРЕЕВ Г.И., СОЗИНОВ П.А., ТИХОМИРОВ В.А. Основы теории принятия решений. М.: Радиотехника, 2017.
4. ВОЛОДИН С.В. Общественное проектирование САПР реального времени.
5. ГАСПАРСКИЙ В. Праксиологический анализ проектно-конструкторских разработок. М.: Изд-во «Мир», 1978.
6. ДЖОНС ДЖ.К. Методы проектирования. М.: Мир, 1986.
7. КЛАРК У. Графики Ганта. Управление и планирование работы. М., Л., 1931.
8. КОНЮХОВСКИЙ П.В. Математические методы исследования операций в экономике. СПб.: Питер, 2002.
9. ХОЛЛ А.Д. Опыт методологии для системотехники. М.: «Советское радио», 1975.
10. ASIMOW M. Introduction to Design, Prentice-Hall Inc., Englewood Cliffs, N. J., 1962.
11. GREGORY S.A., DESIGN SCIENCE (в.) The Design Method, London, 1966, Butterworths.

## REFERENCES:

1. ANDREEV G.I., ZAMARIN M.E., SOZINOV P.A., TIKHOMIROV V.A. Conceptual foundations of project management for the creation of electronic warfare samples. *Radiotekhnika*. 2020;84;10(20):5–14. (In Russian).
2. ANDREEV G.I., SOZINOV P.A., TIKHOMIROV V.A. Managerial decisions in the design of radio engineering systems. Monograph. Edited by P.A. Sozinov. Moscow: Radio Engineering. 2018:560. (In Russian).
3. ANDREEV G.I., SOZINOV P.A., TIKHOMIROV V.A. Fundamentals of decision theory. Moscow: Radio Engineering, 2017. (In Russian).
4. VOLODIN S.V. Public design of real-time CAD. (In Russian).
5. GASPARYSKY V. Praxiological analysis of design developments. Moscow: Mir Publishing House, 1978. (In Russian).

6. JONES J.K. Design methods. Moscow: Mir, 1986. (In Russian).
7. CLARK W. Gantt graphs. Management and work planning. Moscow, Leningrad, 1931. (In Russian).
9. HALL A.D. Experience of methodology for system engineering. Moscow: «Soviet radio», 1975. (In Russian).
8. KONYUKHOVSKY P.V. Mathematical methods of operations research in economics. St. Petersburg: Peter, 2002. (In Russian).
10. ASIMOW M. Introduction to Design, Prentice-Hall Inc., Englewood Cliffs, N. J., 1962.
11. GREGORY S.A. Design Science (V.) The Design Method, London, 1966, Butterworths.

---

**Тихомиров Алексей Валерьевич,**

к.т.н., директор дирекции по продажам запасных частей и послепродажного обслуживания АО «Автоваз»

☎ 445024, г. Тольятти, Южное ш., д. 36  
тел.: +7 (964) 557-54-82, e-mail: AV.tiakhomirov@vaz.ru

**Мараховский Александр Васильевич,**

заместитель генерального директора по общим вопросам АО «Центральный научно-исследовательский радиотехнический институт им. академика А.И. Берга»

☎ тел.: +7 916 074-68-81, e-mail: zglov@cnirti.com

**Печалин Николай Дмитриевич,**

аспирант, начальник корпоративного отдела АО «Центральный научно-исследовательский радиотехнический институт им. академика А.И. Берга»

☎ 107078, г. Москва, ул. Новая Басманная, д. 20, стр. 9  
тел.: +7 (917) 540-28-71, e-mail: corp@cnirti.ru