

**МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ
РОССИЙСКОЙ ФЕДЕРАЦИИ**

Федеральное государственное автономное образовательное учреждение
высшего образования
«Национальный исследовательский
Нижегородский государственный университет им. Н.И. Лобачевского»

С.Ю. Городецкий

**МЕТОДЫ ОПТИМИЗАЦИИ НА ГРАФАХ
С ВЕКТОРНЫМИ ВЕСАМИ РЕБЕР**

Учебно-методическое пособие

Рекомендовано методической комиссией института ИТММ
для студентов ННГУ, обучающихся по направлениям подготовки
01.03.02 «Прикладная математика и информатика»,
02.03.02 «Фундаментальная информатика и информационные технологии»

2-е издание, переработанное и дополненное

Нижний Новгород
2024

УДК: 519.17–37

ББК В18

Г 70

Г 70 Городецкий С.Ю. МЕТОДЫ ОПТИМИЗАЦИИ НА ГРАФАХ С ВЕКТОРНЫМИ ВЕСАМИ РЕБЕР: Учебно-методическое пособие. – 2-е изд., перераб. и доп. – Нижний Новгород: Нижегородский госуниверситет, 2024. – 38 с.

Рецензент: кандидат физ.-мат. наук, доцент **Д. Е. Шапошников**

Пособие посвящено важному подклассу задач динамического программирования – задачам поиска оптимальных путей на графах. Для скалярных постановок рассмотрены два метода – рекуррентных уравнений Беллмана (применительно к задачам на графах) и метод Дейкстры, а также их связь с принципом Беллмана в форме достаточного условия. Далее задачи обобщаются на случай векторных весов ребер, вводятся понятия оптимальности по Парето и Слейтеру. Изучаются свойства скаляризованных задач, порождаемых двумя типами сверток – линейной и Гермейера, а также использование сверток при решении многокритериальных задач в сочетании с методами рекуррентных уравнений и Дейкстры. Описан метод построения множества эффективных путей, не использующий свертки. Для численных экспериментов представлена учебно – исследовательская программная лаборатория «Поиск оптимальных путей на графах с векторными весами ребер».

Учебно-методическое пособие предназначено для студентов ННГУ, обучающихся по направлениям подготовки 01.03.02 «Прикладная математика и информатика» и 02.03.02 «Фундаментальная информатика и информационные технологии»

УДК 519.17–37

ББК В18

С.Ю. Городецкий

© Нижегородский государственный
университет им. Н.И. Лобачевского, 2024

ОГЛАВЛЕНИЕ

ВВЕДЕНИЕ	4
1. ЗАДАЧИ ПОИСКА ОПТИМАЛЬНЫХ ПУТЕЙ НА ОРИЕНТИРОВАННЫХ ГРАФАХ СО СКАЛЯРНЫМИ ВЕСАМИ	5
2. ПРИНЦИП БЕЛЛМАНА В ЗАДАЧАХ ПОИСКА ОПТИМАЛЬНЫХ ПУТЕЙ НА ГРАФАХ	7
3. ИСПОЛЬЗОВАНИЕ РЕКУРРЕНТНЫХ УРАВНЕНИЙ БЕЛЛМАНА ДЛЯ ПОИСКА ОПТИМАЛЬНЫХ ПУТЕЙ НА ГРАФАХ	10
4. МЕТОД ДЕЙКСТРЫ ДЛЯ ПОИСКА ОПТИМАЛЬНЫХ ПУТЕЙ НА ГРАФАХ С НЕОТРИЦАТЕЛЬНЫМИ ВЕСАМИ	13
5. ОБОБЩЕНИЕ ЗАДАЧИ ПОИСКА ОПТИМАЛЬНЫХ ПУТЕЙ ДЛЯ ВЕКТОРНЫХ ВЕСОВ РЕБЕР	18
6. ИСПОЛЬЗОВАНИЕ МЕТОДА СВЕРТОК ДЛЯ ОТЫСКАНИЯ ПОЛУЭФФЕКТИВНЫХ И ЭФФЕКТИВНЫХ РЕШЕНИЙ	21
7. ПРИМЕНЕНИЕ МЕТОДОВ ДЕЙКСТРЫ И БЕЛЛМАНА В СОЧЕТАНИИ С МЕТОДОМ СВЕРТОК	25
8. МОДИФИКАЦИЯ МЕТОДА ДЕЙКСТРЫ ДЛЯ ПОСТРОЕНИЯ ВСЕГО МНОЖЕСТВА ЭФФЕКТИВНЫХ ПУТЕЙ И МНОЖЕСТВА ПАРЕТО	27
9. ОБЩИЕ СВЕДЕНИЯ О КОМПЬЮТЕРНОЙ ЛАБОРАТОРИИ	32
10. КОНТРОЛЬНЫЕ ВОПРОСЫ И ЗАДАНИЯ	35
ЛИТЕРАТУРА	37

ВВЕДЕНИЕ

В курсе «Методы оптимизации» Вы уже познакомились с интересным и полезным классом задач – задачами динамического программирования с дискретным временем [1]. Многие прикладные задачи из области организации производства, перевозок, финансовой деятельности могут быть математически поставлены как задачи динамического программирования [2, 3]. Вам знаком также общий подход к их решению – метод рекуррентных уравнений, предложенный в 60-х годах прошлого века Ричардом Беллманом [2, 4].

Использование этого подхода наиболее просто и естественно в задачах с фиксированным временем протекания управляемого динамического процесса, когда вне зависимости от выбора управлений для достижения требуемых конечных состояний всегда необходимо выполнить одно и то же число шагов. Однако далеко не все задачи обладают этим свойством.

Важным частым случаем задач динамического программирования, для которого данное свойство в прикладных ситуациях часто нарушается, являются задачи построения оптимальных путей на графах. (Заметьте, что в такой форме может быть представлена любая задача динамического программирования с конечным числом состояний и управлений). Метод рекуррентных уравнений Беллмана обобщается и на такие задачи. Кроме того, для задач на графах разработаны специальные методы, которые, как правило, более удобны в использовании по сравнению с общим методом рекуррентных уравнений Беллмана. Один из них – метод Дейкстры, рассмотрен в данной методической разработке. Кроме скалярных задач Вы познакомитесь с их обобщением – задачами многокритериальной оптимизации на графах с векторными весами ребер, а также методами их решения, основанными как на использовании так называемых сверток, так и с безсверточными методами.

1. ЗАДАЧИ ПОИСКА ОПТИМАЛЬНЫХ ПУТЕЙ НА ОРИЕНТИРОВАННЫХ ГРАФАХ СО СКАЛЯРНЫМИ ВЕСАМИ

Пусть X – конечное множество состояний x автономной управляемой динамической системы, u – управление, принимающие в каждом из состояний значения из конечного множества $U(x)$, а $\hat{x} = f(x, u)$ – уравнение данной динамической системы, определяющее следующее состояние \hat{x} , в которое перейдёт система из текущего состояния x при выборе управления u . На переходах $x \xrightarrow{u} \hat{x}$ определяется функция затрат $C(x, u, \hat{x})$. Кроме того, задаются возможные начальные и конечные состояния динамического процесса, образующие множества X^S и X^F . Качество пути обычно оценивается суммой затрат при переходе из начального в конечное состояние по этому пути.

Задачи такого типа могут быть легко представлены в форме задач оптимизации на графах. Для этого достаточно интерпретировать каждое из состояний X динамической системы как вершину некоторого графа, а переходы $\hat{x} = f(x, u)$, как его ориентированные рёбра $g = (x, u, \hat{x})$, снабжённые весами $C(g) = C(x, u, \hat{x})$.

В результате получим ориентированный граф с нагруженными рёбрами $\langle X, G, C: G \rightarrow R^1 \rangle$, где X – множество вершин, G – множество рёбер.

Пусть $\gamma(x', x'')$ – путь из вершины x' в x'' , проходящий по рёбрам графа. (В некоторых случаях будем использовать в качестве аргументов в γ вместо конкретных вершин x' или x'' множества вершин, тогда, например, под $\gamma(X^S, x)$ будет пониматься путь, соединяющий некоторую вершину x^S из X^S с вершиной x).

Критерий качества пути можно определить различным образом. Будем рассматривать два вида критериев, аддитивный и критерий типа максимума:

$$W(\gamma(x', x'')) = W^1(\gamma(x', x'')) = \sum_{g \in \gamma(x', x'')} C(g), \quad (1)$$

или

$$W(\gamma(x', x'')) = W^2(\gamma(x', x'')) = \max_{g \in \gamma(x', x'')} C(g). \quad (2)$$

Оптимальным называется путь γ^* , соединяющий вершину из X^S с вершиной из X^F и обеспечивающий минимум используемого критерия W среди всех таких путей

$$\gamma^* = \arg \min_{\gamma \in \{ \gamma(X^S, X^F) \}} W(\gamma) \quad (3)$$

В качестве прикладных задач, в которых возникают критерии вида (1), (2) можно указать, например, задачи выбора маршрута на дорожной сети исходя из его протяженности, затрат горючего (критерий (1)) или из его радиационной безопасности, проходимости (критерий (2)). На рис.1 приведен пример ориентированного графа с нагруженными рёбрами. Значения весов ребер надписаны над ребрами. Если считать, что веса $C(g)$ отражают расход горючего и использовать критерий (1), то оптимальным (по общему расходу горючего) окажется путь с весами 1–5, отмеченный жирной линией; если же считать, что веса $C(g)$ отражают степень непроходимости участков и используется критерий (2), то оптимальным (по проходимости наихудшего участка пути) окажется путь с весами 3–4, отмеченный пунктиром, а также путь с весами 1–2–4.

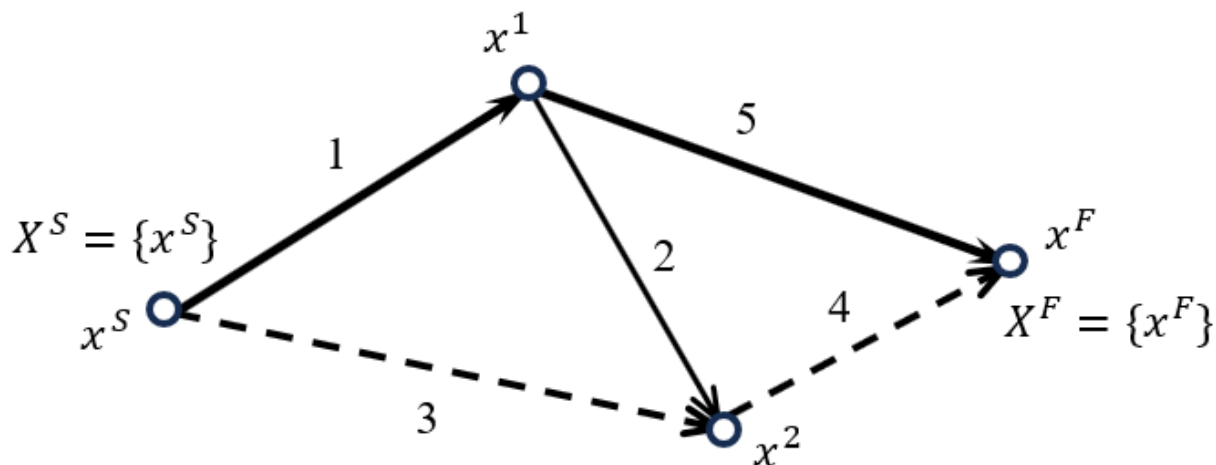


Рис. 1. – Пример графа со скалярными весами ребер

2. ПРИНЦИП БЕЛЛМАНА В ЗАДАЧАХ ПОИСКА ОПТИМАЛЬНЫХ ПУТЕЙ НА ГРАФАХ

Все методы решения рассматриваемых задач (3), (2) или (3), (1) в явной или неявной форме опираются на *принцип Беллмана*. Его можно сформулировать в двух формах: как необходимое и как достаточное условие оптимальности пути (3). Методы решения, естественно, используют принцип Беллмана в форме достаточного условия.

Рассмотрим вначале форму *необходимого условия*. В этой форме принцип звучит так. Если путь γ^* является оптимальным, то любой его начальный участок (от x^S до промежуточного состояния x этого пути), а также любой его конечный участок (от промежуточного состояния x до x^F) являются оптимальными на множестве путей, соединяющих эти состояния, т. е. для x^S , x^F из γ^* и для любой вершины x из γ^*

$$W(\gamma^*(x^S, x)) = \min_{\gamma \in \{\gamma(x^S, x)\}} W(\gamma); W(\gamma^*(x, x^F)) = \min_{\gamma \in \{\gamma(x, x^F)\}} W(\gamma) \quad (4)$$

Справедливость этого утверждения легко доказать методом от противного для аддитивного критерия (1) (проведите это доказательство!). Однако для критерия (2) доказательство провести нельзя, поскольку принцип Беллмана как необходимое условие для этого критерия выполняется не для всякого оптимального пути γ^* . В этом можно убедиться с помощью примера, представленного на рис.1. Участок $x^S - x^2$ оптимального по критерию (2) пути, отмеченного пунктиром, не является оптимальным между этими точками (для него $W = 3$). Оптимальным же будет путь $x^S - x^1 - x^2$ (для него $W = 2$).

Поскольку методы поиска оптимальных путей используют принцип Беллмана в форме достаточного условия, а не необходимого, то нарушение последнего не является препятствием для построения эффективных методов решения задачи (3), (2).

Как *достаточное условие*, принцип можно сформулировать либо по отношению к концу, либо по отношению к началу процесса. *По отношению к концу*: если в каждой вершине графа, в которую попал управляемый процесс (или в начальном множестве вершин) выбирать ребро, являющееся началом оптимального пути из этой вершины в конечное множество X^F , то путь,

составленный из таких ребер, будет оптимальным. По отношению к началу: если в каждую вершину (или множество конечных вершин) входить по ребру, являющемуся последним в оптимальном пути, соединяющем множество X^S с этой вершиной (вершинами), то построенный по таким ребрам путь из X^S в X^F будет оптимальным.

Докажем последнее утверждение для критерия (2) методом математической индукции. Прежде всего, введем понятие *пути* $\gamma^0(x, X^F)$, *оптимального* на участке от текущей вершины x до X^F с учетом *предыстории*. Будем говорить, что $\gamma^0(x, X^F)$ является путем указанного типа, если к нему можно подобрать начальный участок пути $\gamma^*(X^S, x)$ такой, что объединение путей $\gamma^*(X^S, x)$ и $\gamma^0(x, X^F)$ дает оптимальный путь в задаче (2), (3). Значение критерия качества на оптимальном пути обозначим через W^* .

Пусть последнее ребро пути $\gamma(X^S, X^F)$ (по нему процесс приходит в конечную вершину из множества X^F) выбирается, согласно принципу Беллмана, как последнее ребро оптимального пути в задаче (2), (3). Очевидно, что на последнем шаге это оптимально с учетом предыстории.

Предположим, что на последних L шагах (для $L \leq k$) применение способа построения пути, указанного в принципе Беллмана, дает путь $\gamma^0(x, X^F)$, оптимальный с учетом предыстории. Докажем, что это свойство сохранится и для $(k + 1)$ -го шагов. Фактически, нужно доказать, что если для вершины y применение принципа Беллмана породило путь $\gamma(y, X^F)$, состоящий из $(k + 1)$ -го ребра, то этот путь будет оптимальным с учетом предыстории. Убедимся в этом.

Обозначим через g первое ребро пути $\gamma = \gamma(y, X^F)$, через x – конечную вершину этого ребра, а остаток пути γ , после удаления ребра g , – через $\hat{\gamma}(x, X^F)$. Из способа построения пути γ и предположения математической индукции следует, что существует оптимальный на участке от X^S до x путь $\gamma^*(X^S, x)$, оканчивающийся ребром g . Очевидно, что

$$W(\gamma^*(X^S, x)) \leq W^*.$$

Поскольку путь $\hat{\gamma}(x, X^F)$ содержит k ребер, и они выбраны согласно способу, указанному в принципе Беллмана, то, по предположению индукции,

путь $\hat{\gamma}(x, X^F)$ оптимален с учетом предыстории. Следовательно, поскольку мы рассматриваем критерий вида (2),

$$W(\hat{\gamma}(x, X^F)) \leq W^*.$$

Для объединенного пути, составленного из участка $\gamma^*(X^S, x)$ и участка $\hat{\gamma}(x, X^F)$, выполняется

$$W(\gamma^*(X^S, x) \cup \hat{\gamma}(x, X^F)) = \max\{W(\gamma^*(X^S, x)); W(\hat{\gamma}(x, X^F))\} \leq W^* \quad (5)$$

но поскольку всегда $W(\gamma(X^S, X^F)) \geq W^*$, то в действительности в (5) имеет место равенство с W^* .

Рассмотрим теперь путь $\hat{\gamma}^*(X^S, x)$, получаемый из $\gamma^*(X^S, x)$ выбрасыванием ребра g (напомним, что оно соединяет вершину y с вершиной x). Очевидно, что

$$W(\hat{\gamma}^*(X^S, y) \cup \gamma(y, X^F)) = W(\gamma^*(X^S, x) \cup \hat{\gamma}(x, X^F)),$$

и, следовательно,

$$W(\hat{\gamma}^*(X^S, y) \cup \gamma(y, X^F)) = W^*.$$

Последнее равенство доказывает, что любой путь $\gamma(y, X^F)$, состоящий из $(k + 1)$ -го ребра и построенный по способу, указанному в принципе Беллмана, является оптимальным с учетом предыстории.

Отсюда, по индукции, следует справедливость принципа Беллмана (как достаточного условия) для критерия (2).

3. ИСПОЛЬЗОВАНИЕ РЕКУРРЕНТНЫХ УРАВНЕНИЙ БЕЛЛМАНА ДЛЯ ПОИСКА ОПТИМАЛЬНЫХ ПУТЕЙ НА ГРАФАХ

Для решения задачи поиска оптимальных путей на графах можно использовать известный Вам общий метод рекуррентных соотношении Беллмана, приспособив его для случая нефиксированного числа шагов [5]. Нужно сразу отметить, что получаемая процедура не отличается особым удобством и выглядит несколько громоздкой. Существуют другие вычислительные процедуры, специально предназначенные для решения задач оптимизации на графах. Одна из них, предложенная Дейкстрой [6], будет рассмотрена в следующем разделе.

Здесь же (для сравнения) опишем *модифицированный метод уравнений Беллмана*, приспособленный к задачам с *неизвестным временем достижения множества финальных состояний*. Запишем его в такой форме, чтобы вычисления начинались от стартовых вершин X^S графа.

Определим рекурсивно вспомогательные множества X_k . Примем $X_0 = X^S$, а через X_1 обозначим множество вершин, в которые можно попасть из X_0 за один шаг (переходом по одному ребру). Если уже определено множество X_k , то X_{k+1} – множество вершин, в которые можно попасть из X_k за один шаг. Заметим, что в силу введенного определения множества X_i , и X_j при $i \neq j$ могут пересекаться, так как возможно, что в одну и ту же вершину графа можно попасть из X^S за разное число шагов.

Свяжем с каждой вершиной x графа метку $S(x)$. Её значение будет определять минимальные затраты по переходу из X^S в x не более чем за некоторое число шагов k .

В процессе расчета k будет постепенно увеличиваться, принимая значения 0, 1, 2, При этом метка вершины $S(x)$ может уменьшаться. После того как $S(x)$ перестанет изменяться, она будет определять оптимальные затраты по переходу из X^S в x с учетом всех возможных путей в графе.

Перед началом вычислительного процесса для всех вершин x , не входящих в X^S , принимается $S(x) = \infty$, а для x из X^S принимается $S(x) = 0$ и полагается $k = 0$.

Далее рассматривается множество вершин X_{k+1} , в которые можно попасть за один шаг из вершин, входящих в X_k (напомним, что X_{k+1} и X_k могут пересекаться).

Если X_{k+1} не пусто, то для всех x из X_{k+1} происходит корректировка меток по следующей схеме:

$$S(x) := \min \left\{ S(x); \min_{y \in X_k, u \in U(y), f(y,u)=x} \{S(y) \oplus C(y, u, x)\} \right\}, \quad (6)$$

Вместо символа \oplus для задачи с аддитивным критерием (1) должна использоваться операция суммирования, а для критерия (2) – операция взятия максимума из двух аргументов. Если применение правила (6) приводит к уменьшению метки вершины x , происходит запоминание ребра $g^*(x) = (y, u, x)$, входящего в вершину x , обеспечившего это изменение.

После обработки всех вершин из X_{k+1} полагается $k := k + 1$ и процесс повторяется. Необходимо отметить, что значок присваивания «:=», использованный в (6), следует понимать таким образом, что до завершения обработки всех вершин из X_{k+1} в правой части выражения всюду используются «старые» значения $S(x)$ и лишь потом происходит их обновление. Приведенные ниже рис. 2–3 иллюстрируют процесс решения с использованием соотношений (6) для задачи с аддитивным критерием.

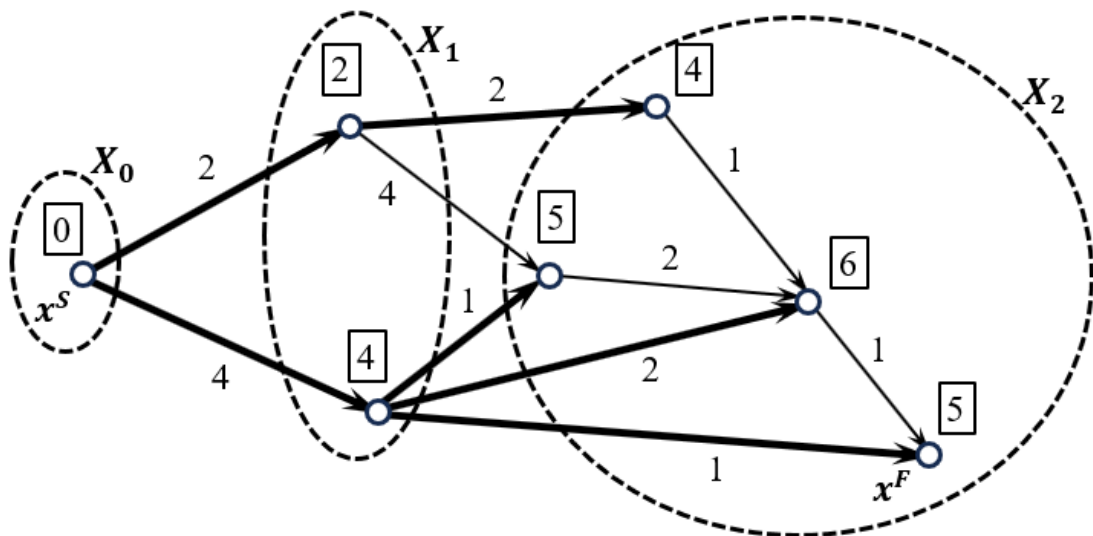


Рис. 2. – Результат итераций для шагов $k = 0$, $k = 1$, и $k = 2$ метода Беллмана

На рисунках помеченные входящие ребра вершин показаны жирными стрелками. Поэтому текущими оценками оптимальных путей к промежуточным вершинам являются выделенные жирностью ветви графа. Прямоугольниками обведены текущие пометки вершин.

Определим критерий останова. Если при попытке перехода к очередному множеству X_{k+1} оно оказывается пусто, процесс корректировки меток и установки пометок ребер заканчивается. После этого по запомненным помеченным ребрам $g^*(x)$ строится оптимальный путь, начиная с последней вершины x^F , определяемой следующим образом. В качестве x^F выбирается та из вершин множества X^F , для которой значение метки $S(x)$ является наименьшим:

$$x^F = \arg \min_{x \in X^F} S(x).$$

Далее по $g^*(x^F)$ определяется вершина x , предшествующая x^F в оптимальном пути, затем – вершина, предшествующая x , и т.д. Например, если $g^*(x^F) = (y, u, x^F)$, то y определяет вершину, предшествующую x^F .

Применение этого правила в примере, показанном на рис. 3, приводит к выделению в качестве оптимального – пути, проведенного по нижним ребрам графа (с весами 4 и 1).

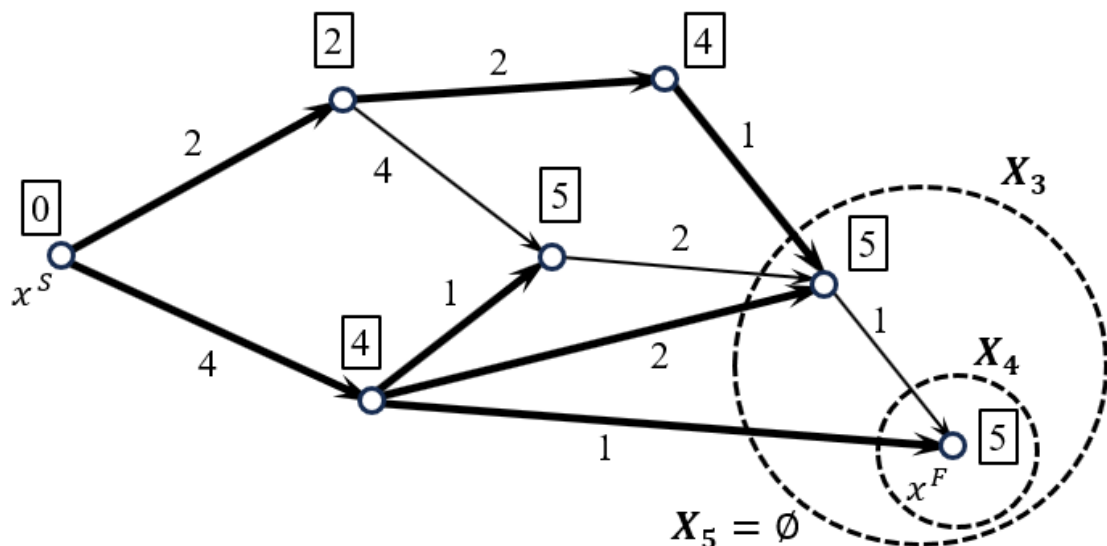


Рис. 3. – Результат завершающих итераций $k = 3$ и $k = 4$ метода Беллмана с последующим построением оптимального пути

4. МЕТОД ДЕЙКСТРЫ ДЛЯ ПОИСКА ОПТИМАЛЬНЫХ ПУТЕЙ НА ГРАФАХ С НЕОТРИЦАТЕЛЬНЫМИ ВЕСАМИ

Метод Дейкстры [6] специально разработан для поиска оптимальных путей на графах. Ограничением на класс решаемых задач является *требование неотрицательности весов ребер*. При дальнейшем изложении будем рассматривать только те задачи, в которых конечное и начальное множества состоят из одной вершины.

Собственно, это ограничение не является существенным, поскольку к данному, казалось бы, частному классу задач легко сводятся задачи более общего вида. Действительно, если, например, начальная вершина не единственна, можно ввести новую фиктивную вершину, предшествующую множеству исходных начальных вершин, связав ее направленными ребрами нулевого веса с вершинами упомянутого множества.

Метод Дейкстры строит оптимальный путь, опираясь на правило, сформулированное в принципе Беллмана (в форме достаточного условия).

Приведем общее описание метода. С каждой вершиной x графа связывается *временная метка* $S(x)$. Вначале метки определяют *верхние оценки критерия качества оптимального пути* из x^S в x . Затем происходит уменьшение значений меток, но не более чем до оптимального значения. Одновременно *помечаются* те из входящих в рассматриваемые вершины ребер, которыми оканчиваются пути из x^S , породившие установленные значения временных меток вершин. На каждом шаге *одна из временных меток (наименьшая) становится постоянной*. Она определяет значение критерия качества оптимального пути до этой вершины. Последовательность помеченных ребер выделяет оптимальный путь. Процесс организован так, что для каждой новой вершины с постоянной меткой выделенный для неё путь обязательно проходит только по вершинам, уже имеющим постоянные метки, и только по помеченным ребрам. Именно поэтому для получаемых в методе путей выполняется правило из принципа Беллмана, что обеспечивает оптимальность найденных путей.

АЛГОРИТМ МЕТОДА ДЕЙКСТРЫ

ШАГ 0 – *начальные операции*. Вершине x^S приписывается нулевая метка, которая считается постоянной, всем остальным вершинам – бесконечные, которые считаются временными; в качестве вершины v начала очередной итерации выбирается вершина x^S

ШАГ 1 – *обновление меток*. У всех вершин, достижимых из v , обновляются метки согласно правилу:

– если критерий аддитивный вида (1), то

$$S(x) := \min \{ S(x); S(v) + C(v, u, x) \mid x=f(v, u) \};$$

– если критерий типа максимума вида (2), то

$$S(x) := \min \{ S(x); \max \{ S(v); C(v, u, x) \mid x=f(v, u) \} \} \quad (7)$$

Запоминается входящее в вершину x ребро $g^*(x)$, соответствующее значению минимума: если минимум в (7) достигается на первом элементе, то $g^*(x)$ не изменяется, в противном случае полагается $g^*(x) = (v, u, x)$.

ШАГ 2 – *получение постоянной метки*. Из всех вершин с временными метками выбирается вершина, имеющая наименьшую временную метку. Эта метка становится постоянной, а соответствующая вершина объявляется вершиной v – началом следующей итерации.

ШАГ 3 – *критерий останова*. Если $v = x^F$, то процесс вычислений останавливается и выполняется переход на шаг 4, иначе – выполняется возврат на шаг 1.

Шаг 4 – *построение пути*. После останова значение метки $S(x^F)$ определяет значение критерия качества оптимального пути, а сам оптимальный путь проводится по отмеченным ребрам (через вершины с постоянными метками).

Назовем путь, построенный по указанным на шаге 4 правилам, *путем, построенным по правилам Дейкстры*, и обозначим его $\gamma^D(X^S, X^F)$.

Для обоснования метода Дейкстры достаточно доказать, что путь, проходящий по отмеченным ребрам в вершину с постоянной меткой (назовем ее вершиной x), есть оптимальный путь для этой вершины. (Заметим, что путь,

построенный методом Дейкстры обязательно проходит только по вершинам с постоянными метками).

Для доказательства рассмотрим другие варианты выбора пути в x . При этом, либо на таких путях все вершины окажутся с постоянными метками, но будут присутствовать непомеченные ребра, либо встретятся вершины с временными метками.

Проанализируем первый из этих двух случаев, предполагая, что построенный путь $\gamma(x^S, x)$ может включать несколько непомеченных ребер. Один из возможных вариантов показан на рис. 4, помеченные ребра выделены жирно.

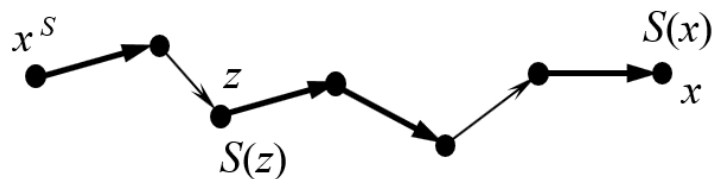


Рис. 4. – Путь $\gamma(x^S, x)$, включающий вершины с постоянными метками (помечены черными кружками) и содержащий несколько непомеченных ребер (показаны тонкими стрелками)

Пусть первое непомеченное ребро приводит в некоторую вершину, которую обозначим символом z . Поскольку все вершины участка пути $\gamma(x^S, z)$ имеют постоянные метки, то включение непомеченного ребра (приводящего в нашем случае в вершину с постоянной меткой) не может уменьшить значение критерия качества пути, поскольку при определении метки вершины z вариант прихода в z по этому непомеченному ребру уже рассматривался и не привел к значению критерия качества перехода меньшему, чем $S(z)$. Само же значение постоянной метки $S(z)$ соответствует критерию качества пути $\gamma^D(x^S, z)$, связывающего x^S с z и построенного по правилу Дейкстры (т.е. использующего только помеченные ребра и вершины с постоянными метками – это показано на рис. 5).

Поскольку

$$W(\gamma^D(x^S, z)) = S(z) \leq W(\gamma(x^S, z)),$$

то в пути $\gamma(x^S, x)$ его участок $\gamma(x^S, z)$ можно заменить на обходной путь $\gamma^D(x^S, z)$, не ухудшая качества пути.

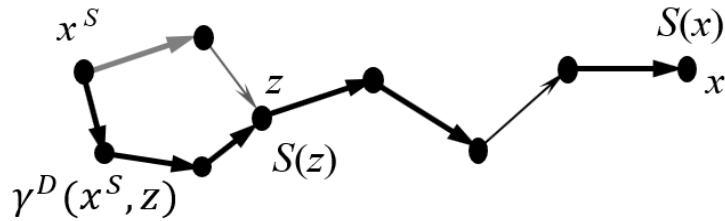


Рис. 5. – Обход участка с первым непомеченным ребром с помощью фрагмента пути, построенного из x^S в z по правилам Дейкстры (помеченные ребра выделены жирным)

В полученном новом пути от x^S до x (назовем его $\hat{\gamma}(x^S, x)$) одно непомеченное ребро исчезло. К оставшимся непомеченным нужно последовательно применить ту же процедуру. В итоге исходный путь $\gamma(x^S, x)$ будет полностью заменен на путь $\gamma^D(x^S, z)$, соответствующий правилам Дейкстры, причем окажется, что

$$W(\gamma^D(x^S, x)) \leq W(\gamma(x^S, x)).$$

Итак, вершины с постоянными метками всегда следует соединять только по помеченным ребрам.

Осталось рассмотреть второй случай, когда путь $\gamma(x^S, x)$ содержит вершины с временными метками (см. рис. 6). Заметим, что по только что доказанному, соседние вершины с постоянными метками следует соединять только помеченными ребрами.

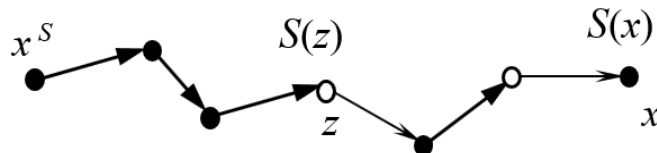


Рис. 6. – Путь $\gamma(x^S, x)$ с включением вершин с временными метками (помечены белыми кружками). Из вершин с постоянными метками (помечены черными кружками) исходят только помеченные ребра

Пусть z – первая вершина с временной меткой в рассматриваемом пути $\gamma(x^S, x)$, а $\gamma(x^S, z)$ часть этого пути до вершины z . Заметим, что ребро, следующее за временной меткой, не может (согласно правилам метода Дейкстры) быть помеченным, а предшествующее этой вершине ребро может быть как помеченным, так и не помеченным).

Поскольку z – первая вершина с временной меткой, то при вычислении $S(z)$ вариант движения по $\gamma(x^S, z)$ рассматривался и, следовательно, $S(z) \leq W(\gamma(x^S, z))$, а в силу неотрицательности весов ребер, выполняется неравенство $W(\gamma(x^S, z)) \leq W(\gamma(x^S, x))$.

С другой стороны, значение критерия качества пути $\gamma^D(x^S, x)$, построенного по правилу Дейкстры, равно постоянной метке $S(x)$, которая меньше или равна временной метки $S(z)$ (напомним, что по правилам построения постоянных меток все они – меньше или равны временных). Таким образом

$$W(\gamma^D(x^S, x)) = S(x) \leq S(z) \leq W(\gamma(x^S, z)) \leq W(\gamma(x^S, x))$$

Отсюда следует, что путь $\gamma(x^S, x)$ не улучшает значение критерия качества пути по сравнению с путем, построенным методом Дейкстры. Оптимальность получаемых по правилам Дейкстры решений доказана.

В целях самоконтроля ответьте на пару следующих вопросов.

1. Объясните, почему в приведенном выше доказательстве, касающемся второго случая нарушения правила Дейкстры, вначале нужно рассматривать именно первое непомеченное ребро?

2. Почему в методе Дейкстры требуется неотрицательность весов ребер? Где это используется при доказательстве оптимальности построенного по правилам метода Дейкстры пути?

5. ОБОБЩЕНИЕ ЗАДАЧИ ПОИСКА ОПТИМАЛЬНЫХ ПУТЕЙ ДЛЯ ВЕКТОРНЫХ ВЕСОВ РЕБЕР

Типичной для прикладных задач является ситуация, когда с каждым ребром $g = (x, u, y)$ связывается не одна числовая характеристика $C(g)$, а несколько: $C_i(g)$ ($i = 1, \dots, n$), образующих векторный вес ребра

$$C(g) = (C_1(g), \dots, C_n(g)).$$

В соответствии с этим, критерии качества пути (1), (2) становятся векторными: $W(\gamma) = (W_1(\gamma), \dots, W_n(\gamma))$ с компонентами $W_i(\gamma)$, где-либо для всех i от 1 до n

$$W_i(\gamma) = \sum_{g \in \gamma} C_i(g), \quad (8)$$

либо

$$W_i(\gamma) = \max_{g \in \gamma} C_i(g) \quad (9)$$

В общем случае компоненты векторного критерия являются противоречивыми в том смысле, что их минимальные значения достигаются на различных путях γ^* .

Возможная противоречивость задачи по совокупной минимизации компонент векторного критерия приводит к необходимости иного, по сравнению с (3), подхода к определению понятия решения при оптимизации с векторным критерием [1, 11] (т.е. в многокритериальной оптимизации)

$$W(\gamma) \Rightarrow \min_{\gamma \in \{\gamma(X^S, X^F)\}} \quad (10)$$

К множеству решений, очевидно, должны быть отнесены те пути γ , соединяющие X^S с X^F , которые обеспечивают некоторый компромисс с точки зрения задачи минимизации всех компонент векторного критерия.

Компромисс можно понимать различным образом. Общепринятыми являются понятия полуэффективных (слабо эффективных) решений γ^0 и эффективных решений γ^* .

Допустимый путь γ^0 называют *полуэффективным (слабо эффективным)*, если за счет выбора другого допустимого пути γ нельзя получить значение векторного критерия $W(\gamma)$, меньше $W(\gamma^0)$ сразу по всем своим компонентам. То же определение в виде формальной математической записи имеет вид:

$\gamma^o \in \{\gamma(X^S, X^F)\}$ и $\nexists \gamma \in \{\gamma(X^S, X^F)\}$, что $W(\gamma) < W(\gamma^o)$.

Пути γ^o называют также *оптимальными по Слейтеру*, а множество значений критериев для таких путей $S = \{W(\gamma): \gamma \in \{\gamma^o\}\}$ – *множеством Слейтера*.

Аналогично вводится понятие *множества Парето* $P = \{W(\gamma): \gamma \in \{\gamma^*\}\}$ и множества *эффективных (оптимальных по Парето) путей* γ^* . Различие понятий состоит в том, что требования, накладываемые на эффективный путь γ^* – более жесткие, чем на слабоэффективный. А именно, требуется, чтобы для допустимого γ^* не нашлось никакого другого допустимого пути γ , не худшего по всем компонентам W_i пути γ^* и лучшего хотя бы для одного значения i . То же определение эффективного пути γ^* , представленное в виде формальной математической записи имеет вид:

$\gamma^* \in \{\gamma(X^S, X^F)\}$ и $\nexists \gamma \in \{\gamma(X^S, X^F)\}$, что $W(\gamma) \leq W(\gamma^*)$ и $W(\gamma) \neq W(\gamma^*)$.

На рис. 7 для случая $n = 2$ проиллюстрирован вид множеств Парето и Слейтера. Заметьте, что множество Парето является частью множества Слейтера, т.е. $P \subseteq S$.

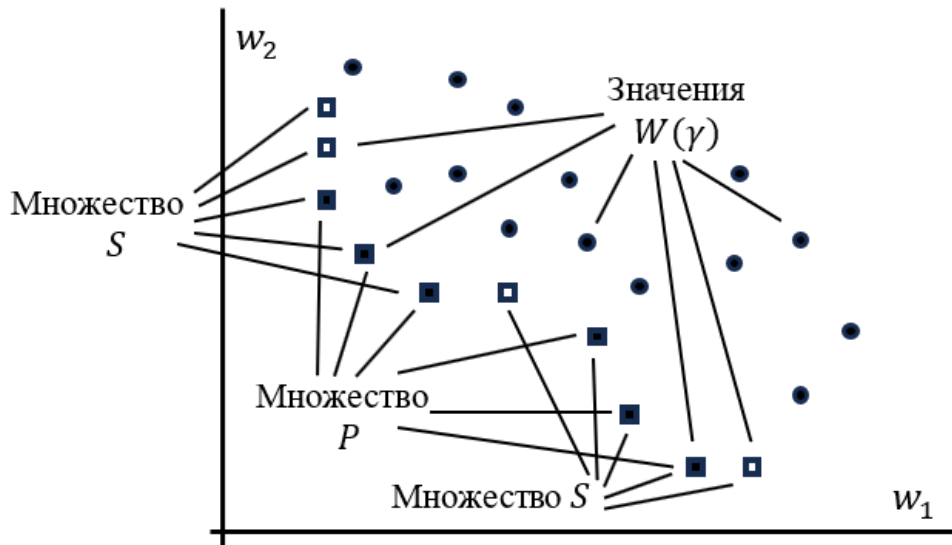


Рис. 7. – Вид множеств Парето и Слейтера в пространстве значений критериев. Круглыми и квадратными маркерами показано множество векторных стоимостей всех путей в графе, связывающих вершины из множества X^S с X^F

Эффективные (оптимальные по Парето) решения таковы, что при переходе от одного эффективного решения к другому некоторые компоненты векторного критерия строго убывают, но зато некоторые другие при этом строго возрастают.

Для поиска эффективных (полуэффективных) решений многокритериальных задач существуют методы двух типов. Методы первого типа позволяют сразу находить все множество эффективных решений, обладая при этом сравнительно высокой вычислительной сложностью (описание метода такого типа приведено в разделе 8). Методы второго типа требуют меньших вычислительных затрат, но позволяют находить каждый раз только отдельные решения. К этой последней группе методов относятся *методы сверток*.

Контрольное задание. Подумайте, какой геометрический вид (по отношению к векторам $W(\gamma^0)$ и $W(\gamma^*)$) имеют множества, в которых не могут располагаться векторные стоимости $W(\gamma)$ допустимых путей, если γ^0 – полуэффективный, а γ^* – эффективный пути.

6. ИСПОЛЬЗОВАНИЕ МЕТОДА СВЕРТОК ДЛЯ ОТЫСКАНИЯ ПОЛУЭФФЕКТИВНЫХ И ЭФФЕКТИВНЫХ РЕШЕНИЙ

Идея *метода сверток* состоит в том, чтобы свести задачу (10) с векторным критерием качества $W(\gamma)$ к задаче со скалярным критерием [11] $\Psi_\lambda(\gamma)$, являющимся сверткой векторного: $\Psi_\lambda(\gamma) = \tilde{\Psi}_\lambda(W(\gamma))$. В действительности возникает даже не одна скалярная задача, а семейство задач, зависящих от параметров $\lambda \in \Lambda$ (коэффициентов свертки):

$$\Psi_\lambda(\gamma) \Rightarrow \min_{\gamma \in \{\gamma(X^S, X^F)\}}, \quad (11)$$

где Λ – множество допустимых значений для параметров свертки.

Конечно же, функция свертки $\tilde{\Psi}_\lambda$ не может быть выбрана произвольно. Зададимся вопросом, как ее выбрать и как понимать термин «сведение задачи»? Очевидно, в методе сверток функция $\tilde{\Psi}_\lambda(w)$ должна строиться так, чтобы решения задач (12) метода сверток (обозначим эти решения γ_λ^*) нужным образом были связаны с искомыми множествами полуэффективных $\{\gamma^o\}$ и (или) эффективных $\{\gamma^*\}$ решений многокритериальной задачи (10).

Сформулируем идеальные требования к такой взаимосвязи. Для некоторых типов сверток они могут выполняться полностью, для других – лишь частично [11]. Выделим две группы требований.

Требования типа I.

Ia. Можно указать подмножество $\Lambda^o \subseteq \Lambda$, что $\forall \lambda \in \Lambda^o$ решение γ_λ^* задачи со сверткой будет полуэффективным (слабо эффективным), т.е. $\gamma_\lambda^* \in \{\gamma^o\}$.

Ib. Можно указать подмножество $\Lambda^* \subseteq \Lambda$, что $\forall \lambda \in \Lambda^*$ решение γ_λ^* задачи со сверткой будет эффективным, т.е. $\gamma_\lambda^* \in \{\gamma^*\}$.

Требования типа II.

IIa. Для всякого полуэффективного (слабо эффективного) решения $\gamma^o \exists \lambda \in \Lambda$, что $W(\gamma^o) \in \{W(\gamma_\lambda^*): \gamma_\lambda^* \in \arg \min_{\gamma \in \{\gamma(X^S, X^F)\}} \tilde{\Psi}_\lambda(W(\gamma))\}$.

Заметим, что сложная форма записи этого утверждения связана с возможной неединственностью решения задачи со сверткой, когда разным решениям γ_λ^* могут соответствовать одинаковые значения векторного

критерия, а также с ситуациями, когда им соответствуют разные значения векторного критерия $W(\gamma_\lambda^*)$, но одинаковые значения функции свертки.

II б. Для всякого эффективного решения $\gamma^* \exists \lambda \in \Lambda$, что $W(\gamma^*) \in \{W(\gamma_\lambda^*): \gamma_\lambda^* \in \arg \min_{\gamma \in \{\gamma(X^S, X^F)\}} \tilde{\Psi}_\lambda(W(\gamma))\}$ или же (для некоторого типа сверток) справедливо более сильное утверждение: $W(\gamma_\lambda^*) = W(\gamma^*)$, т.е. метод свертки найдет путь γ_λ^* , эквивалентный желаемому γ^* по значению векторного критерия.

Существует несколько различных способов свертки векторного критерия в скалярный. Рассмотрим только два из них.

Первый называется *методом линейной свертки*. В нем вводится скалярный критерий

$$\Psi_\lambda^L(\gamma) = \tilde{\Psi}_\lambda^L(W(\gamma)) = \sum_{i=1}^n \lambda_i W_i(\gamma), \quad \lambda = (\lambda_1, \dots, \lambda_n) \in \Lambda, \quad (12)$$

где $\Lambda = \{\lambda \in R^n: \lambda_i \geq 0, \sum_{i=1}^n \lambda_i = 1\}$,

и решается задача

$$\Psi_\lambda^L(\gamma) \Rightarrow \min_{\gamma \in \{\gamma(X^S, X^F)\}} \quad (13)$$

Известно, что её решениями будут являться некоторые из полуэффективных путей γ^0 . Изменяя коэффициенты λ_i при $\lambda \in \Lambda$ можно находить различные полуэффективные решения. Выбор набора весовых коэффициентов $\lambda_i > 0$, $(i = 1, \dots, n)$ приведет к отысканию эффективного решения γ^* , зависящего от выбранных λ_i . Таким образом интуитивно понятно, что для линейной свертки выполнены требования типа Ia и Ib.

Задание 1. Используя формальное определение полуэффективного (слабо эффективного) и эффективного путей постройте аналитические доказательства утверждений Ia и Ib из группы требований I для метода линейной свертки. Уточните, что представляют из себя подмножества Λ^0 и Λ^* . Для доказательства примените метод рассуждений от противного.

Дадим геометрическую интерпретацию метода линейной свертки. Рассмотрим поверхности равного уровня критерия свертки (12) в пространстве переменных W_1, \dots, W_n :

$$W_1 \lambda_1 + \dots + W_n \lambda_n = C \quad (14)$$

Очевидно, это будут гиперплоскости, ортогональные вектору $\lambda = (\lambda_1, \dots, \lambda_n)$. Уменьшение значения Ψ_λ^I соответствует уменьшению C , т.е. параллельному смещению гиперплоскости (14) в направлении против вектора λ (см. рис. 8). Решением задачи (13) будет путь γ^0 , соответствующий той точке множества Слейтера, через которую пройдет гиперплоскость (14) с заданной нормалью λ и минимально возможным значением C .

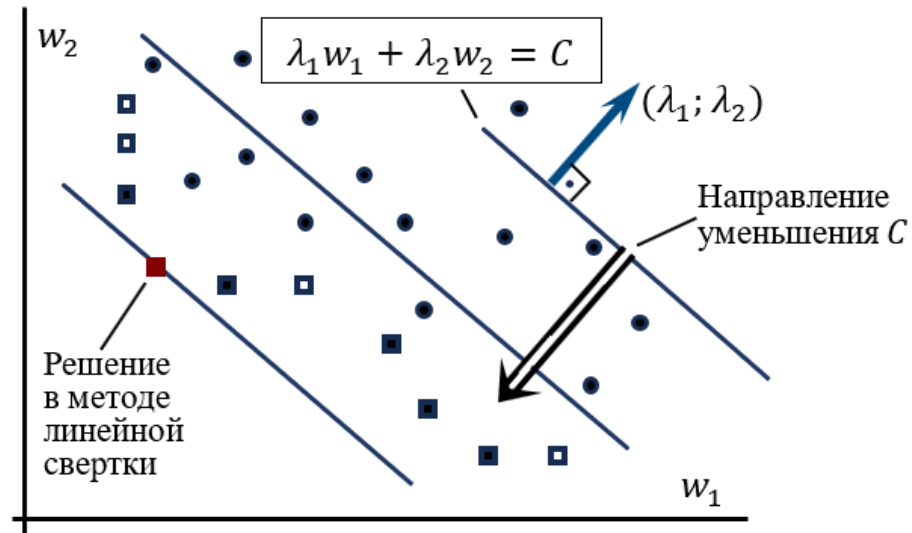


Рис. 8. – Геометрическая интерпретация метода линейной свертки при $n = 2$

Задание 2. Подумайте, выполняются ли для метода линейной свертки требования типа II? В частности, позволяет ли метод линейной свертки за счет изменения коэффициентов λ_i отыскать все полуэффективные решения? Постройте аналитическое обоснование либо приведите контрпример.

Мы рассмотрели метод линейной свертки. Другие методы свертки, в общем плане, от него ничем не отличаются за исключением вида используемого скалярного критерия свертки $\tilde{\Psi}_\lambda(w)$. Однако свойства этих свертки в аспекте выполнения требований I и II могут быть заметно иными.

Рассмотрим второй тип свертки, а именно, метод свертки Гермейера Ю.Б. В этом методе вместо задачи (13) с линейной сверткой (12) решаются задачи вида (10) с использованием в качестве $\Psi_\lambda(\gamma)$ свертки вида $\Psi_\lambda(\gamma) = \Psi_\lambda^\Gamma(\gamma) = \tilde{\Psi}_\lambda^\Gamma(W(\gamma))$ – свертки Гермейера, представленной ниже:

$$\Psi_\lambda^\Gamma(\gamma) = \tilde{\Psi}_\lambda^\Gamma(W(\gamma)) = \max_{i=1, \dots, n} \lambda_i W_i(\gamma), \quad \lambda = (\lambda_1, \dots, \lambda_n) \in \Lambda, \quad (15)$$

где $\Lambda = \{\lambda \in R^n: \lambda_i \geq 0, \sum_{i=1}^n \lambda_i = 1\}$.

Линии равного уровня этой свертки в случае $n = 2$ показаны на рис. 9.

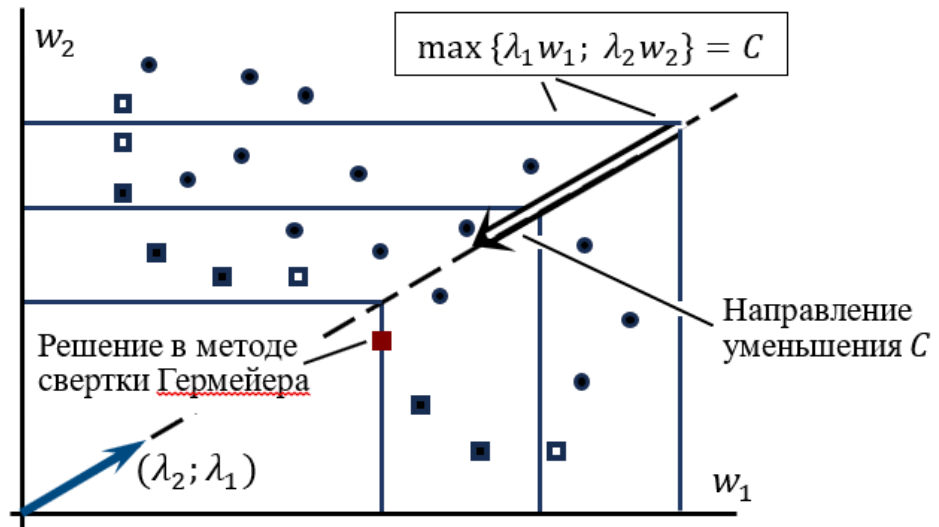


Рис. 9. – Геометрическая иллюстрация метода свертки Гермейера

Объясните, почему направляющий вектор на рис. 9 имеет координаты $(\lambda_2; \lambda_1)$, записанные в обратном порядке, подумайте, какие координаты будет иметь подобный направляющий вектор при $n > 2$.

Заметим, что метод свертки Гермейера применим только в случае расположения множества векторных стоимостей всех допустимых путей целиком в положительном октанте.

Задание 3. Используя геометрические представления о применении метода свертки Гермейера проверьте выполнение требования Ia. Как нужно выбрать при этом множество Λ^0 ? Выясните, выполняется ли для свертки Гермейера требование Ib при $\Lambda^* = \{\lambda \in R^n: \lambda_i > 0, \sum_{i=1}^n \lambda_i = 1\}$.

Задание 4. Используя геометрические представления определите, соблюдаются ли для метода свертки Гермейера требования IIa и IIb. Предложите конструктивный способ геометрического обоснования требования IIb, т.е. укажите конкретное правило выбора значений вектора весов λ для определения заданного эффективного пути γ^* .

Аккуратный аналитический анализ свойств методов линейной свертки и свертки Гермейера, а также информацию по свертке другого типа (свертке Вержбицкого) можно найти в главе 1 издания [11].

7. ПРИМЕНЕНИЕ МЕТОДОВ ДЕЙКСТРЫ И БЕЛЛМАНА В СОЧЕТАНИИ С МЕТОДОМ СВЕРТОК

Поиск эффективных и полуэффективных решений в задачах определения оптимальных путей на графах с векторными весами при использовании метода сверток сводится к отысканию путей, оптимальных в смысле используемого критерия свертки $\Psi_\lambda(\gamma)$. Для того, чтобы к возникающей задаче оптимизации (11) можно было применять эффективные методы поиска оптимального пути нужно, чтобы для этой задачи был справедлив принцип Беллмана в форме достаточного условия. Далеко не для любой функции $\Psi_\lambda(\gamma)$ это так.

Например, если для аддитивных компонент векторного критерия (8) применить свертку Гермейера (15) или для критериев типа максимума (9) применить линейную свертку (12), то в задаче оптимизации (11) принцип Беллмана выполнен не будет. Подтверждающий контрпример построен в [8] (попробуйте построить аналогичный самостоятельно!).

При согласованном выборе типа свертки и вида критерия, – (12) с (8) или (15) с (9), принцип Беллмана для задачи (11) будет справедлив, что позволит применить к решению задачи (11) метод Дейкстры или метод уравнений Беллмана. Для этого преобразуем $\Psi_\lambda(\gamma)$ к следующим видам.

Для (8), (12)

$$\Psi_\lambda^L(\gamma) = \sum_{i=1}^n \lambda_i (\sum_{g \in \gamma} C_i(g)) = \sum_{g \in \gamma} (\sum_{i=1}^n C_i(g) \lambda_i) = \sum_{g \in \gamma} \hat{C}(g), \text{ где} \\ \hat{C}(g) = \sum_{i=1}^n C_i(g) \lambda_i; \quad (16)$$

для (9), (15)

$$\Psi_\lambda^\Gamma(\gamma) = \max_{i=1, \dots, n} \lambda_i (\max_{g \in \gamma} C_i(g)) = \max_{g \in \gamma} \hat{C}(g),$$

где

$$\hat{C}(g) = \max_{i=1, n} C_i(g) \lambda_i. \quad (17)$$

Таким образом, отыскание полуэффективного решения, соответствующего выбранным значениям коэффициентов свертки в случае согласованного выбора типа критерия свелось к применению метода Дейкстры или метода уравнений Беллмана для соответствующей однокритериальной задачи с новыми критериями качества пути (1) или (2), в

которых в качестве весов ребер $C(g)$ использованы веса $\hat{C}(g)$ из (16), (17). Таким образом, в графе векторные веса ребер заменяются на скалярные веса, полученные по формулам сверток (16) или (17), после чего на графе решается полученная скалярная задача оптимизации. Оптимальный путь этой вспомогательной задачи является полуэффективным путем в исходной задаче, соответствующим выбранным значениям коэффициентов свертки. Изменяя эти коэффициенты, можно получить различные полуэффективные пути для исходной задачи. Однако отыскание по этому способу всех решений многокритериальной задачи, если их число и априорные характеристики полностью неизвестны, представляет отдельную проблему, поскольку априори неясно, какие значения весов λ_i следует выбирать.

Здесь возможно два пути. Первый путь заключается в разработке алгоритмов последовательного выбора коэффициентов λ применяемой свертки в зависимости от результатов использования их предыдущих значений.

Задание 5. Для задач с двумерными векторными критериями качества предложите метод последовательного определения значений весовых коэффициентов λ^{k+1} (с использованием информации о результатах решения задач со свертками для предыдущих значений λ^s , $s \leq k$). Метод должен через конечное число коррекций вектора весов λ^k найти все доступные для определения применяемой сверткой оптимальные по Парето точки на плоскости значений критериев.

Методы такого типа приводят к многократному применению метода сверток с априори неизвестным (хотя и конечным) числом повторений, поэтому общие вычислительные затраты суммируются.

Второй путь заключается в построении методов, позволяющих без использования сверток сразу определять все множество эффективных решений. В качестве дополнительного материала приведем краткое описание модификации метода Дейкстры для задач с векторными весами, не использующего параметрические скаляризаторы. Несмотря на возможную высокую вычислительную сложность методы такого типа представляют интерес, поскольку указанные выше альтернативные методы также вычислительно дорогостоящи.

8. МОДИФИКАЦИЯ МЕТОДА ДЕЙКСТРЫ ДЛЯ ПОСТРОЕНИЯ ВСЕГО МНОЖЕСТВА ЭФФЕКТИВНЫХ ПУТЕЙ И МНОЖЕСТВА ПАРЕТО

Среди известных алгоритмов укажем на работу [7], в которой предложен метод построения сразу всего множества Парето в многокритериальных задачах оптимизации на графах без ограничения на знаки весов ребер. Следует подчеркнуть, что этот алгоритм строит только множество Парето (т.е. находит решения в пространстве критериев), а сами эффективные пути не определяет. Кроме того, алгоритм обладает достаточно плохой оценкой сложности порядка $O(m \cdot n^{2n-3} + m \cdot n^n)$, где n – число вершин в графе, а m – размерность вектора критериев. При $m = 1$ это дает оценку сложности $O(n^{2n-3} + n^n)$, что намного хуже, чем у обычного метода Дейкстры, который для $m = 1$ обладает оценкой сложности $O(n^2)$.

В тоже время, можно предложить другой алгоритм, непосредственно *обобщающий метод Дейкстры на задачи с неотрицательными векторными весами ребер*. Попробуем выяснить, как можно построить такое обобщение (приведенная модификация метода Дейкстры для неотрицательных векторных весов предложена Городецким С.Ю. в 1996 г., ее первое описание можно найти в издании [9]). Для большей определенности рассмотрим метод для случая аддитивного векторного критерия качества (1). Для критерия вида (2) метод строится аналогично. Заметим, что определение всего множества оптимальных по Парето путей гарантируется именно для критерия (1).

ОБЩЕЕ ОПИСАНИЕ ОБОБЩЕННОГО МЕТОДА ДЕЙКСТРЫ

Принципы построения представим на примере задач с аддитивным векторным критерием. Вместо обычных меток вершин будем использовать *множества $S(x)$ векторных меток $s(x)$* . Множественная метка $S(x)$ должна являться оценкой множества Парето в задаче о поиске оптимальных путей между вершинами x^S и x . Для того, чтобы можно было восстанавливать сами эффективные пути, для каждой метки $s(x)$ из множества $S(x)$ необходимо *помечать (запоминать) ребро g^** , приход по которому в вершину x приводит к получению значения $s(x)$. Поскольку вершина \hat{x} , предшествующая вершине x для ребра $g^* = (\hat{x}, u, x)$, тоже имеет не скалярную, а множественную метку

$S(\hat{x})$, необходимо также запоминать ту из меток \hat{s} множества $S(\hat{x})$, которая порождает значение $s(x)$ по правилу $s(x) = \hat{s} + C(g^*)$.

Таким образом, на элементах $s \in S(x)$ должны быть определены два отображения $g^*(x, s)$ и $\hat{s}(x, s)$, возвращающие значения g^* и \hat{s} . Их не нужно определять только для вершины $x = x^S$, т.к. она является начальной.

В силу множественного характера меток вершин приходится изменять понятия «вершина с постоянной меткой» и «вершина с временной меткой». Теперь часть меток вершины x могут оказаться постоянными (т.е. неизменяемыми), а часть меток – временными. Для описания этого разделения из множества $S(x)$ будем выделять подмножество $S^*(x)$ постоянных меток вершины x , обозначая оставшуюся часть меток (т.е. все временные метки) через $\check{S}(x) = S(x) \setminus S^*(x)$, где « \setminus » – операция исключения.

Неизбежно изменится также правило выделения новых постоянных меток вершин, поскольку их сравнения производятся по векторным критериям. К новым постоянным меткам нужно будет относить те временные метки вершин, которые будут входить в подмножество Парето на множестве сравнения \check{V} , образованном временными метками всех вершин, а также всеми метками финишной вершины. Включение в множество \check{V} меток финишной вершины связано с тем, что временные метки вершин, «худшие» текущих меток финишной вершины x^F не могут в последующем привести к «улучшению» $S(x^F)$ (из-за неотрицательности компонент векторных весов ребер) и поэтому не должны объявляться постоянными. Это будет автоматически обеспечено именно за счет участия в сравнении меток финишной вершины.

Заметим, что поскольку метки из \check{V} должны «помнить» к каким вершинам они относятся, то на \check{V} необходимо задавать отображение $\chi(v)$ со значением на множестве вершин. Для него должно выполняться: $v \in \check{S}(\chi(v))$.

После сделанных пояснений перейдем к пошаговому описанию предлагаемого алгоритма, обобщающего метод Дейкстры на многокритериальные задачи.

АЛГОРИТМ ОБОБЩЕННОГО МЕТОДА ДЕЙКСТРЫ ДЛЯ ЗАДАЧ С ВЕКТОРНЫМИ ВЕСАМИ

ШАГ 0. *Начальные операции.* Вершине x^S присваивается множество меток, состоящее из одной нулевой точки $S(x^S) = \{0: 0 \in R^n\}$, а остальным вершинам x , отличным от x^S , – множество из одной бесконечной точки: $S(x) = \{\infty: \infty \in R^n\}$. Множества постоянных меток объявляются пустыми $S^*(x) = \emptyset$ для всех вершин x кроме x^S , а для x^S : $S^*(x) = S(x^S)$.

В качестве множества начальных меток – центров следующей итерации выбирается множество $\check{V} = \{0: 0 \in R^n\}$ вершины $\chi(0) = x^S$.

ШАГ 1. *Обновление меток.* У всех вершин x , достижимых из $\chi(v)$ для $v \in \check{V}$ обновляются множественные метки так, что за их новое значение принимаются подмножества Парето, выделяемые из старых множеств $S(x)$, дополненных новыми метками вида $v + C(\chi(v), u, x)$, где $f(\chi(v), u) = x$. Т.е. выполняется коррекция:

$$S(x) := P\left(S(x) \cup_{u: f(\chi(v), u) = x, v \in \check{V}} (v + C(\chi(v), u, x)) \right),$$

где P – оператор выделения подмножества Парето. Для меток s вновь построенного множества $S(x)$ корректируются описанные выше отображения $g^*(x, s)$ и $\hat{s}(x, s)$. А именно, если некоторая метка s входила в старое множество $S(x)$, то для неё значения $g^*(x, s)$ и $\hat{s}(x, s)$ не изменяются, а если s – метка вновь вошедшая в $S(x)$, т.е. существуют предшествующие ей метка $\hat{v} \in \check{V}$ и вершина \hat{u} , для которых $f(\chi(\hat{v}), \hat{u}) = x$, такие, что $s = \hat{v} + C(\chi(\hat{v}), \hat{u}, x)$, то $g^*(x, s) = (\chi(\hat{v}), \hat{u}, x)$ и $\hat{s}(x, s) = \hat{v}$.

ШАГ 2. *Выделение множества \check{V} новых постоянных меток – центров итерации.* Формируется множество меток F , в которое включаются все метки из $S(x^F)$ и временные метки вершин $x \neq x^F$, т.е. элементы множеств $\check{S}(x)$:

$$F = S(x^F) \cup_{x \neq x^F} \check{S}(x).$$

Из него выделяется подмножество Парето с исключенными из него постоянными метками вершины x^F :

$$F^* = P(F) \setminus S^*(x^F).$$

Метки, попавшие в F^* объявляются постоянными для соответствующих им вершин. А именно, для всех x :

$$S^*(x) := S^*(x) \cup (\check{S}(x) \cap F^*).$$

Полагаем $\check{V} = F^*$ и для $v \in \check{V}$ определяем $\chi(v) = x$, где x это та вершина, для которой $v \in S^*(x)$.

Действия, при первом выполнении шагов 1, 2 и 3 иллюстрирует рис. 10.

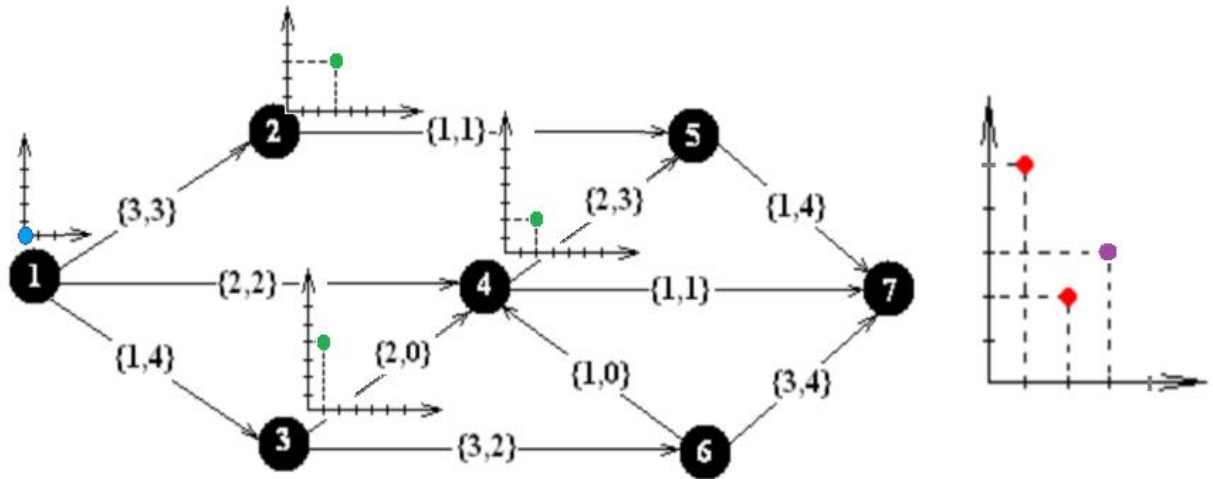


Рис. 10. – Метка (0; 0) стартовой вершины 1 (выделена синим) – начальная метка итерации, она порождает временные метки (показаны зеленым цветом) во множественных метках вершин 2, 4 и 3. Справа – сопоставление временных меток на плоскости сравнения; метка (3; 3) отвергается, метки (1; 4) и (2; 2) объявляются постоянными (выделены красным) и становятся центрами следующей итерации

Действия, при следующем выполнении шагов 1, 2 и 3 показаны на рис. 11.

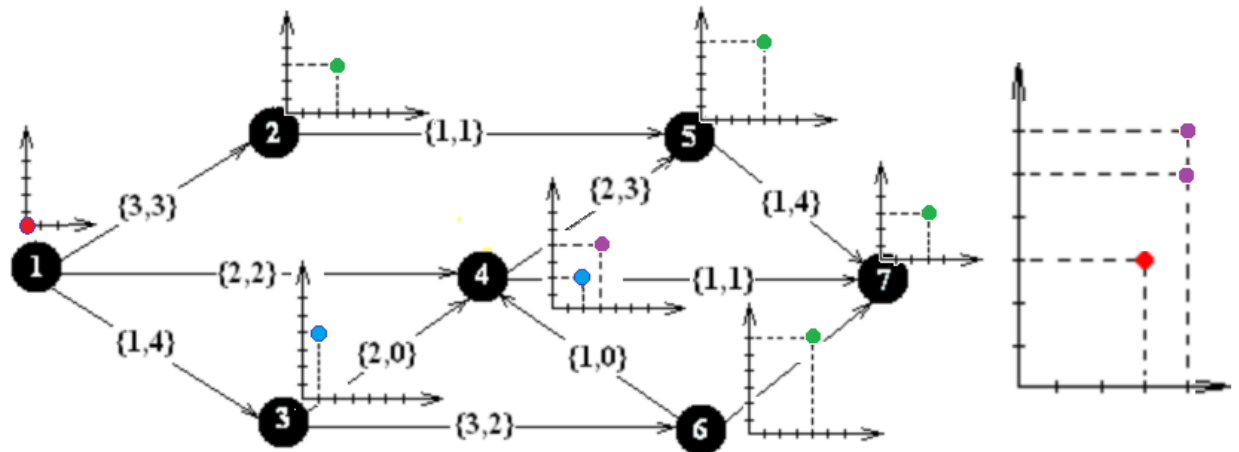


Рис. 11. – Центрами итераций становятся метки (1; 4) вершины 3 и (2; 2) вершины 4 (выделена синим), они порождают 4 новые временные метки у вершин 4, 6, 5 и 7, но метка (4; 4) вершины 4 (помечена фиолетовым) отвергается. Оставшиеся четыре временные метки (отмечены зеленым) сопоставляются на плоскости сравнения (показана справа), две совпадающие метки (3; 3) (выделены красным) объявляются постоянными и далее станут центрами итерации

ШАГ 3. *Проверка критерия остановки.* Если $\check{V} \neq \emptyset$, то происходит возврат на шаг 1. Если $\check{V} = \emptyset$, то процесс останавливается и множество эффективных путей строится с использованием множества отмеченных ребер, определяемых отображениями $g^*(x, s)$, начиная с $x = x^F$, для $s \in S^*(x^F)$.

Заметим, что в момент остановки некоторые метки вершин, не являющихся финишными, могут оставаться временными, но все метки финишной вершины обязаны уже стать постоянными.

Для того, чтобы освоить приведенный выше алгоритм рекомендуется применить его для примера графа с небольшим числом вершин и размерностью векторных весов ребер $n = 2$. Фрагмент такого применения показан на рис. 10-11.

В качестве дополнения к материалу раздела 8 укажем книгу Д. И. Когана [10], в значительной части посвященную вопросам многокритериального динамического программирования.

9. ОБЩИЕ СВЕДЕНИЯ О КОМПЬЮТЕРНОЙ ЛАБОРАТОРИИ

Теоретический материал данной методической разработки поддержан учебно-исследовательской программной лабораторией «Поиск оптимальных путей на графах с векторными весами», разработанной для семейства операционных систем Windows в лаборатории кафедры ТУиДС института ИТММ ННГУ.

Программная лаборатория снабжена иллюстрированным справочником по основным вопросам используемого теоретического материала, конструктором и редактором графов, а также средствами для выполнения вычислительных экспериментов (см. рис. 12)

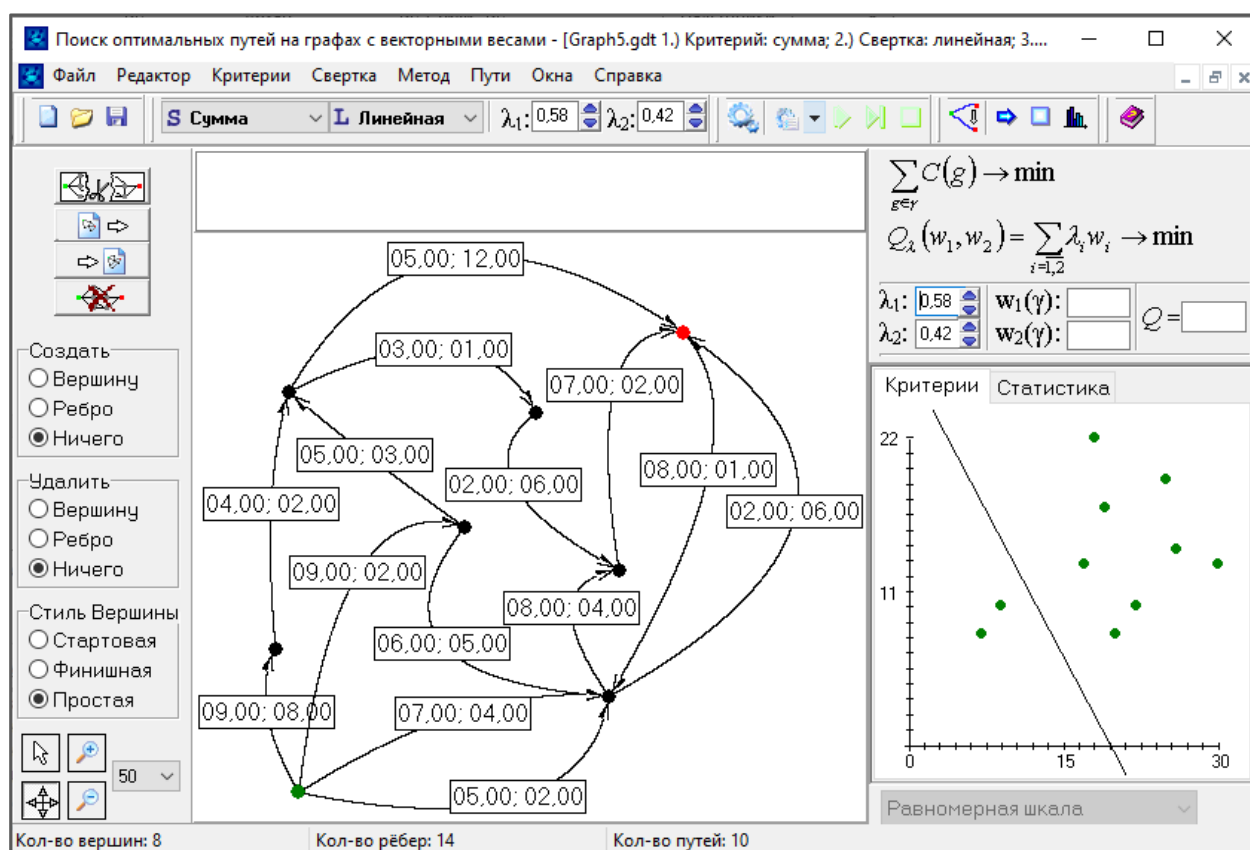


Рис. 12. – Вид основного окна программной лаборатории OptWay в режиме загрузки и редактирования графа с векторными весами ребер (зеленым выделена стартовая вершина, красным – финишная); справа на плоскости значений критериев зелеными точками показано множество векторных стоимостей путей

А именно, имеется архив типовых графов и развитый редактор графов (для случая векторных весов размерности $n = 2$), система сохранения

и загрузки отредактированных графов. Кроме того, поддерживается постановщик задач, позволяющий задать для выбранного графа множества начальных и конечных точек, определить тип оптимизируемых критериев (8) или (9), а также вид свертки (12) или (15), значения коэффициентов свертки λ_1, λ_2 . Имеется система сбора статистики по исследуемой задаче на графе, основанная на автоматическом переборе всех путей.

Важным инструментом является подсистема, выполняющая поиск оптимального пути. При этом может использоваться метод сверток в сочетании с одним из методов поиска оптимальных путей на графах со скалярными весами. Программа способна наглядно иллюстрировать работу этих методов, включая пошаговый режим демонстрации (см. рис. 13).

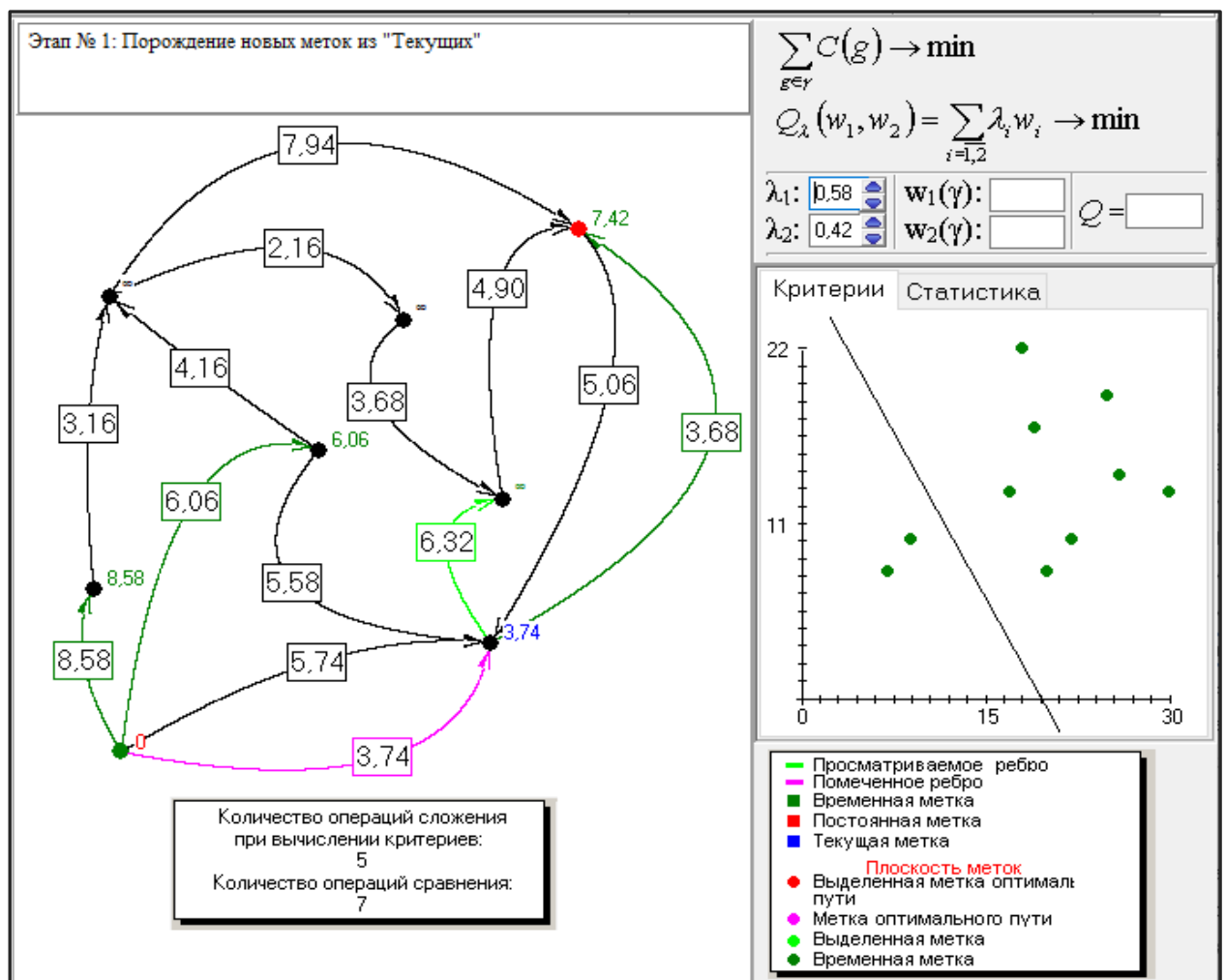


Рис. 13. – Вид рабочего окна программы OptWay в процессе пошагового выполнения метода Дейкстры (ведется подсчет текущего числа операций сложения и сравнения, отображаются метки вершин, выделяются просмотренные и помеченные ребра)

В программную лабораторию включена также реализация метода из раздела 8, не использующего свертки. На рис. 14. показан вид рабочего окна программы на промежуточном этапе решения задачи безсверточным методом.

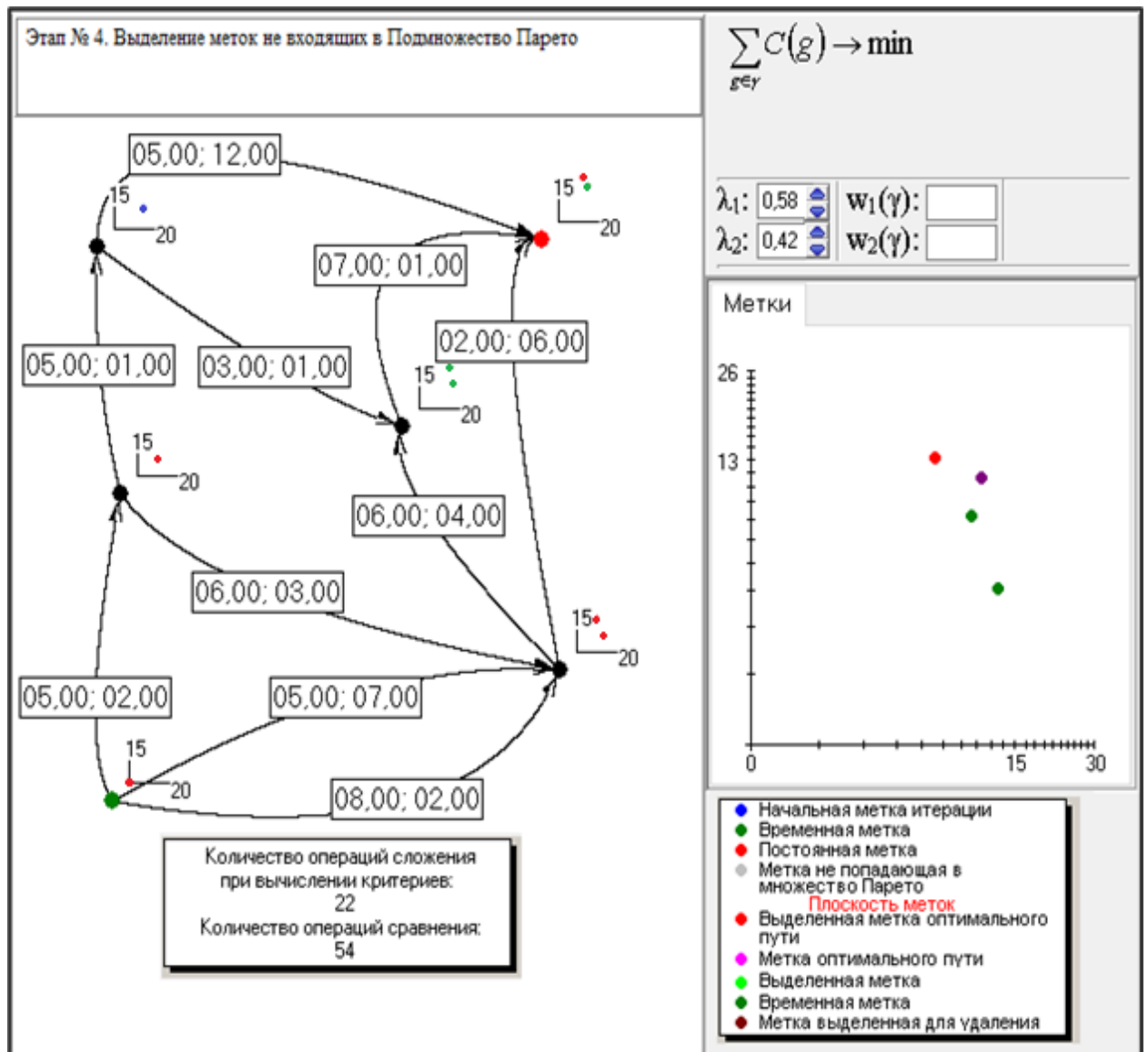


Рис. 14. – Промежуточное состояние процесса определения множества всех эффективных путей модифицированным методом Дейкстры (предложен в [9]), не использующим свертки. В процессе решения граф не скаляризуется, метод использует множественные метки вершин; при выборе новых постоянных элементов во временных частях меток вершин временные метки сравниваются как векторы на общей плоскости сравнения (на рисунке она показана справа)

10. КОНТРОЛЬНЫЕ ВОПРОСЫ И ЗАДАНИЯ

С использованием материала данной методической разработки, а также возможностей компьютерной программной лаборатории, необходимо найти ответы на следующие вопросы.

1. В чем заключается задача поиска оптимальных путей на графах с векторными весами? Что такое эффективный путь, полуэффективный? Постройте свои примеры.

2. Как отображается задача с векторными весами ребер в пространстве критериев? Что такое множество Парето и множество Слейтера? Постройте графы, для которых эти множества совпадают и для которых – не совпадают.

3. В чем заключается метод линейной свертки и свертки Гермейера? Предложите свои виды сверток. Дайте их геометрическую интерпретацию.

4. Все ли эффективные решения позволяет найти метод линейной свертки? Приведите подтверждающие Ваше мнение примеры. Выполнено ли это свойство для свертки Гермейера?

5. Выполните для метода линейной свертки задания 1 и 2 из раздела 6.

6. Выполните для метода свертки Гермейера задания 3 и 4 из раздела 6.

7. Сформулируйте принцип Беллмана в двух его формах. Всегда ли выполняется принцип Беллмана в форме необходимого условия в задачах со скалярными весами ребер для аддитивного критерия, для критерия максимума? Если нет, то приведите пример на графе всего с тремя ребрами.

8. В чем заключается и всегда ли выполняется принцип Беллмана как достаточное условие? Почему выполнение этого принципа важно для построения вычислительных методов?

9. В чем заключается требование согласованности типа свертки с типом векторного критерия? Как выглядит скаляризованная задача?

10. Постройте пример графа, для которого при несогласованном выборе типа критерия и вида свертки нарушится принцип Беллмана.

11. Почему не нарушается принцип Беллмана в форме достаточного условия при согласованном выборе?

12. Как вычисляются скалярные веса ребер в «свернутой» задаче при выборе типа свертки, согласованного с типом критерия.

13. Пронаблюдайте процесс работы метода Дейкстры и ответьте на вопросы: как корректируются метки вершин, как выбирается новая постоянная метка, когда заканчивается работа метода, зачем нужно требование неотрицательности весов ребер графа?

14. В чем Вы видите связь метода Дейкстры с принципом Беллмана в форме достаточного условия?

15. Сравните затраты на поиск оптимальных путей в методе Дейкстры и методе рекуррентных уравнений Беллмана.

16. Постройте примеры графов, структура которых обеспечит меньшие вычислительные затраты для метода Дейкстры по сравнению с методом рекуррентных уравнений Беллмана, а также примеры, приводящие к прямо противоположному результату.

17. Выполните задание 5 из раздела 7.

18. Изучите работу метода, не использующего свертки. Оцените его затраты при построении всего множества эффективных путей на конкретных примерах. Сравните результаты с подходом на основе сверток из задания 5 раздела 7.

ЛИТЕРАТУРА

1. Городецкий С.Ю. Лекции по нелинейному математическому программированию: Учебно-методическое пособие. – Нижний Новгород: Изд-во ННГУ, 2020. – 173 с. – Текст: электронный. URL: <http://e-lib.unn.ru/MegaPro/UserEntry?Action=FindDocs&ids=794639&idb=0>.
2. Беллман Р., Дрейфус О. Прикладные задачи динамического программирования. – М.: Наука, 1965. – 458 с.
3. Калихман И. Л., Войтенко М. А. Динамическое программирование в примерах и задачах: Учебное пособие. – М.: Высш. школа, 1979. – 125 с.
4. Беллман Р. Динамическое программирование. – М.: ИЛ, 1960. – 400 с.
5. Габасов Р., Кириллова Ф.М. Основы динамического программирования. – Минск: Изд-во БелГУ, 1975. – 264 с.
6. Кристофидес Н. Теория графов. Алгоритмический подход. – М.: Мир, 1978. – 432 с.
7. Corley H. W., Moon J. D. Shortest paths in networks with vector weights // J. Optim. Theory and Appl., 1985, v.46, № 1. – с.79-86.
<https://doi.org/10.1007/BF00938761>
8. Городецкая Н.И., Васин Ю.Г. К вопросу решения задачи поиска оптимального пути в сетях с векторными весами. // Математическое моделирование, управление и оптимизация.: Деп. сб. научн. трудов конф. молодых ученых НИИ ПМК и фак. ВМК ГГУ. – Горький, ГГУ, 1988. Деп. в ВИНТИ № 5714–В88. –13 с.
9. Городецкий С.Ю. Поиск оптимальных путей на графах с векторными весами: Методические указания. – Н. Новгород: ННГУ, 1996. – 28 с.
10. Коган Д. И. Динамическое программирование и дискретная многокритериальная оптимизация: Учебное пособие (серия «Модели и методы конечномерной оптимизации»; вып. 3). – Нижний Новгород: Изд-во ННГУ, 2005. – 260 с.
11. Городецкий С.Ю., Гришагин В.А. Нелинейное программирование и многоэкстремальная оптимизация: Учебное пособие (серия «Модели и методы конечномерной оптимизации»; вып. 2). – Нижний Новгород: Изд-во ННГУ, 2007. – 489 с.

Станислав Юрьевич Городецкий

**МЕТОДЫ ОПТИМИЗАЦИИ НА ГРАФАХ
С ВЕКТОРНЫМИ ВЕСАМИ РЕБЕР**

Учебно-методическое пособие

Федеральное государственное автономное образовательное учреждение
высшего образования «Национальный исследовательский
Нижегородский государственный университет им. Н.И. Лобачевского»
603022, Нижний Новгород, пр. Гагарина, 23