

Министерство науки и высшего образования Российской Федерации
НОВОСИБИРСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ

Д.В. ВАГИН
И.И. ПАТРУШЕВ
А.М. ГРИФ

ЦИФРОВЫЕ МОДЕЛИ И ОЦЕНИВАНИЕ ПАРАМЕТРОВ

Утверждено Редакционно-издательским советом университета
в качестве учебного пособия

НОВОСИБИРСК
2021

УДК 53:51(075.8)
В 124

Рецензенты:

д-р техн. наук, профессор *М.Э. Рояк*
канд. техн. наук, ведущий инженер-программист
АО «Завод Экран» *А.В. Волкова*

Работа подготовлена на кафедре прикладной математики НГТУ

Вагин Д.В.

В 124 Цифровые модели и оценивание параметров: учебное пособие / Д.В. Вагин, И.И. Патрушев, А.М. Гриф. – Новосибирск: Изд-во НГТУ, 2021. – 59 с.

ISBN 978-5-7782-4391-0

В данном учебном пособии рассмотрены элементы теории из раздела численных методов решения обратных задач и машинного обучения. Пособие может быть рекомендовано как для самостоятельного изучения курса «Цифровые модели и оценивание параметров», так и для подготовки к выполнению практических заданий.

УДК 53:51(075.8)

ISBN 978-5-7782-4391-0

© Вагин Д.В., Патрушев И.И., Гриф А.М., 2021
© Новосибирский государственный
технический университет, 2021

ВВЕДЕНИЕ

Изучение физических свойств объектов при невозможности их прямого измерения приводит, как правило, к необходимости решения обратных задач. Такие задачи возникают, например, при поиске залежей полезных ископаемых, изучении геологической структуры литосферы Земли, оптимальном планировании работ, проектировании и во множестве других научных и технических областей.

В данном учебном пособии рассматриваются основные элементы теории численного решения обратных задач. Среди них:

- Основные определения, связанные с обратными задачами;
- Регуляризация обратных задач;
- МНК функционал обратной задачи;
- Решение обратной задачи методом Гаусса–Ньютона;
- Вопросы параметризации обратных задач;
- Генетический алгоритм для решения специального класса обратных задач;
- Основные идеи и определения, связанные с машинным обучением.

Дополнительно к теоретическому материалу приводятся примеры решения некоторых задач.

Данное учебное пособие основано на работе [1].

1. ОСНОВНЫЕ ОПРЕДЕЛЕНИЯ

Математические модели любых физических задач включают в себя множество скалярных или векторных величин, которые могут быть постоянными или являться функциями других величин, входящих в математическую модель. При решении задачи обычно выделяют три основные группы входящих в модель величин: 1) *исходные (входные) данные* x ; 2) *параметры модели* a ; 3) *результаты (выходные данные)* y [2]. Тогда постановка задачи может выглядеть следующим образом: по данным значениям входных данных x при фиксированных значениях параметров a требуется найти решение y . Такие задачи называются *прямыми*. Их решение можно рассматривать как математическое моделирование причинно-следственной связи, присущей явлению. Тогда входные данные x соответствуют «причинам», а выходные данные y – «следствиям». Поясним это на примере. Пусть исследуется движение тела, брошенного со скоростью v_0 под углом α к поверхности Земли (рис. 1).

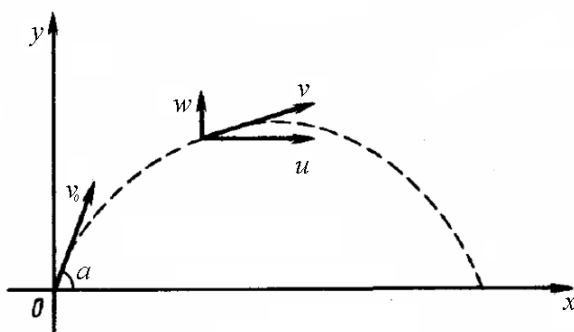


Рис. 1. Пример математической модели

Математическая модель этой задачи включает в себя: систему координат Oxy ; время t ; скорость $v = v(t)$, которая разлагается соответственно на горизонтальную составляющую u и вертикальную составляющую w ; начальную скорость v_0 и угол α . Прямую задачу естественно сформулировать как задачу нахождения функций $x(t)$, $y(t)$, $u(t)$, $w(t)$ по задаваемым входным данным v_0 , α . Параметром модели является величина ускорения свободного падения g .

На практике также возникает необходимость решения *обратных* задач. В этом случае искомыми являются входные данные x , а известными – результаты y . Для рассматриваемой математической модели (рис. 1) обратную задачу можно сформулировать так: по заданным $x(t)$, $y(t)$, $u(t)$, $w(t)$ требуется найти значения v_0 , α . Решение обратной задачи – это в некотором смысле попытка установить, какие «причины» x привели к известным «следствиям» y .

Далее для любой (не важно, прямой или обратной) задачи, которую мы будем рассматривать, условимся называть входными данными известные величины $s \in S$ и выходными данными – искомые величины $r \in R$.

Ранее считалось, что для того чтобы решение математической задачи имело смысл, т.е. задача была *корректной* (по Адамару–Петровскому), она должна удовлетворять трем условиям: 1) решение $r \in R$ *существует* для любых входных данных $s \in S$; 2) это решение *единственно*; 3) решение *устойчиво по отношению к малым возмущениям входных данных*. Если какое-либо из этих условий нарушается, задача называется *некорректной*. Если нарушается первое условие, т.е. для каких-то s не существует r , то это может свидетельствовать о дефекте математической модели, описывающей явление. Требование единственности решения для ряда задач абсолютно естественно, тогда как для других (например, отыскание корней полинома) мы можем принять за решение набор удовлетворяющих нас значений или, сузив область входных данных S , добиться того, чтобы решение было единственным. Требование устойчивости решения по входным данным означает, что оно должно зависеть от входных данных непрерывным образом. То есть для любого $\varepsilon > 0$ должна существовать $\delta = \delta(\varepsilon) > 0$ такая, что всякому исходному данному s^* , удовлетворяющему условию $\Delta(s^*) < \delta$, соответствует приближённое решение r^* , для которого

$\Delta(r^*) < \varepsilon$, где Δ означает абсолютную погрешность числа. Таким образом, для устойчивой задачи её решение теоретически можно найти со сколь угодно высокой точностью ε , если обеспечена достаточно высокая точность входных данных δ . Неустойчивость решения r означает, что существует такое $\varepsilon > 0$, что какое бы малое $\delta > 0$ ни было задано, найдутся такие исходные данные s^* , что $\Delta(s^*) < \delta$, но $\Delta(r^*) \geq \varepsilon$. Приведём примеры некоторых устойчивых и неустойчивых задач. Устойчивыми задачами являются, например: задача о вычислении корней полинома (так как корни полинома являются непрерывными функциями его коэффициентов) и задача вычисления определённого интеграла. Неустойчивыми задачами являются, например: задача о вычислении ранга матрицы и задача вычисления производной приближённо заданной функции.

Неустойчивость задачи о вычислении ранга матрицы легко продемонстрировать, взяв матрицу $A = \begin{pmatrix} 1 & 0 \\ 0 & 0 \end{pmatrix}$. Её ранг равен 1. Но если мы немного изменим коэффициент: $a_{22} = \varepsilon$, то ранг станет равен 2.

Для демонстрации неустойчивости задачи дифференцирования возьмём приближённую функцию $f^*(x)$ такую, что $f^*(x) = f(x) + \alpha \sin(x/\alpha^2)$, $0 < \alpha \ll 1$. Положим $u(x) = f'(x)$, $\Delta(f^*) = \max_{[a,b]} |f(x) - f^*(x)|$, $\Delta(u^*) = \max_{[a,b]} |u(x) - u^*(x)|$, где $u^*(x) = (f^*)'(x)$. Очевидно, что $u^*(x) = u(x) + \alpha^{-1} \cos(x/\alpha^2)$. И, следовательно, при том, что $\Delta(f^*) = \alpha$, мы получаем $\Delta(u^*) = \alpha^{-1}$. То есть сколь угодно малой погрешности f может отвечать сколь угодно большая погрешность производной f' .

Заметим также, что одна и та же задача может оказаться как устойчивой, так и неустойчивой в зависимости от выбора способа вычисления абсолютных погрешностей $\Delta(s^*)$ и $\Delta(r^*)$. К тому же часто на практике требование малости абсолютной погрешности является избыточным и вместо абсолютной устойчивости достаточно наличия *относительной устойчивости*. Её определение отличается от определения

абсолютной устойчивости только тем, что $\Delta(s^*)$ и $\Delta(r^*)$ заменяются на $\delta(s^*)$ и $\delta(r^*)$ соответственно.

Длительное время считалось, что некорректные задачи, решения которых неустойчивы, не имеют физического смысла и не представляют ценности для приложений. Однако это не так. Многие важные прикладные задачи некорректны, например, обратные задачи геофизики, спектрографии, многие задачи распознавания образов, задачи синтеза и др.

Для устойчивой вычислительной задачи важным свойством является ее *обусловленность*. Дело в том, что наличие устойчивости для конкретной задачи вовсе не означает, что её решение действительно удастся получить с любой желаемой точностью, хотя бы потому, что на практике погрешность входных данных ограничена снизу точностью измерительной аппаратуры, к тому же при вычислениях неизбежно возникают ошибки округления, связанные с неточностью представления вещественных чисел на ЭВМ. Обусловленность отражает чувствительность решения задачи к погрешности входных данных. Задачу называют *хорошо обусловленной*, если малым погрешностям входных данных отвечают малые погрешности решения. Иначе задачу называют *плохо обусловленной*. Если для задачи удастся ввести некоторую числовую оценку связывающую погрешности входных данных и решения, то её называют *числом обусловленности задачи*. То есть если $\Delta(r^*) \leq v_\Delta \Delta(s^*)$ или $\delta(r^*) \leq v_\delta \delta(s^*)$, то v_Δ и v_δ – соответственно числа абсолютной и относительной обусловленности задачи. На практике обычно наиболее полезным является v_δ . Так, если $v_\delta \approx 10^N$, то N показывает число верных цифр, которое может быть утеряно в результате по сравнению с числом верных цифр входных данных. В качестве примеров плохо обусловленных задач можно привести задачу вычитания приближённых чисел одного знака и задачу вычисления корней полиномов высокого порядка. В первом случае $v_\delta = |a+b|/|a-b|$ и при близких a и b v_δ может быть очень большим. Второй случай можно проиллюстрировать на классическом примере Дж. Уилкинсона. Пусть полином $P(x)$ имеет вид

$$P(x) = (x-1)(x-2)\dots(x-20) = x^{20} - 210x^{19} + \dots$$

Возьмём коэффициент $a = -210$ при x^{19} и немного изменим его значение: $a^* = -210 + 2^{-23}$. Корни изменённого полинома будут приближенно равны:

$$\begin{aligned}x_1^* &\approx 1.00000, & x_2^* &\approx 2.00000, & x_3^* &\approx 3.00000, & x_4^* &\approx 4.00000, \\x_5^* &\approx 5.00000, & x_6^* &\approx 6.00001, & x_7^* &\approx 6.99970, & x_8^* &\approx 8.00727, \\x_9^* &\approx 8.91725, & x_{10,11}^* &\approx 10.0953 \pm 0.643501i, & x_{12,13}^* &\approx 11.7936 \pm 1.65233i, \\x_{14,15}^* &\approx 13.9924 \pm 2.51883i, & x_{16,17}^* &\approx 16.7307 \pm 2.81262i, \\x_{18,19}^* &\approx 19.5024 \pm 1.94033i, & x_{20}^* &\approx 20.8469.\end{aligned}$$

Видно, что разные корни найдены с разной точностью. Если оценить $v = v_8$ для этих корней, то получим:

$$\begin{aligned}v_1 &\approx 2 \cdot 10^{-15}, & v_2 &\approx 9 \cdot 10^{-8}, & v_3 &\approx 10^{-4}, & v_4 &\approx 10^{-1}, & v_5 &\approx 3 \cdot 10^1, \\v_6 &\approx 2 \cdot 10^3, & v_7 &\approx 8 \cdot 10^4, & v_8 &\approx 2 \cdot 10^6, & v_9 &\approx 2 \cdot 10^7, & v_{10} &\approx 2 \cdot 10^8, \\v_{11} &\approx 9 \cdot 10^9, & v_{12} &\approx 4 \cdot 10^9, & v_{13} &\approx 10^{10}, & v_{14} &\approx 2 \cdot 10^{10}, & v_{15} &\approx 3 \cdot 10^{10}, \\v_{16} &\approx 3 \cdot 10^{10}, & v_{17} &\approx 3 \cdot 10^{10}, & v_{18} &\approx 10^{10}, & v_{19} &\approx 3 \cdot 10^9, & v_{20} &\approx 5 \cdot 10^8.\end{aligned}$$

Подчеркнём, что обусловленность является свойством задачи и не зависит от выбранного метода её решения и точности, с которой будут выполняться вычисления.

2. КЛАССИФИКАЦИЯ ОБРАТНЫХ ЗАДАЧ МАТЕМАТИЧЕСКОЙ ФИЗИКИ

Разделение прямых и обратных задач по принципу причина – следствие весьма универсально, но есть исключения. Иногда под обратными задачами понимают задачи, в которых просто не хватает какой-либо информации для того, чтобы считать их корректными прямыми задачами математической физики. Поэтому под обратными задачами часто просто понимают те задачи, которые невозможно отнести к прямым. Как результат, обратные задачи классифицируют по тому, какой именно информации не хватает для того, чтобы задача была прямой. Таким образом, классификация может быть следующей [3].

1. *Коэффициентные обратные задачи.* Эти задачи характеризуются неизвестными коэффициентами или правой частью уравнения. В качестве характерного примера можно привести параболическое уравнение

$$\frac{\partial u}{\partial t} = \frac{\partial}{\partial x} \left(k(x) \frac{\partial u}{\partial x} \right) + f(x, t), \quad 0 < x < l, \quad 0 < t \leq T.$$

Простейшая прямая задача состоит в нахождении функции $u(x, t)$, удовлетворяющей данному уравнению и начальным и краевым условиям:

$$u(0, t) = 0, \quad u(l, t) = 0, \quad 0 < t \leq T,$$

$$u(x, 0) = u_0(x), \quad 0 < x < l.$$

В прикладных проблемах часто свойства среды неизвестны и их нужно определять. В нашем случае можно поставить задачу идентификации коэффициента $k(x)$. В простейшем случае однородной среды неизвестным является коэффициент $k(x) = \text{const}$, для кусочно-одно-

родной среды – несколько констант. При зависимости свойств среды интерес может представлять коэффициентная обратная задача по восстановлению $k = k(u)$.

Список возможных постановок коэффициентных обратных задач не исчерпывается вышеотмеченными и легко может быть продолжен. Характерной является задача по нахождению пары неизвестных функций $\{u(x, t), k(x)\}$. Основная особенность рассматриваемой обратной задачи состоит в нелинейности коэффициентной обратной задачи.

Можно выделить, как самостоятельную, задачу определения неизвестной правой части $f(x, t)$. Более частные постановки связаны, например, с выбором зависимости $f(x, t) = \eta(t)\psi(x)$.

Интерес может представлять неизвестная зависимость источника (правой части) от времени при известном распределении по пространству – функция $\eta(t)$ неизвестна, а функция $\psi(x)$ задана.

2. Граничные обратные задачи. Эти задачи заключаются в идентификации недостающих граничных условий. Например, для приведённого выше уравнения значения функции $u(x, t)$ при $x = l$ неизвестны, но известно значение в некоторой внутренней точке x^* – $u(x^*, t) = \varphi(t)$. Типичная постановка граничной обратной задачи состоит в идентификации потока на части границы, недоступной измерению.

3. Эволюционные обратные задачи. В таких задачах идентифицируются начальные условия. Применительно к рассматриваемой задаче это означает, что вместо условия $u(x, 0) = u_0(x, 0)$ задано условие $u(x, T) = u_T(x)$. Необходимо найти решение в предыдущие моменты времени (*ретроспективная обратная задача*).

3. РЕГУЛЯРИЗАЦИЯ ОБРАТНЫХ ЗАДАЧ

Рассмотрим задачу

$$u = Az . \quad (3.1)$$

Здесь u – это вектор размерности $m \times 1$; A – известная матрица размерности $m \times n$; z – вектор размерности $n \times 1$. Если вектор z известен, а вектор u является искомым, то (3.1) – это прямая задача, решаемая простым умножением матрицы на вектор. Если же, наоборот, неизвестным является вектор z при известном u , то задача (3.1) является простейшим примером обратной задачи и её решение уже не столь тривиально.

Формально решение z можно найти как

$$z = A^{-1}u . \quad (3.2)$$

Однако оператор A^{-1} существует лишь для квадратных невырожденных матриц. Не составляет труда свести обратную задачу (3.1) к системе с квадратной матрицей. Для этого домножим левую и правую части (3.1) на A^T . Тогда z можно будет найти как

$$z = (A^T A)^{-1} A^T u . \quad (3.3)$$

Если система (3.1) переопределена, то её точного решения не существует и (3.3) – это её решение в смысле МНК, которое наиболее часто используется на практике. Если система (3.1) была недоопределена (далее будем рассматривать именно эту ситуацию), то, так как $A^T A$ – это квадратная матрица размерности $n \times n$, а $A^T u$ – вектор размерности $n \times 1$, получается, что система (3.3) аналогична системе (3.2), но с квадратной матрицей. Поэтому, не нарушая общности,

вернёмся к системе (3.2), положив при этом, что матрица A имеет размерность $n \times n$, а векторы z и u – размерность $n \times 1$.

Теперь нахождению z мешает только вырожденность матрицы A . Что означает, что решение z не единственно (т. е. обратная задача (2.1) является некорректной). В этом случае говорят, что существует множество эквивалентных решений, удовлетворяющих (3.1).

Большое количество практических задач представимо в виде (3.1), при этом возможно, что $A = A(z)$. Например, в обратных задачах геофизики в качестве неизвестного z выступают параметры среды (удельная электрическая проводимость и др.), а в качестве известного u – сигналы в приёмных установках. Обратные задачи геофизики теоретически могут быть корректными (иметь единственное решение), но для этого необходимо провести измерения во всех точках пространства, на всех частотах или временах, что на практике конечно невозможно. Неединственность решения не означает, что все решения разумны. Следовательно, область эквивалентности может быть существенно сокращена с помощью ограничений на допустимые значения параметров. Однако и эти ограничения могут не избавить нас от неединственности решения. Тогда задачу (3.1) необходимо «изменить», например, так:

$$u = (A + \alpha I)\tilde{z}, \quad (3.4)$$

где I – это единичная матрица и $\alpha > 0$ – некоторое вещественное положительное число. Добавление слагаемого αI к матрице системы называется *регуляризацией Тихонова*. Система (3.4) при некотором $\alpha \geq \tilde{\alpha}$ ($\tilde{\alpha}$ в общем случае зависит от A и u) имеет единственное и устойчивое решение [4]:

$$\tilde{z} = (A + \alpha I)^{-1}u. \quad (3.5)$$

То есть задача (3.4) является корректной. Вообще *регуляризацией* (задачи, решения) называется любое изменение задачи, делающее её корректной. При этом желательно, чтобы регуляризованное решение «несильно» увеличивало невязку исходной системы (3.1), т. е.

$$\|u - A\tilde{z}\| \rightarrow \min_{\alpha}, \quad \tilde{z} \neq 0. \quad (3.6)$$

По сути, в нашем случае регуляризация состояла в том, что мы исходную некорректную задачу (3.1) заменили корректной задачей (3.4). Требование $\alpha \geq \tilde{\alpha}$ и условие (3.6) означают, что α следует выбрать равным $\tilde{\alpha}$. На практике оптимальное значение $\tilde{\alpha}$ не известно (хотя иногда его можно оценить), и тогда либо выбирается $\alpha \gg \tilde{\alpha}$, либо (если это возможно и вычислительно не слишком накладно) подбирается в процессе решения задачи (3.4).

ВЫБОР ПАРАМЕТРА РЕГУЛЯРИЗАЦИИ

В теории методов приближённого решения некорректных задач значительное внимание уделяется вопросу выбора параметра регуляризации. Наибольшее распространение получили: выбор параметра регуляризации по невязке, обобщённой невязке (при учёте погрешности в задании не только в правой части, но и оператора A), квазиоптимальный выбор и др. Выбор оптимального значения параметра регуляризации во многом определяют работоспособность вычислительного алгоритма.

Параметр регуляризации α должен быть согласован с погрешностью входных данных, и чем меньше погрешность, тем меньше может быть α . Однако при уменьшении параметра регуляризации растёт погрешность решения и начинает проявляться некорректность обратной задачи. Тем самым имеется некоторый оптимум для параметра регуляризации, при котором погрешность приближённого решения минимальна.

Оптимальное значение параметра регуляризации зависит не только от величины погрешности в задании правой части, но и от класса априорных ограничений на решение.

Часто на практике применяется выбор параметра регуляризации по невязке (3.6). Обозначим

$$\varphi(\alpha) = \|u - A\tilde{z}\|. \quad (3.7)$$

Будем искать оптимальное значение α из условия

$$\varphi(\alpha) = \delta, \quad (3.8)$$

где δ – погрешность задания правой части. В достаточно общих условиях $\varphi(\alpha)$ является неубывающей и уравнение (3.8) имеет решение.

Для его приближённого решения применяются различные вычислительные процедуры. Например, используется последовательность

$$\alpha_k = \alpha_0 q^k, \quad q > 0, \quad k = 0, 1, \dots \quad (3.9)$$

Для приближённого решения (3.8) можно использовать и более быстро сходящиеся методы. Установлено, что функция $\psi(\beta) = \varphi(1/\beta)$ является убывающей и выпуклой функцией. Поэтому для решения уравнения $\psi(\beta) = \delta$ можно использовать метод Ньютона:

$$\beta_{k+1} = \beta_k - \frac{\psi(\beta_k) - \delta}{\psi'(\beta_k)}.$$

Этот метод будет сходиться при любом начальном приближении $\beta_0 > 0$. Для того чтобы не вычислять производную $\psi'(\beta_k)$, можно использовать метод секущих:

$$\beta_{k+1} = \beta_k - \frac{\beta_k - \beta_{k-1}}{\psi(\beta_k) - \psi(\beta_{k-1})} (\psi(\beta_k) - \delta).$$

Имеются и другие способы выбора параметра регуляризации. Отметим лишь тот общий момент, что выбор параметра регуляризации приводит к увеличению вычислительной работы и носит в той или иной степени итерационный характер.

Также отметим, что существуют методы итерационного решения некорректных задач, в которых в качестве параметра регуляризации выступает число итераций, которое должно быть согласованным с δ .

Эти методы по своей сути являются итерационными методами минимизации функционала невязки $\|u - Az\|^2 \rightarrow \min$.

4. МНК ФУНКЦИОНАЛ ОБРАТНОЙ ЗАДАЧИ, МЕТОД ГАУССА–НЬЮТОНА

Предположим (для конкретности), что мы решаем обратную задачу геофизики (для ещё большей конкретности – задачу электроразведки). Тогда у нас в качестве входных данных выступают сигналы в приёмных измерительных установках. В качестве искомых параметров выступают геометрические и электрофизические параметры среды. Обозначим результаты измерений в приёмниках как $\tilde{\varepsilon}_k^i$, $i=1...N^r$, $k=1...N^t$, где N^r – это число приёмников, N^t – число моментов времени, в которых фиксировались значения. Положим, что искомыми являются m параметров среды. То есть у нас есть вектор параметров $\mathbf{p}=(p_1, p_2, ..., p_m)$. Для любого конкретного \mathbf{p} можно решить прямую задачу и получить $\varepsilon_k^i(\mathbf{p})$. Естественно искать вектор \mathbf{p} из условия минимума уклонения значений $\varepsilon_k^i(\mathbf{p})$ от $\tilde{\varepsilon}_k^i$. Если для оценки этого уклонения использовать норму пространства L_2 , то получим

$$\sum_i^{N^r} \sum_k^{N^t} \left(\varepsilon_k^i(\mathbf{p}) - \tilde{\varepsilon}_k^i \right)^2 \rightarrow \min_{\mathbf{p}}. \quad (4.1)$$

Сумму неотрицательных значений в левой части (4.1) естественно понимать как сумму штрафов за отклонения значений $\varepsilon_k^i(\mathbf{p})$ от $\tilde{\varepsilon}_k^i$. При этом, если значения $\tilde{\varepsilon}_k^i$ существенно отличаются друг от друга по абсолютной величине, то и разности $\varepsilon_k^i(\mathbf{p}) - \tilde{\varepsilon}_k^i$ будут отличаться так же сильно. А после возведения в квадрат вклады слагаемых станут

совсем не сопоставимы. Поэтому часто вместо (4.1) используют взвешенные суммы

$$\sum_i^{N^r} \sum_k^{N^t} \left(w_k^i \left(\varepsilon_k^i(\mathbf{p}) - \tilde{\varepsilon}_k^i \right) \right)^2 \rightarrow \min_{\mathbf{p}}, \quad (4.2)$$

где w_k^i – это коэффициенты, с помощью которых можно или выровнять влияние на сумму значений $\tilde{\varepsilon}_k^i$ (положив, например, $w_k^i = 1 / \tilde{\varepsilon}_k^i$), или, наоборот, понизить влияние на сумму тех значений $\tilde{\varepsilon}_k^i$, которые, например, имеют большую ошибку измерений (сильно зашумлены).

Сумма в формуле (4.2) есть не что иное, как взвешенный МНК функционал невязки измеренных и рассчитанных сигналов в приёмных установках. Для нахождения его минимума, в принципе, можно использовать любой метод поиска минимума многопараметрической неявно заданной функции. Выбор наиболее эффективного метода минимизации во многом определяется вычислительной сложностью нахождения значений целевой функции и её производных и скоростью сходимости метода. Практика решения реальных обратных задач геофизики показала, что лучшими являются методы, использующие линеаризацию некоторых входящих в функционал выражений или функций. Основным здесь является метод Гаусса–Ньютона. Применим его для решения (4.2).

Введём обозначение: $\delta \varepsilon_k^i(\mathbf{p}) = \left(\varepsilon_k^i(\mathbf{p}) - \tilde{\varepsilon}_k^i \right)$ и продифференцируем сумму из (4.2) по параметрам, приравняем полученные выражения к нулю. В результате получим

$$2 \sum_i^{N^r} \sum_k^{N^t} \left(w_k^i \right)^2 \delta \varepsilon_k^i(\mathbf{p}) \frac{\partial \left(\delta \varepsilon_k^i(\mathbf{p}) \right)}{\partial p_j} = 0, \quad j = 1, 2, \dots, m. \quad (4.3)$$

Таким образом, мы получили систему из m в общем случае нелинейных по \mathbf{p} уравнений. Заметим, что коэффициент 2 в левой части можно отбросить (что мы и сделаем).

Выполним линеаризацию $\varepsilon_k^i(\mathbf{p})$ в окрестности точки \mathbf{p} . С помощью разложения в ряд Тейлора, взяв только слагаемые до первой производной, получаем

$$\delta\varepsilon_k^i(\mathbf{p}) \approx \delta\varepsilon_k^i(\mathbf{p} - \Delta\mathbf{p}) + \sum_{j=1}^m \Delta p_j \frac{\partial(\delta\varepsilon_k^i(\mathbf{p} - \Delta\mathbf{p}))}{\partial p_j}. \quad (4.4)$$

Положим $\mathbf{p} - \Delta\mathbf{p} = \mathbf{p}^n$, тогда \mathbf{p}^n – это некоторый начальный (заданный, $n = 0$) или текущий (внутри итерационного процесса, $n > 0$) вектор значений параметров, тогда $\mathbf{p} = \mathbf{p}^{n+1}$ – новый вектор значений параметров, вычисляемый на данной итерации. Следует отметить, что скорость сходимости метода Гаусса–Ньютона сильно зависит от близости \mathbf{p}^n к «истинному» \mathbf{p} . Поэтому выбор хорошего начального приближения \mathbf{p}^0 является отдельной и весьма непростой задачей.

Подставив (4.4) в (4.3), получим линейную систему из m уравнений для m неизвестных – Δp_j вида [5]

$$\mathbf{A}\Delta\mathbf{p} = \mathbf{b}, \quad (4.5)$$

где

$$A_{qs} = \sum_i^{N^r} \sum_k^{N^t} (w_k^i)^2 \frac{\partial(\delta\varepsilon_k^i(\mathbf{p}^n))}{\partial p_q} \frac{\partial(\delta\varepsilon_k^i(\mathbf{p}^n))}{\partial p_s}, \quad (4.6)$$

$$b_q = - \sum_i^{N^r} \sum_k^{N^t} (w_k^i)^2 \delta\varepsilon_k^i(\mathbf{p}^n) \frac{\partial(\delta\varepsilon_k^i(\mathbf{p}^n))}{\partial p_q}.$$

Осталось определиться с входящими в (4.6) производными по параметрам. Их можно приближённо вычислить, например, по формуле

$$\frac{\partial(\delta\varepsilon_k^i(\mathbf{p}^n))}{\partial p_q} \approx \frac{\delta\varepsilon_k^i(\mathbf{p}^n + \delta p_q) - \delta\varepsilon_k^i(\mathbf{p}^n)}{\delta p_q}. \quad (4.7)$$

Из (4.7) следует, что для вычисления значения одной производной требуется решать дополнительную прямую задачу (для нахождения $\varepsilon_k^i(\mathbf{p}^n + \delta p_q)$). Очевидно, решение прямых задач для расчёта производных и будет составлять основное время решения обратной задачи.

Если матрица (4.5) вырождена, то это означает, что решение обратной задачи будет неединственным. Для преодоления неединственности решения в (4.2) нужно внести соответствующие регуляризующие добавки. Использование классической регуляризации Тихонова для (4.2) даёт

$$\sum_i^{N^r} \sum_k^{N^l} \left(w_k^i \delta \varepsilon_k^i(\mathbf{p}) \right)^2 + \alpha \sum_j^m \left(p_j^n - \bar{p}_j \right)^2 \rightarrow \min_{\mathbf{p}}. \quad (4.8)$$

Сумма при α налагает штрафы на отклонения абсолютных значений искомых параметров от некоторых известных \bar{p}_j , что приводит к невырожденности матрицы системы:

$$(\mathbf{A} + \alpha \mathbf{I}) \Delta \mathbf{p} = \mathbf{b} - \alpha (\mathbf{p}^n - \bar{\mathbf{p}}), \quad (4.9)$$

где \mathbf{I} – это единичная матрица. Значение α , как уже было сказано раньше, может быть подобрано постепенным увеличением от некоторого достаточно малого значения или, если первое невозможно, то «на глаз» (не самый лучший вариант). Иногда вместо $\alpha \mathbf{I}$ удобно использовать матрицу $\mathbf{I} \cdot \boldsymbol{\alpha}$, где $\boldsymbol{\alpha} = (\alpha_1, \alpha_2, \dots, \alpha_m)$. То есть для каждого параметра используется «своя» α_j , что позволяет добиться физичности значений параметров p_j с наименьшим уроном для функционала (4.8), в котором в этом случае второе слагаемое имеет вид $\sum_j^m \alpha_j (p_j^n - \bar{p}_j)^2$. Если \bar{p}_j не известно, то его можно взять равным p_j^n .

Другим вариантом регуляризации является сглаживание значений геометрически близко расположенных однотипных параметров. Это актуально, например, если параметрами являются значения удельной

электрической проводимости в обратных задачах электроразведки. Искомый объект (нефтегазоносный слой или рудная залежь) может занимать большой объём пространства при примерно одинаковых средних значениях электрофизических характеристик. Если размеры ячеек (на которые разбита исследуемая область) значительно меньше такого объекта, то разумно потребовать плавного изменения искомых параметров. Границы объекта в этом случае скорее всего окажутся размыты, но большая часть объекта (его объём и примерное положение) будет найдена правильно. Регуляризация сглаживанием не отрицает регуляризацию (4.8), а дополняет её в определённых исследователем подобластях исследуемого объёма среды. С учётом сглаживания значений параметров функционал (4.8) примет вид

$$\begin{aligned} & \sum_i^{N^r} \sum_k^{N^t} \left(w_k^i \delta \varepsilon_k^i(\mathbf{p}) \right)^2 + \alpha \sum_j^m \left(p_j^n - \bar{p}_j \right)^2 + \\ & + \sum_j^m \gamma_j \sum_{l \in M_j} (p_j - p_l)^2 \rightarrow \min_{\mathbf{p}}, \end{aligned} \quad (4.10)$$

где M_j – это множество номеров параметров, значения которых сглаживаются со значением j -го параметра. СЛАУ (4.9) принимает вид

$$(\mathbf{A} + \alpha \mathbf{I} + \mathbf{G}) \Delta \mathbf{p} = \mathbf{b} - \alpha(\mathbf{p}^n - \bar{\mathbf{p}}), \quad (4.11)$$

где элементы матрицы \mathbf{G} определяются формулами:

$$G_{ij} = -(\gamma_i + \gamma_j), \quad G_{ii} = \dim(M_i) \gamma_i + \sum_{j \in M_i} \gamma_j, \quad (4.12)$$

где $\dim(M_i)$ – число элементов множества M_i .

ТЕСТИРОВАНИЕ ПРОЦЕДУР

Любую программу перед её практическим применением необходимо протестировать. Для тестирования программ решения обратных задач вместо реальных практических измеренных данных используют

так называемые *синтетические* данные. Синтетическими данными называют теоретические аналоги результатов практических измерений, рассчитанные с помощью программы решения соответствующей прямой задачи при некотором заданном наборе известных значений параметров, являющихся искомыми в обратной задаче.

Так как решение обратной задачи неизбежно включает в себя решение соответствующей прямой задачи, например для вычисления значения функционала или значений производных по параметрам, то **первым этапом** отладки обратной задачи является отладка соответствующей прямой задачи. Для отладки прямой задачи пользуются упрощёнными моделями, имеющими аналитическое решение или (что существенно надёжнее) известными решениями сложных моделей.

Вторым этапом является проверка «на нулевой функционал». Она заключается в том, что для обратной задачи в качестве начальных значений искомым параметров задаются «истинные» значения, т. е. значения параметров, использовавшиеся для генерации синтетических данных. Очевидно, что при «истинных» начальных значениях параметров функционал обратной задачи (без регуляризирующих добавок) должен быть равен нулю. Эта проверка всегда полезна и контролирует *правильность запуска* прямой задачи из обратной.

Третьим этапом является проверка алгоритма на простой обратной задаче. Эта проверка связана с самим алгоритмом решения обратной задачи и для каждого метода является своей. Для метода Гаусса–Ньютона отличным тестом является задача с одним параметром и одним приёмником. Если в качестве начального значения параметра взять $p^0 = p - \delta p$, где p – это истинное значение этого параметра, а δp – приращение параметра, используемое для численного вычисления производной (4.7), то СЛАУ (4.5) будет иметь вид

$$\left(\frac{\delta \varepsilon(p) - \delta \varepsilon(p^0)}{\delta p} \right)^2 \Delta p = \frac{\delta \varepsilon(p) - \delta \varepsilon(p^0)}{\delta p} (\delta \varepsilon(p) - \delta \varepsilon(p^0)).$$

Очевидно, что решением данного СЛАУ будет $\Delta p = \delta p$, причём независимо (!) от δp (по сути независимо от качества вычисления производной). Чаще всего непрохождение этого теста для метода Гаусса–

Ньютона является признаком ошибки в вычислении разности $\delta\epsilon(p) - \delta\epsilon(p^0)$ или ошибки в сборке СЛАУ.

Последним (на что можно только надеяться) **этапом** проверки алгоритма решения обратной задачи являются тесты с множеством параметров, но упрощёнными прямыми задачами. Упрощённые прямые задачи – это такие задачи, результаты для которых (для любых значений параметров) могут быть легко вычислены аналитически. В крайнем случае упрощённой прямой задачей может являться просто вычисление алгебраического выражения. Такое возможно, если алгоритм решения обратной задачи является достаточно абстрактным.

5. ПАРАМЕТРИЗАЦИЯ В ОБРАТНЫХ ЗАДАЧАХ

То, какие величины будут выбраны в качестве искоемых параметров, во многом определяет, с какой точностью будет решена обратная задача и будет ли она решена вообще. Рассмотрим, к примеру, всё ту же обратную задачу электроразведки. Её решением является распределение удельной электрической проводимости в изучаемом объёме среды. Естественным желанием здесь является разбиение данного объёма на непересекающиеся подобласти (ячейки), покрывающие данный объём, и поиск удельной электрической проводимости этих ячеек. Однако встаёт вопрос о том, какие размеры ячеек нужно взять. Предположим, что исследуемая область (с учётом её возможного расширения из-за влияния соседних областей (рис. 2)) имеет размеры 10×10 км по латерали и 3 км в глубину.

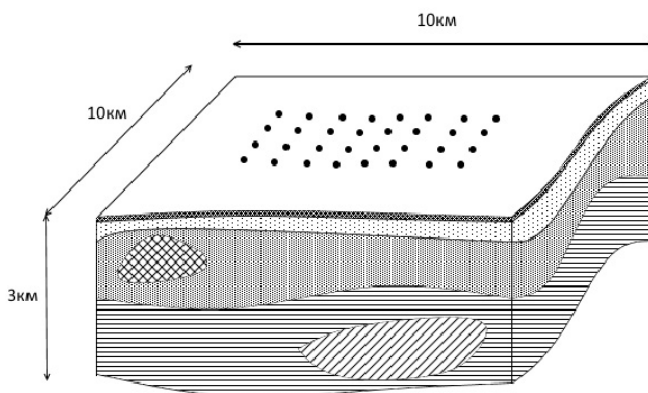


Рис. 2. Пример среды для задачи электроразведки

Учтём, что приповерхностные породы имеют слоистую структуру, в которой толщины слоёв меняются от 10 до 300 м. Если выбрать размер ячейки $500 \times 500 \times 10$ м, то в указанном объёме будет 120 000 ячеек и столько же параметров удельной электрической проводимости. Для описанного выше метода решения обратных задач (Гаусса–Ньютона) на каждой итерации для вычисления производных по параметрам потребуется решать столько же прямых трёхмерных задач. Для этого потребуется несколько десятков очень мощных суперкомпьютеров и очень много времени, что очевидно не доступно рядовому исследователю. К тому же при таком огромном числе параметров огромной будет и область эквивалентности решений. То есть решение не только будет не единственным (что можно преодолеть регуляризацией), но и получаемая картина будет напоминать шахматную доску, где белые и чёрные клетки означают области с повышенной и пониженной удельной электрической проводимостью. Выделить из такой картины какие-либо структуры невозможно. Сглаживание же попросту убьёт смысл использования мелких (10 м по высоте) ячеек. Для того чтобы сделать задачу посильной хотя бы десятку соединённых в сеть персональных ЭВМ, число параметров должно быть не больше нескольких тысяч. Этого можно достичь, используя дополнительную информацию о геометрии слоёв, которую можно получить с помощью сейсморазведки (можно обойтись и без сейсморазведки, но с ней намного лучше). При этом высоты ячеек будут соответствовать толщинам слоёв, а сглаживание осуществляться только по латерали. Сглаживание по вертикали здесь сыграет отрицательную роль, ведь сглаживаться будут характеристики заведомо разных пород (принадлежащих разным геологическим слоям). Пример решения задачи в такой постановке показан на рис. 3 [6].

Решение такой задачи может занять день или даже несколько дней. Но его всё ещё можно ускорить. Для этого можно поступить следующим образом. Учитывая, что среда в целом слоистая и лишь местами содержит «аномальные» включения (как правило, являющиеся целевыми объектами поиска), можно попытаться восстановить близкую к истинной удельную электрическую проводимость этих слоёв, с достаточной для дальнейшего выделения аномальных объектов точностью. Оценивая невязки в приёмниках поля между измеренными сигналами и рассчитанными сигналами от слоистой среды, можно как минимум по латерали, а при достаточном опыте ещё и по вертикали оценить положение центров аномалий. После чего в соответствующие места

могут быть помещены так называемые объекты-зародыши (обычно параллелепипеды небольших размеров). Параметрами здесь станут положение, размеры и, конечно, удельная электрическая проницаемость соответствующих объектов. Число параметров при этом может быть «всего» несколько сотен, что снизит область эквивалентности, уменьшит вычислительную сложность метода Гаусса–Ньютона и улучшит сходимость обратной задачи.

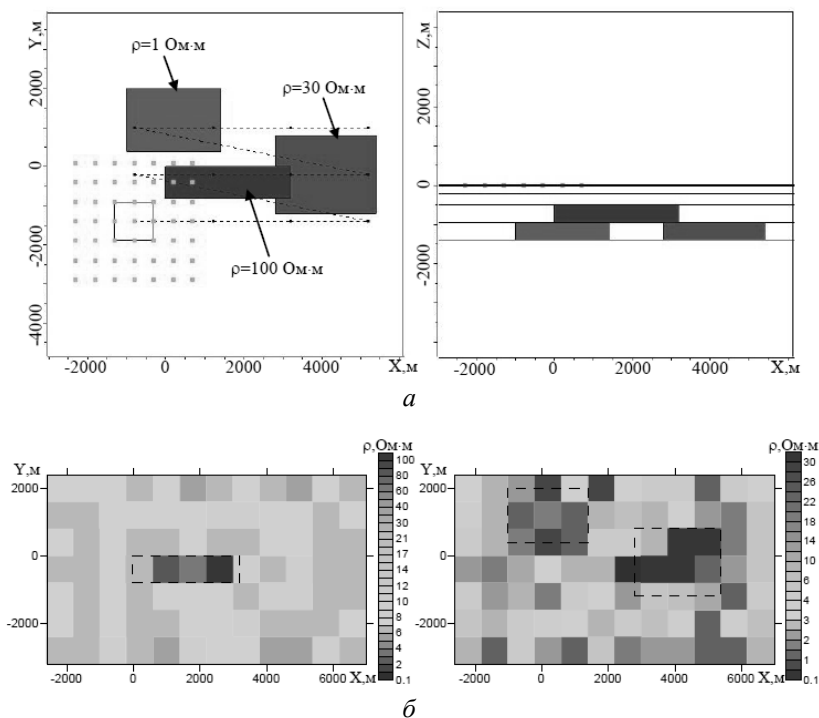


Рис. 3. Пример решения обратной задачи без сглаживания параметров. Истинная модель (а) и результат решения обратной задачи (б, сечение)

Таким образом, учёт всей имеющейся априорной информации об изучаемой среде позволяет более качественно выполнить параметризацию обратной задачи, что приводит к значительному ускорению решения обратной задачи и к тому, что это решение становится более качественным.

ПРИМЕРЫ РЕШЕНИЯ ОБРАТНЫХ ЗАДАЧ МЕТОДОМ ГАУССА–НЬЮТОНА

1. Постановка. Задача электроразведки. Однородное полупространство. Источник поля – заземлённая электрическая линия с постоянным значением силы тока $I = 1$ А с координатами электродов $A(0,0,0)$, $B(100,0,0)$. Измеряется разность потенциалов в приёмных линиях с координатами $M1(200,0,0)$, $N1(300,0,0)$; $M2(500,0,0)$, $N2(600,0,0)$; $M3(1000,0,0)$, $N3(1100,0,0)$. Неизвестным является значение удельной электрической проводимости. В качестве истинного значения выбрать $\sigma = 0.1$ См/м. В качестве начального значения выбрать $\sigma^0 = 0.01$ См/м.

Решение. Потенциал электрического поля V , создаваемый электрической линией AB с постоянным током, расположенной на поверхности земли, в однородном полупространстве складывается из потенциалов, создаваемых её электродами $V_{AB} = V_B(r) + V_A(r)$. Положим, что ток в линии течёт от электрода A к электроду B . Для электрода, по которому ток втекает в среду, $V(r) = \frac{I}{2\pi r \sigma}$, где r – это расстояние

от точки измерения до электрода. Получаем $V_{AB} = \frac{I}{2\pi\sigma} \left(\frac{1}{r_B} - \frac{1}{r_A} \right)$.

Следовательно, разность потенциалов в линии MN будет равна

$$V_{AB}^{MN} = \frac{I}{2\pi\sigma} \left(\left(\frac{1}{r_B^M} - \frac{1}{r_A^M} \right) - \left(\frac{1}{r_B^N} - \frac{1}{r_A^N} \right) \right).$$

Сгенерируем синтетические данные для истинного значения параметра. При $\sigma = 0.1$ с точностью до семи знаков мантиссы имеем:

$$\bar{V}_{AB}^{M_1N_1} = 5.305165\text{E-}03, \quad \bar{V}_{AB}^{M_2N_2} = 2.652582\text{E-}04,$$

$$\bar{V}_{AB}^{M_3N_3} = 3.215251\text{E-}05.$$

Для удобства обозначим $V_{AB}^{M_iN_i} \equiv V_i$, $i = 1, 2, 3$. Очевидно, что при начальном значении параметра $\sigma = \sigma^0$ имеем $V_1^0 = 5.305165\text{E-}02$, $V_2^0 = 2.652582\text{E-}03$, $V_3^0 = 3.215251\text{E-}04$. Так как значения в приёмниках

поля отличаются на более чем на два порядка (сравните V_1 и V_3), то в функционал обратной задачи следует ввести весовые коэффициенты w_i , которые удобно взять равными обратным значениям практических (в данном случае синтетических) значений в приёмниках: $w_i = 1 / \bar{V}_i$. Заметим, что искомый параметр всего один, следовательно, обратная задача корректна и не требует регуляризации, её функционал имеет вид

$$\Phi(\sigma) = \sum_{i=1}^3 (w_i (V_i(\sigma) - \bar{V}_i))^2 \rightarrow \min_{\sigma}.$$

Решим эту задачу методом Гаусса–Ньютона. Так как решение прямой задачи дано в виде простой аналитической формулы, то производные по искомому параметру нет смысла считать численно. Дифференцируя по σ , получаем

$$\frac{\partial V_i(\sigma)}{\partial \sigma} = -\frac{I}{2\pi\sigma^2} \left(\left(\frac{1}{r_B^{M_i}} - \frac{1}{r_A^{M_i}} \right) - \left(\frac{1}{r_B^{N_i}} - \frac{1}{r_A^{N_i}} \right) \right).$$

Положим $I=1$, $w_i = 1 / \bar{V}_i$. Единственный элемент матрицы СЛАУ обратной задачи равен

$$\begin{aligned} a_{11} &= \sum_{i=1}^3 \left(w_i \frac{\partial V_i(\sigma)}{\partial \sigma} \right)^2 \bigg|_{\sigma=\sigma^0} = (1.884956\text{E}+02 \cdot (-5.305165\text{E}+00))^2 + \\ &= (3.769911\text{E}+03 \cdot (-2.652582\text{E}-01))^2 + \\ &+ (3.110177\text{E}+04 \cdot (-3.215251\text{E}-02))^2 = 3.0\text{E}+06. \end{aligned}$$

Вычислим так же единственный элемент правой части:

$$\begin{aligned} b_1 &= -\sum_{i=1}^3 \left(w_i^2 \frac{\partial V_i(\sigma)}{\partial \sigma} (V_i(\sigma) - \bar{V}_i(\sigma)) \right) \bigg|_{\sigma=\sigma^0} = \\ &= -(3.553058\text{E}+04 \cdot (-5.305165\text{E}+00) \cdot 4.774648\text{E}-02) - \end{aligned}$$

$$-(1.421223\text{E}+07 \cdot (-2.652582\text{E}-01) \cdot 2.387324\text{E}-03) - \\ -(9.673199\text{E}+08 \cdot (-3.215251\text{E}-02) \cdot 2.893726\text{E}-04) = 2.7\text{E}+04 .$$

Следовательно, $\Delta\sigma^{01} = b_1 / a_{11} = 9.0\text{E}-03$. И значит, новое приближение $\sigma^1 = \sigma^0 + \Delta\sigma^{01} = 1.9\text{E}-02$.

$$V_1^1 = 2.792192\text{E}-02 , V_2^0 = 1.396096\text{E}-03 , V_3^0 = 1.692238\text{E}-04 .$$

Заметим, что $\Phi(\sigma^1) = 5.452355\text{E}+01$, тогда как $\Phi(\sigma^0) = 2.43\text{E}+02$.

Т а б л и ц а 1

Номер итерации	Значение σ	Значение Φ
0	1.000000E-02	2.430000E+02
1	1.900000E-02	5.452355E+01
2	3.439000E-02	1.091935E+01
3	5.695328E-02	1.713815E+00
4	8.146980E-02	1.551988E-01
5	9.656632E-02	3.793067E-03
6	9.988210E-02	4.180105E-06
7	9.99986E-02	5.797021E-12
8	1.000000E-01	0.000000E+00

Продолжение итерационного процесса даст истинное значение искомой проводимости полупространства. В табл. 1 представлен полный листинг значений искомого параметра и функционала по итерациям.

Отметим следующий важный момент. Даже эта на первый взгляд очень простая задача таит в себе подводные камни. Так, если бы в качестве σ^0 было задано значение 1 См/м, то мы бы получили $\sigma^1 < 0$, что демонстрирует, как плохо может работать линеаризация при случайных значениях параметров. Чтобы избежать указанной проблемы, в функционал обратной задачи можно (несмотря на корректность самой

задачи) добавить регуляризацию (на физичность решения) или использовать релаксацию получаемых приращений. Также отметим то, что вместо аналитического дифференцирования могло использоваться численное (как в реальных задачах), в этом случае разумно (так как линейаризованный функционал может сильно отличаться от исходного нелинейного) получаемые приращения параметров ограничить по модулю, связав их с дельтами по параметрам, использовавшимся для численного дифференцирования.

2. Постановка. Задача гравиразведки. Источником гравитационной аномалии является тело $\Omega = [-100, 100] \times [-200, 200] \times [-200, -100]$ м. Приёмники поля расположены на оси OX $[-1000, 1000]$ м с шагом 10 м. Неизвестным параметром является аномальная плотность тела $\Delta\rho$. В приёмниках меряется z -составляющая гравитационного поля. Для уменьшения сложности обратной задачи считать размеры тела по оси OY известными.

Решение. Гравитационное поле, создаваемое аномальной плотностью искомого тела, в области приёмников описывается уравнением

$$\Delta\mathbf{g} = \frac{\gamma}{4\pi} \int \frac{\Delta\rho}{\|\mathbf{r}\|^3} (\tilde{x}\mathbf{e}_x + \tilde{y}\mathbf{e}_y + \tilde{z}\mathbf{e}_z) d\Omega,$$

где γ – коэффициент, связанный со значением напряжённости гравитационного поля; \mathbf{r} – радиус-вектор, направленный из точки тела Ω в точку приёмника; $\tilde{x}, \tilde{y}, \tilde{z}$ – центрированные (относительно точки тела Ω) координаты приёмника; $\mathbf{e}_x, \mathbf{e}_y, \mathbf{e}_z$ – орты системы координат.

Ограничимся глубиной поиска в 500 м и разобьём область под приёмниками на ячейки. В силу сделанного упрощения область поиска вдоль оси OY известна и будет разбиваться только на одну ячейку. Для разбиения области вдоль оси OX выберем шаг 100 м, а вдоль оси OZ – 50 м. Итак, получаем разбиение области $[-1000, 1000] \times [-200, 200] \times [-500, 0]$ на $20 \times 1 \times 10$ ячеек, в которых будем искать значения $\Delta\rho$, полагая их постоянными внутри ячеек.

Так как в приёмниках меряется только z -составляющая гравитационного поля, а значение $\Delta\rho$ постоянно на ячейке, то поле в i -м приёмнике (отклик) будет равно

$$\Delta g_z^i = \gamma \sum_j \frac{\Delta\rho_j}{4\pi} \int \frac{\tilde{z}_i}{\|\mathbf{r}_i\|^3} d\Omega_j, \quad \text{где } \Omega_j -$$

j -я ячейка. Обратим внимание на то, что поле в приёмниках Δg_z^i зависит

от искоемых параметров $\Delta\rho_j$ линейно. То есть, в отличие от рассмотренной выше обратной задачи геоэлектрики, обратная задача гравиразведки является линейной и для её решения достаточно одной итерации метода Гаусса–Ньютона, так как производные откликов по параметрам

постоянны и равны $\frac{\partial\Delta g_z^i}{\partial\Delta\rho_j} = \frac{\gamma}{4\pi} \int \frac{\tilde{z}_i}{\|r_i\|^3} d\Omega_j$. Элементы матрицы и правой части СЛАУ обратной задачи будут равны:

$$A_{qs} = \sum_i^N (w^i)^2 \frac{\partial\Delta g_z^i}{\partial\Delta\rho_q} \frac{\partial\Delta g_z^i}{\partial\Delta\rho_s}, \quad b_q = -\sum_i^N (w^i)^2 (\Delta g_z^i - \Delta\bar{g}_z^i) \frac{\partial\Delta g_z^i}{\partial\Delta\rho_q},$$

где N – число приёмников; $\Delta\bar{g}_z^i$ – практические данные; q, s – номера параметров (в данном случае совпадают с номерами ячеек), веса w^i можно взять равными $w^i = 1.0$).

Для вычисления интеграла $\int \frac{\tilde{z}_i}{\|r_i\|^3} d\Omega_j$ можно использовать любой численный метод. Хорошим решением являются квадратурные формулы Гаусса с высоким порядком аппроксимации.

Перед началом решения обратной задачи сгенерируем синтетические данные. Для упрощения вычислений положим $\gamma = 1$. Положим, что искомое распределение $\Delta\rho$ имеет следующий вид (рис. 4).

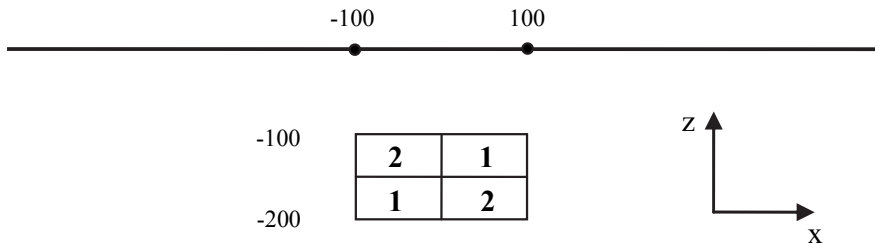


Рис. 4. Истинное распределение $\Delta\rho$

Соответствующий график сигнала в приёмниках представлен на рис. 5. Для интегрирования использовалась схема Гаусса по пяти точкам.

Сначала решим обратную задачу без регуляризации, т. е. с $\alpha = 0$. Полученное решение представлено на рис. 6. Среднее квадратов отклонений $\text{Err} = \sum_i^N (\Delta g_z^i - \Delta \bar{g}_z^i)^2 / N$ получается равным 5.87202e-007.

То есть сигналы в приёмниках от истинной модели (рис. 4) и подобранной модели (рис. 6) неразличимы. Проявляется неединственность решения обратной задачи.

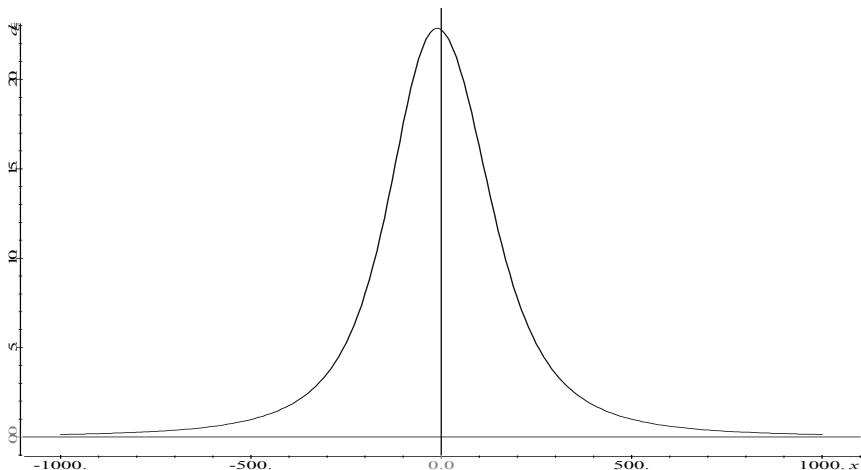


Рис. 5. Синтетические данные $\Delta \bar{g}_z$

0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
0.5	1.0	0.5	0.7	0.3	0.6	0.1	0.4	-0.2	2.4	-0.1	0.1	-1.5	-0.2	-1.2	-0.3	-1.4	-0.3	-1.0	0.0
-7.0	-15.0	-2.2	-3.7	-4.8	-4.9	0.4	-3.7	6.1	-2.6	17.7	-5.3	20.4	7.4	7.8	8.2	20.0	8.7	4.2	6.1
27.7	49.0	-49.7	-47.7	-8.1	-22.4	-31.8	-19.5	-55.3	5.0	-40.6	89.1	-51.4	-11.0	48.6	-21.7	-73.3	-28.1	52.0	-52.2
-35.2	-50.1	74.6	-6.3	17.6	114.6	48.5	82.5	32.9	34.3	26.5	-100.8	74.1	53.7	-31.9	27.9	11.3	-35.8	-52.1	172.3
-45.3	-82.6	-119.9	193.6	-71.3	-143.9	68.2	-139.5	128.2	-135.2	85.7	-59.1	86.9	3.3	-123.5	20.5	55.5	225.3	-282.2	65.6
228.8	58.1	168.8	-25.6	18.5	20.3	-69.2	3.8	52.5	-77.8	98.7	-104.9	-189.8	72.5	-344.5	172.7	-11.5	-50.7	-137.5	-369.9
-182.4	-218.1	193.0	15.8	-200.7	37.0	-28.3	-93.9	41.8	-111.1	121.4	-301.4	191.7	231.3	-109.4	10.7	1.2	-138.2	544.7	-11.2
-22.1	85.3	11.1	-62.6	180.4	8.5	2.4	11.9	36.0	211.1	-50.4	119.8	-127.6	-157.4	187.3	-36.8	15.6	137.8	-65.6	115.0

Рис. 6. Найденное распределение $\Delta \bar{g}_z$ без регуляризации

Начиная с $\alpha = 1e-20$ будем увеличивать α и следить за значением среднего квадратов отклонений (табл. 2).

Таблица 2

Значение α	Значение Err
1e-20	5.87202e-007
1e-19	5.87202e-007
1e-18	5.87202e-007
1e-17	5.87202e-007
1e-16	5.87202e-007
1e-15	5.22685e-007
1e-14	6.4108e-007
1e-13	5.1996e-007
1e-12	5.21088e-007
1e-11	5.45525e-007
1e-10	5.90438e-007
1e-09	6.11669e-007
1e-08	6.25488e-007
1e-07	7.16555e-007

Здесь следует остановиться, так как наблюдается устойчивое возрастание отклонений откликов в приёмниках. В качестве решения можно выбрать распределение параметров при $\alpha = 1e-10 - 1e-8$. Из рис. 7 и 8 видно, что верхняя часть аномального тела (и соответствующий слой вмещающей среды) восстанавливаются почти точно. Неплохо восстанавливается и нижняя часть объекта с соответствующим слоем вмещающей среды.

0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.1	2.1	1.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
0.0	0.0	0.0	0.0	-0.1	0.0	-0.2	0.2	-0.8	0.3	0.3	1.9	-0.2	0.4	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
0.0	0.0	0.0	-0.1	0.1	-0.2	0.3	-0.8	0.9	0.6	-0.2	0.4	-0.6	0.2	0.0	0.1	0.0	0.0	0.0	0.0	0.0	0.0
0.0	0.0	0.0	0.0	0.0	-0.1	0.1	-0.3	0.6	0.4	-0.1	0.1	-0.3	0.1	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
0.0	0.0	0.0	0.0	0.0	0.0	0.1	-0.1	0.3	0.2	0.0	0.0	-0.1	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.2	0.2	0.0	0.0	-0.1	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.1	0.1	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.1	0.1	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0

Рис. 7. Найденное распределение $\Delta\rho$ с $\alpha = 1e-10$

7. Два слоя: $\sigma_1 = 0.01$ См/м, $\sigma_2 = 0.1$ См/м, $h_2 = \infty$. Приёмник 2. Источник 2. Определить значение h_1 . Добавить шум, равный 10 % от значения измерения.

8. Два слоя: $\sigma_1 = 0.01$ См/м, $\sigma_2 = 0.1$ См/м, $h_2 = \infty$. Приёмник 3. Источник 2. Определить значение h_1 . Добавить шум, равный 10 % от значения измерения.

9. Два слоя: $h_1 = 20$ м, $h_2 = \infty$. Приёмники 1–3. Источник 2. Определить значения σ_1 , σ_2 .

10. Два слоя: $h_1 = 20$ м, $h_2 = \infty$. Приёмники 1–2. Источник 2. Определить значения σ_1 , σ_2 . Добавить шум, равный 10 % от значения измерения.

11. Два слоя: $h_1 = 20$ м, $h_2 = \infty$. Приёмники 2–3. Источник 2. Определить значения σ_1 , σ_2 . Добавить шум, равный 10 % от значения измерения.

12. Два слоя: $\sigma_1 = 0.01$ См/м, $h_2 = \infty$. Приёмники 1–3. Источник 2. Определить значения h_1 , σ_2 .

13. Два слоя: $\sigma_1 = 0.01$ См/м, $h_2 = \infty$. Приёмники 1–3. Источник 2. Определить значения h_1 , σ_2 . Добавить шум, равный 1 %, 2 %, 5 % от значения измерения.

14. Задача гравirazведки. Данные, как в примере. Использовать сглаживание значений параметров в ячейках.

6. ГЕНЕТИЧЕСКИЙ АЛГОРИТМ ДЛЯ РЕШЕНИЯ ОБРАТНЫХ ЗАДАЧ

Часть задач оптимизации относится к классу комбинаторных и они, в большинстве случаев, имеют не одно, а множество решений различного качества. Существует множество алгоритмов для решения таких задач. Ядром всех комбинаторных алгоритмов являются операции полного или сокращённого перебора подмножеств решений. Стратегия комбинаторного перебора в различных алгоритмах реализуется по-разному. Для поиска лучшего решения, как правило, осуществляются направленный, случайный и комбинированный переборы всевозможных значений параметров задачи. В этой связи разрабатывается большое число точных, переборных и эвристических методов решения этих задач. До последнего времени не существовало эффективного механизма поиска решений на множестве альтернатив, что затрудняло получение качественных результатов за приемлемое время [7].

В конце 1960-х годов американский исследователь Джон Холланд в качестве принципов комбинаторного перебора вариантов решения оптимизационных задач предложил использовать методы и модели механизма развития органического мира на Земле. Поскольку основные законы эволюции живых организмов были исследованы генетикой, то и предложенный механизм получил название «генетические алгоритмы».

В природе строение (*фенотип*) любого живого организма определяется набором его генов (*генотипом*), который он получает от своих родителей (или родителя при бесполом размножении). Если родителей два, то от каждого родителя он наследует по половине генов. Поэтому при половом размножении новый организм не

может быть точной копией одного из родителей. Последнее возможно лишь при бесполом размножении. И в обоих случаях любые гены могут быть переданы новому организму с изменениями (*мутациями*). Мутации носят случайный характер, как и то, какие именно гены достанутся новой особи от её родителей при половом размножении. Вместе с генами новому организму передаются особенности фенотипа родителей, что определяет его выживаемость и шансы на оставление собственного потомства. Так образуются биологические виды, состоящие из особей со схожим фенотипом, обеспечивающим их выживание. Естественный отбор здесь играет роль функции, аргументом которой является фенотип особи, а значением – вероятность выживаемости (*приспособленность*). Эволюция же, по сути, является природным механизмом максимизации приспособленности живых организмов.

Перенос этих природных сущностей и механизмов на решение обратных задач приводит к следующей концепции:

1) искомые (оптимальные) параметры задают генотип с фенотипом, имеющим максимальную приспособленность (минимум функционала обратной задачи);

2) процесс решения обратной задачи представляется в виде эволюции популяции особей, где каждое новое поколение имеет большую (среднюю) приспособленность, т. е. меньшее значение функционала обратной задачи;

3) вероятность для конкретной особи того, что её гены будут переданы особям следующего поколения, зависит от её приспособленности, что обеспечивает закрепление «хороших» генов в течение эволюционного процесса;

4) при создании нового поколения с определённой вероятностью происходят мутации генов (изменения значений параметров), что потенциально обеспечивает нахождение минимально возможного при данной параметризации значения функционала обратной задачи.

В простейшем случае этот эволюционный процесс можно представить в виде следующей последовательности действий (рис. 9), получившей название *простой генетический алгоритм* (ПГА).

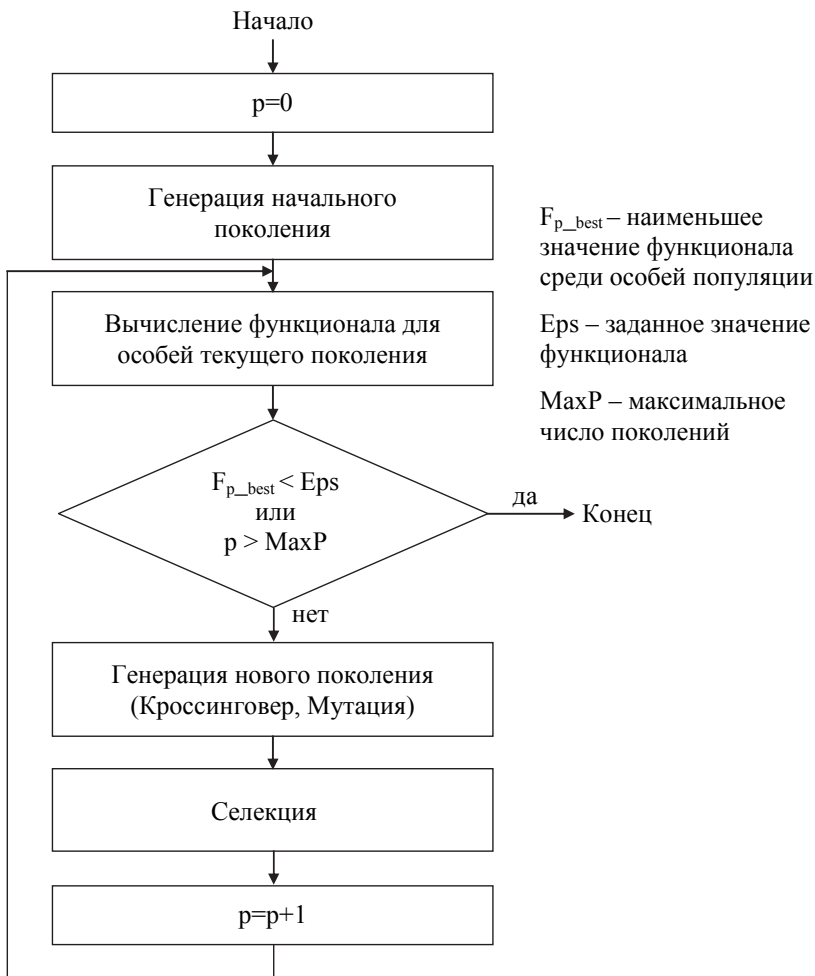


Рис. 9. Простой генетический алгоритм

Рассмотрим подробнее основные элементы ПГА.

КРОССИНГОВЕР

Кроссинговером называется рекомбинация генетической информации (обмен генетической информацией). В природе такая рекомбинация происходит между парой хромосом. При этом рекомбинация происходит, как правило, между *гомологичными* участками хромосом. Рассмотрим пример, изображённый на рис. 10.

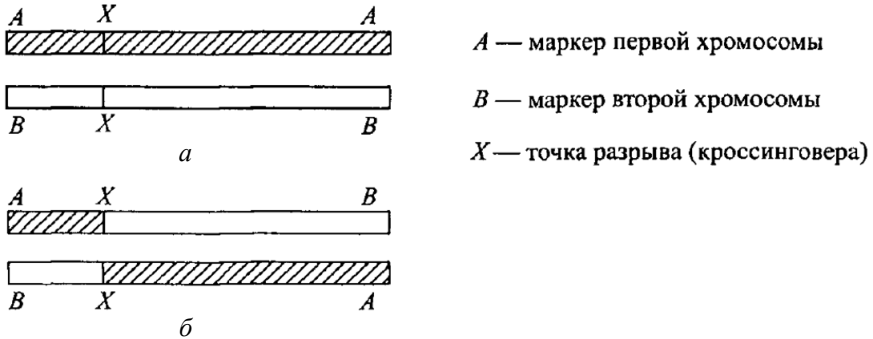


Рис. 10. Кроссинговер с одной точкой разрыва:

a — до кроссинговера; *б* — после кроссинговера

Здесь две исходные хромосомы (*a*) обмениваются своими участками так, что получаются две новые хромосомы (*б*), имеющие генетическую информацию от обеих родительских хромосом. Этот пример иллюстрирует кроссинговер с одной точкой разрыва. В принципе их может быть больше (рис. 11).

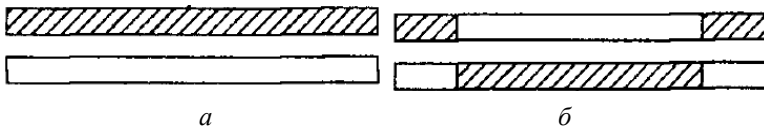


Рис. 11. Кроссинговер с двумя точками разрыва:

a — до кроссинговера; *б* — после кроссинговера

Более того, в абстрактной математической задаче в кроссинговере может участвовать произвольное число хромосом (родительских особей). Точки разрыва, как правило, определяются случайным образом,

как и то, из какой родительской хромосомы будет взята генетическая информация.

Для того чтобы закрепить присутствие в поколениях генотипов с высокой приспособленностью, в кроссинговере при генерации особей нового поколения должны в качестве родителей участвовать особи с высокой приспособленностью из предыдущего поколения. При этом простой выбор особей с наилучшими значениями приспособленности может привести к быстрой сходимости алгоритма к одному генотипу, и даже наличие мутаций может не помочь решить задачу за заданное максимальное число поколений. Поэтому придерживаются правила, согласно которому вероятность быть отобранным для кроссинговера (оставить потомство) у особи связана со значением её функции приспособленности. Таким образом, даже особи с малой приспособленностью имеют ненулевую вероятность передать свой генотип новому поколению, что поддерживает разнообразие популяции и положительно влияет на скорость решения задачи.

МУТАЦИИ

В биологии под *мутациями* понимают как изменение самих генов, так и изменение их последовательности или даже числа в хромосоме. Для математической задачи оптимизации, как правило, разумными являются только изменения самих генов (значений параметров). Мутации не направлены и происходят с определённой вероятностью, внося таким образом разнообразие в популяцию. При этом если вероятность мутаций будет чрезмерно высока, то решение задачи (достижение оптимума) может стать проблематичным, так как мутация может разрушить генотипы с высокой приспособленностью. Полное отсутствие мутаций также недопустимо, так как в этом случае найти оптимум может стать в принципе невозможно, если, например, гены с «правильными» значениями вообще не присутствуют в исходной популяции.

СЕЛЕКЦИЯ

Селекция представляет собой форму искусственного отбора, при которой предпочтение отдаётся особям с определёнными фенотипами. В ПГА селекция используется для отбора особей нового поколения.

Обычно размеры популяции ограничиваются, так как для каждой особи необходимо вычислять значение функции приспособленности. Но в процессе генерации особей нового поколения можно создать их больше, чем задано, и убрать «лишних» особей с помощью селекции. Более того, при решении абстрактной математической задачи не обязательно «убивать» всех особей предыдущего поколения. Часть особей предыдущего поколения с высокими значениями функции приспособленности может войти в новое поколение без изменений. В общем, возможно применение любой эвристики.

НЕЧЁТКИЕ ГЕНЕТИЧЕСКИЕ АЛГОРИТМЫ РЕШЕНИЯ ЗАДАЧ ОПТИМИЗАЦИИ И ПРОЕКТИРОВАНИЯ

При решении многих задач оптимизации, проектирования и принятия решений часто приходится встречаться с таким понятием, как неопределённость в отношении какого-либо параметра. Для решения данной проблемы был предложен подход, основанный на создании гибридных систем и алгоритмов на основе интеграции нечётких математических моделей и методов эволюционного поиска. Одним из таких методов являются нечёткие эволюционные и адаптивные алгоритмы.

Как известно, генетические алгоритмы представляют собой адаптивный поисковый метод, который основан на селекции лучших элементов в популяции, подобно эволюционной теории Ч. Дарвина. Генетические алгоритмы эффективно используют информацию, накопленную в процессе эволюции.

В настоящее время используется новая парадигма решения оптимизационных задач на основе генетических алгоритмов и их различных модификаций. Они осуществляют поиск баланса между эффективностью и качеством решений за счёт «выживания» сильнейших альтернативных решений в неопределённых и нечётких условиях.

Для работы генетических алгоритмов выбирают множество натуральных параметров оптимизационной проблемы и кодируют их в последовательность конечной длины в некотором алфавите. Они работают до тех пор, пока не будет выполнено заданное число генераций (итераций алгоритма) или на некоторой генерации не будет получено решение определённого качества, или, когда не найден локальный

оптимум, происходит возникновение преждевременной сходимости и алгоритм не может найти выход из этого состояния. Генетический алгоритм, как правило, анализируют различные области пространства решений одновременно и поэтому они более приспособлены к нахождению новых областей с лучшими значениями целевой функции.

Мягкие вычисления – сложная компьютерная методология, основанная на нечёткой логике, генетических вычислениях, нейрокомпьютинге и вероятностных вычислениях. Составные части не конкурируют, но создают эффект взаимного усиления для достижения робастности, низкой цены решения, повышения эффективности приложений. Можно выделить четыре наиболее крупные составные части данного направления:

- нечёткая логика (приближённые вычисления, грануляция информации, вычисления на словах);
- нейрокомпьютинг (обучение, адаптация, классификация, системное моделирование и идентификация);
- генетические вычисления (синтез, настройка и оптимизация с помощью систематизированного случайного поиска и эволюции);
- вероятностные вычисления (управление неопределённостью, сети доверия, хаотические системы, предсказание).

Математический аппарат теории нечётких систем используется в данном случае для кодирования, подбора оптимальных параметров генетических алгоритмов, значений вероятности генетических операторов, выбора функции пригодности, создания нечётких генетических операторов. Есть два преимущества нечёткого кодирования:

- 1) кодовые последовательности могут быть неоднородными и ориентироваться на отдельные многообещающие области поиска, что позволяет сократить область поиска и соответственно вычислительные затраты; в закодированную последовательность может быть неявным образом включена функция пригодности;
- 2) нечёткое кодирование позволяет выполнять так называемое слабое кодирование оптимизируемых структур.

В настоящее время наибольшие успехи в интеграции систем и подходов нечёткой логики и генетических алгоритмов достигнуты в следующих областях:

- 1) применение механизмов эволюционных алгоритмов для решения проблем оптимизации и поиска в условиях нечёткой, неопределённой или недостаточной информации об объекте, параметрах и

критериях решаемой задачи, совместно с использованием систем нечёткой логики;

2) использование нечётких инструментов и методов, основанных на нечёткой логике для моделирования различных компонентов и операторов генетических алгоритмов, а также для адаптации и управления основными параметрами генетического алгоритма для динамической настройки и улучшения его работы.

Как правило, под нечётким алгоритмом (НГА) понимают гибридные структуры, относящиеся ко второй области. Таким образом, нечёткий алгоритм можно определить как алгоритм, сочетающий поисковые возможности генетических алгоритмов и возможности математического аппарата нечёткой логики. Математический аппарат теории нечётких систем используется в данном случае для кодирования, подбора оптимальных параметров генетических алгоритмов, значений вероятности генетических операторов, выбора функции пригодности и критерия останова, создания нечётких генетических операторов.

ПРАКТИЧЕСКОЕ ПРИМЕНЕНИЕ

Основные проблемы при реализации генетических алгоритмов заключаются в следующем. Важно найти оптимальный размер популяции и достаточное число поколений. Наибольшие затруднения вызывают вопросы управления процессом генетического поиска. Эффективность работы генетического алгоритма напрямую зависит от выбора его параметров. Так, при большой вероятности кроссинговера увеличивается вероятность уничтожения решений, имеющих высокие значения функции пригодности. Низкая вероятность кроссинговера может затруднить получение лучших решений и не гарантирует более быстрой конвергенции. Высокая вероятность мутации «хороших» решений увеличивает разнообразие популяции и соответственно время решения. Низкая вероятность мутации может приводить к пропуску некоторых, близких к оптимальным, решений.

Положительной стороной генетических алгоритмов является возможность их эффективного распараллеливания. Нет также больших проблем в сочетании генетических и детерминированных алгоритмов в ситуациях, когда последние могут эффективно находить локальные оптимумы целевой функции.

ПРИМЕР ПРОГРАММЫ ПОДБОРА ПАРАМЕТРОВ С ПОМОЩЬЮ ПГА

```
#include<time.h>
#include<iostream>
#include<fstream>
using namespace std;
ofstream logfile;
int MAXITER=50;    // максимальное число генераций (поколений)
int POPNUM=10000;  // число особей в популяции
int POPNUM2=POPNUM*2; // временное увеличение популяции (до
селекции)
int PARNUM=50;     // число генов
int MaxParent=5;   // максимальное число родителей
int NABC=10; // генами являются натуральные числа от 1 до 10
double FunctionalMin=0.1; // значение функционала для выхода
double MutationProbability=0.2; // вероятность мутаций
int *TrueVector;
int **Population;
int **PopulationTemp;
double *PopFunc;
int *IndBest;
double GetRandomDouble(){return (1.0*rand()/RAND_MAX);}
int GetRandomValue(int RealMax)
{
    int val=(int)(RealMax*GetRandomDouble()+0.5);
    return ((val<RealMax)? val : (RealMax-1));
}
int GetRandomValueUp(int RealMax){return
(int)(RealMax*GetRandomDouble()+0.5);}
double GetFunctional(int *TestVector) // Вычисление функционала
{
    int i;
    double err=0.0;
    for(i=0;i<PARNUM;i++){err+=(TestVector[i]!=TrueVector[i]);}
    return err;
}
// Сортировка особей по значению функционала
void GetIndBest(int n)
```

```

{
    int i,j,k;
    for(i=0;i<n;i++){IndBest[i]=i;}
    for(i=1;i<n;i++)
    {
        k=IndBest[i];
        for(j=i-1;j>=0 && PopFunc[IndBest[j]]>PopFunc[k];j--)
            {IndBest[j+1]=IndBest[j];}
        IndBest[j+1]=k;
    }
}
// Выбор второго родителя
int GetMother() return GetRandomValue(POPNUM);}
void GetNewChild(int *father,int *mother,int *child) // Кроссинговер
{
    int i;
    int IndCross=GetRandomValue(PARNUM);
    for(i=0;i<IndCross;i++){child[i]=father[i];}
    for(i=IndCross;i<PARNUM;i++){child[i]=mother[i];}
}
void Mutation(int *species) // Мутации
{
    double v=GetRandomDouble();
    if(v<MutationProbability)
    {
        int i=GetRandomValue(PARNUM);
        species[i]=GetRandomValueUp(NABC);
    }
}
void CreateNewPopulation() // Генерация особей нового поколения
{
    int i,j,k;
    int nChilds=POPNUM2/MaxParent;
    for(i=0;i<MaxParent;i++)
    {
        for(j=0;j<nChilds;j++)
        {

```

```

        k=GetMother();
        GetNewChild(Population[i],Population[k],PopulationTemp[i*nChilds+j]);
        Mutation(PopulationTemp[i*nChilds+j]);
    }
}

void Selection(double &BestFunctional) // Селекция
{
    int i,j;
    for(i=0;i<POPNUM2;i++){PopFunc[i]=GetFunctional(PopulationTemp[i]);}
    GetIndBest(POPNUM2);
    BestFunctional=1e+30;
    for(i=0;i<POPNUM;i++)
    {
        for(j=0;j<PARNUM;j++){Population[i][j]=PopulationTemp[IndBest[i]][j];}
    }
    if(PopFunc[IndBest[i]]<BestFunctional){BestFunctional=PopFunc[IndBest[i]];}
}

int main() // Основная функция программы
{
    int i,j,iter;
    double BestFunctional;
    ifstream inf;
    ofstream ofp;
    clock_t t1,t2;
    bool show_on_display;
    bool write_to_log;
    logfile.open("Genesis.log");
    show_on_display=1;
    write_to_log=0;
    TrueVector=new int[PARNUM];
    Population=new int *[POPNUM];
    for(i=0;i<POPNUM;i++){Population[i]=new int [PARNUM];}
    PopulationTemp=new int *[POPNUM2];
    for(i=0;i<POPNUM2;i++){PopulationTemp[i]=new int [PARNUM];}
    PopFunc=new double [POPNUM2];
    IndBest=new int [POPNUM2];

```

```

t1=clock();
srand((unsigned)time(NULL));
// Генерируем истинные параметры для теста
for(i=0;i<PARNUM/10;i++)
{
    for(j=0;j<10;j++){TrueVector[10*i+j]=j+1;}
}
// Создаем первое поколение
iter=0;
for(i=0;i<POPNUM;i++)
{
    for(j=0;j<PARNUM;j++)

PopulationTemp[i][j]=GetRandomValueUp(NABC);
    }
    PopFunc[i]=GetFunctional(PopulationTemp[i]);
}
GetIndBest(POPNUM);
BestFunctional=1e+30;
for(i=0;i<POPNUM;i++)
{
for(j=0;j<PARNUM;j++){Population[i][j]=PopulationTemp[IndBest[i]][j];}
if(PopFunc[IndBest[i]]<BestFunctional){BestFunctional=PopFunc[IndBest[i]];}
}
if(show_on_display){cout<<"iter=   "<<iter<<"   BestFunctional=
"<<BestFunctional<<"\n";}
// Цикл по поколениям
for(iter=1;iter<=MAXITER;iter++)
{
    CreateNewPopulation();
    Selection(BestFunctional);
    if(show_on_display){cout<<"iter=   "<<iter<<"   BestFunctional=
BestFunctional= "<<BestFunctional<<"\n";}
    if(write_to_log){logfile<<"iter=   "<<iter<<"   BestFunctional=
BestFunctional= "<<BestFunctional<<"\n";}
    if(BestFunctional<FunctionalMin){break;}
}
t2=clock();

```

```

        logfile<<"\n"<<"time= "<<(int)(t2-t1)<<"ms"<<" iter= "<<iter<<"
BestFunctional= "<<BestFunctional<<"\n';
        ofp.open("best");
        for(i=0;i<PARNUM;i++){ofp<<Population[0][i]<<"\n";}
        ofp.close();
        ofp.clear();
        logfile.close();
        logfile.clear();
        return 0;
}

```

ЗАДАЧИ

Реализовать ПГА. Генотип состоит из N чисел. Фенотип состоит из M чисел. Числа фенотипа вычисляются регулярным образом из чисел генотипа.

1. Числа генотипа и фенотипа являются цифрами 0–9. Значение функционала – сумма модулей разностей цифр фенотипа. Мутация меняет цифру на единицу в плюс или минус, если это возможно.

2. Числа генотипа и фенотипа являются цифрами 0–9. Значение функционала – сумма модулей разностей цифр фенотипа. Мутация меняет цифру на случайную.

3. Числа генотипа и фенотипа являются вещественными значениями. Значение функционала – корень из суммы квадратов чисел фенотипа. Мутация меняет число не более чем на 10 % в плюс или минус.

4. Числа генотипа и фенотипа являются вещественными значениями. Значение функционала – корень из суммы квадратов чисел фенотипа. Мутация меняет число не более чем на 100 % в плюс или минус. Кроссинговер выполнять только среди лучших особей.

5. С помощью ПГА восстановить коэффициенты полинома десятой степени по его значениям в заданном наборе точек. Значения зашумить на 1, 2, 5 %.

7. МАШИННОЕ ОБУЧЕНИЕ

Термин «машинное обучение» впервые был использован А. Тьюрингом, и в настоящее время включает довольно широкий спектр подходов и методов анализа данных. Характерной особенностью этих методов является зависимость их эффективности от объема анализируемых данных (*dataset*), часть которых может использоваться для так называемого обучения.

Методы машинного обучения можно разделить на две группы:

- обучение с учителем (*supervised learning*),
- обучение без учителя (*unsupervised learning*).

Обучение с учителем включает задачи *классификации* и задачи *регрессии*. Здесь входные данные представляются в виде множества объектов X и множества меток Y . Задача состоит в поиске правила отображения $X \rightarrow Y$. Объекты множества X это обычно некоторые наборы чисел $\{x_1, x_2, \dots, x_n\}$, которые называются факторами. Если множество Y дискретное, то это будет задача классификации, а если Y представляет собой непрерывные величины – то задача регрессии. Обучение без учителя анализирует множество X без каких-либо меток. Сюда входят задачи *кластеризации* и задачи *понижения размерности* [8]. Далее кратко рассмотрим эти методы.

ОБУЧЕНИЕ БЕЗ УЧИТЕЛЯ

Кластеризация (*clustering*) данных позволяет разбить исходное множество данных на группы. Одним из простых и эффективных алгоритмов кластеризации является *алгоритм k-средних* (максимизация математического ожидания). Это итерационный алгоритм, в котором на каждой итерации уточняется положение центров кластеров (k – количество центров):

Шаг 1: выбор начального положения центров кластеров;

Цикл до достижения сходимости:

Е-шаг: приписываем точки к ближайшим центрам кластеров;

М-шаг: задаем новые центры кластеров в соответствии со средними значениями.

Для простоты рассмотрим этот алгоритм на множестве данных с двумя факторами, которые можно изобразить точками на плоскости. На рис. 12 показан ход алгоритма k -средних.

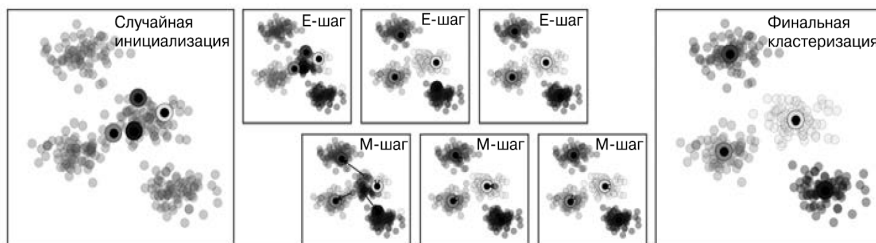


Рис. 12. Иллюстрация работы алгоритма k -средних

Интуитивно понятно, что данный алгоритм будет эффективен в случаях, когда границы между кластерами – линейные функции. Для более сложных данных метод может не дать осмысленного результата (рис. 13).

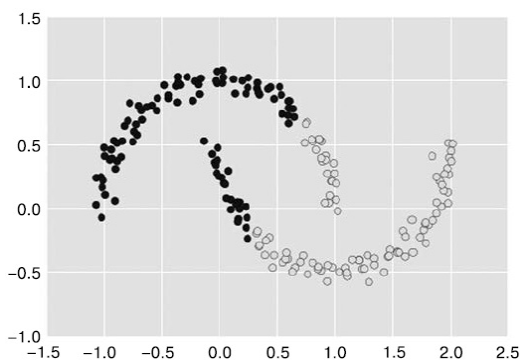


Рис. 13. Неудовлетворительный результат алгоритма k -средних в случае нелинейных границ

Методы **понижения размерности** (dimensionality reduction) предназначены для поиска более сжатых представлений данных. Это мощный инструмент в части предобработки информации и её визуализации. Одним из таких методов является *метод главных компонент* (principal component analysis, PCA). PCA позволяет решить следующие задачи:

- аппроксимировать данные линейными многообразиями меньшей размерности: найти линейное многообразие заданной размерности $k \leq n$ (количество факторов), сумма квадратов расстояний до которого минимальна;
- найти подпространство заданной размерности, в ортогональной проекции на которое среднеквадратичное расстояние между каждой парой точек максимально.

Отличие PCA и МНК проиллюстрировано на рис. 14, где показаны два варианта аппроксимирующей прямой по МНК (рис. 14, *а* и *б*) и по PCA (рис. 14, *в*) [9]. Исходные данные имеют размерность 2, а результатом PCA является одномерное пространство, в котором сохраняется наибольшее количество информации об исходном множестве. Если не понижать размерность пространства, то поиск главных компонент по сути будет являться аффинным преобразованием. Оси преобразовываются таким образом, чтобы разброс проекции данных на главные оси был максимальным, при этом длина оси будет соответствовать её «важности» для описания данных. Пример такого аффинного преобразования показан на рис. 15. Проекции данных на главные оси называются главными компонентами.

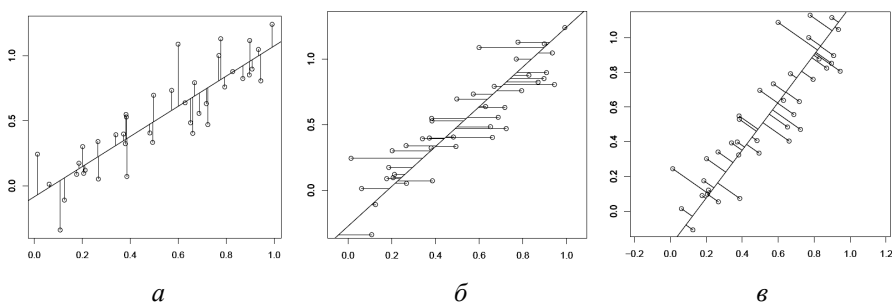


Рис. 14. МНК-аппроксимация:

а – для $y(x)$; *б* – $x(y)$; *в* – результат метода главных компонент

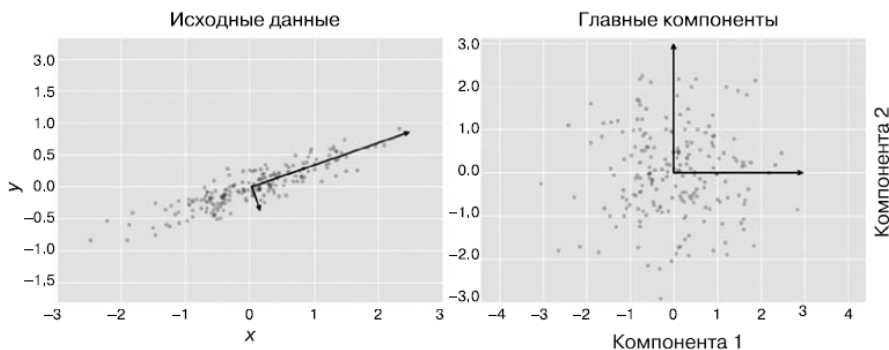


Рис. 15. Переход к главным осям

В случае применения РСА для понижения размерности наименьшие главные компоненты обнуляются, в результате чего исходные данные отображаются на пространство меньшей размерности, сохраняя максимальную дисперсию. Рассмотрим пример использования РСА на популярном для тестирования методов машинного обучения наборе данных «Ирисы Фишера». В нем данные о 150 экземплярах ириса, по 50 экземпляров из трёх видов *Iris setosa*, *Iris virginica*, *Iris versicolor*. Каждый ирис описан четырьмя параметрами: длина и ширина наружной доли околоцветника (sepal length и sepal width соответственно), длина и ширина внутренней доли околоцветника (petal length и petal width соответственно) (рис. 16).

Обозначения данных:



Применим к этим данным метод главных компонент для понижения размерности до двух. Результат этой процедуры изображен на рис. 17. Такое представление данных легче воспринимается. Также на рис. 17 визуально можно разделить данные на два множества, а можно примерить метод k -средних (рис. 18). Видно, что данный метод довольно успешно разделяет данные на три кластера.

Набор данных размеченный, поэтому на этом наборе можно применить и методы обучения с учителем (т. е. решить задачу классификации).

Отметим, что наиболее универсальным и гибким подходом к классификации является использование **нейронных сетей**.

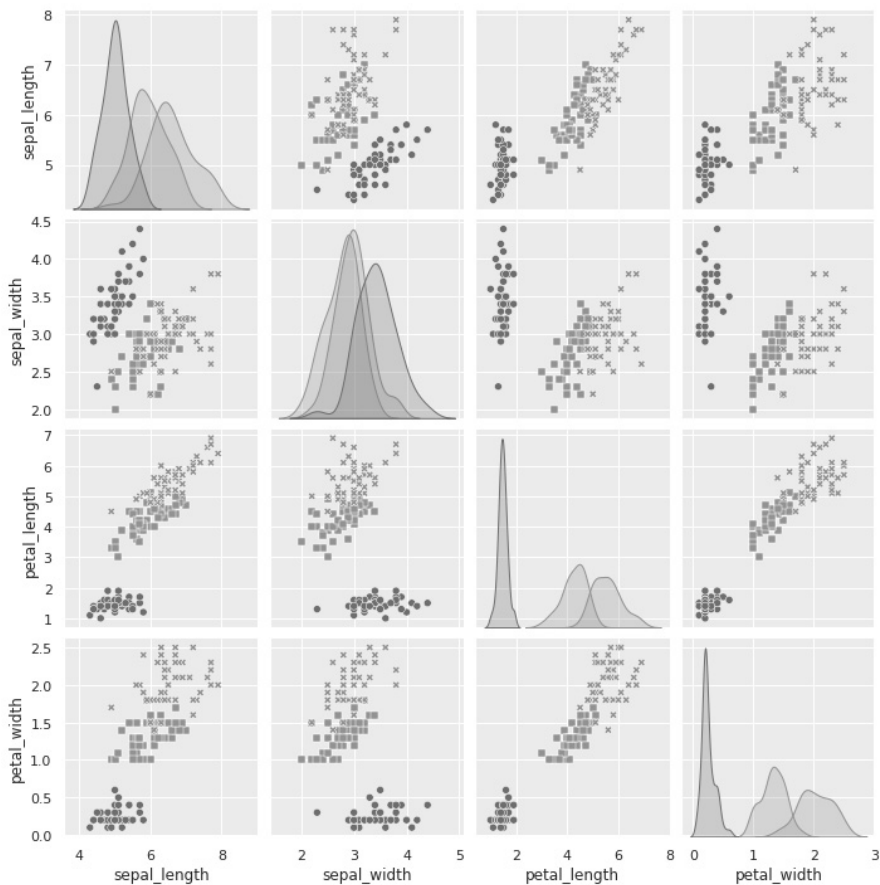


Рис. 16. Набор данных «Ирисы Фишера»

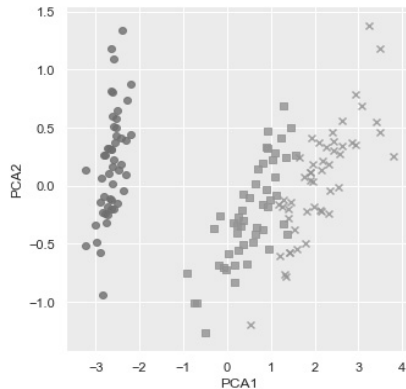


Рис. 17. Проекция набора данных «Ирисы Фишера» на двумерное пространство методом главных компонент

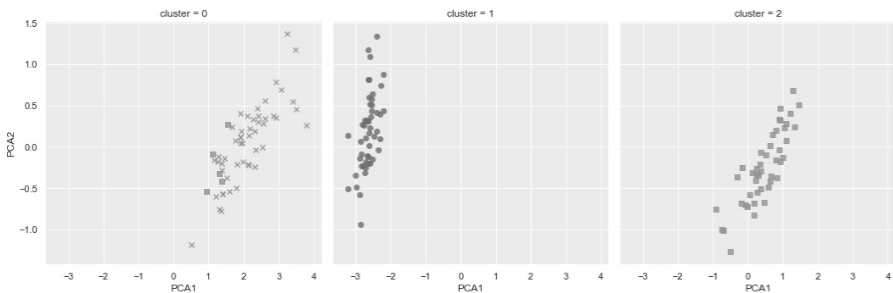


Рис. 18. Кластеризация спроецированных данных методом k -средних

ИСКУССТВЕННЫЕ НЕЙРОННЫЕ СЕТИ ДЛЯ РЕШЕНИЯ ОБРАТНЫХ ЗАДАЧ

Существует достаточно обширный класс методов, в котором связь «откликов» геофизической модели и ее параметров выражается в виде некоторого «черного ящика», имеющего структуру, аналогичную современному представлению ученых о устройстве внутренних связей человеческого мозга – сети нейронов. Поэтому зако-

номерны попытки применения математической модели *искусственных нейронных сетей* (ИНС, НС, ANN – англ.) не только для автоматизации достаточно простых и рутинных процессов, выполняемых людьми, но и для решения наукоёмких вычислительно-сложных задач.

Суть данных методов сводится к выбору архитектуры нейронной сети, заданию *гиперпараметров*, способу представления и предобработке входных данных, структуре целевой функции, выбору метода для нахождения параметров сети (оптимизации), определению критериев качества для процесса обучения и др. Наиболее распространёнными являются три вида нейронных сетей:

1) *полносвязная нейронная сеть* – это сеть, в которой каждый нейрон связан со всеми остальными нейронами, находящимися в соседних слоях, и в которой все связи направлены строго от входных нейронов к выходным;

2) *сверточная нейронная сеть* – сеть со специальной архитектурой, оперирующая понятиями свертки, подвыборки и полносвязных слоев, изначально нацеленная на эффективное распознавание изображений;

3) *рекуррентная нейронная сеть* – вид нейронных сетей, где связи между элементами образуют направленную последовательность, благодаря чему появляется возможность обрабатывать серии событий во времени или пространственные цепочки.

Рассмотрим несколько аспектов применения нейронных сетей к интерпретации площадных данных наземных электромагнитных исследований на примере трёхмерных данных магниторазведки и импульсной электроразведки. Искомые величинами при интерпретации данных магниторазведки являются параметры аномальной намагниченности геологических структур по трём координатным осям. Их оценка производится с использованием полученной в результате практических полевых исследований информации о распределении вектора магнитной индукции по дневной поверхности изучаемой геологической области. Логично предполагать, что критерием качества применения нами нейронной сети к решению данной задачи было бы малое отклонение полученных с помощью нее оценок для намагниченности геологических структур от их истинных значений. Данный критерий выражается с помощью общепринятой целевой функции (МНК-функции ошибки).

Для возможности восстановления функции аномальной намагниченности геологической среды необходимо ввести ее некоторую параметризацию – т. е. выделить ее основные, значимые для данной задачи параметры, и задать способ, которым можно эти параметры изменять, получая «отклик» в приёмниках. Для данной специфики задачи одной из возможных параметризаций может служить представление функции аномальной намагниченности в виде кусочно-постоянной, заданной на дискретизированной в виде блочных структур геологической области. В этом случае оцениваемая геологическая среда состоит из набора блоков, в каждом из которых значение аномальной намагниченности является постоянным. Тогда вектор из оцениваемых значений намагниченности является искомым и вместе с тем составляет выходной вектор нейронной сети без каких-либо изменений и функций активации – решается задача регрессии.

Следующим важным моментом является способ представления и предобработки входных данных. Как уже отмечалось ранее, входными данными для данной задачи (помимо параметризации, гиперпараметров и т. п.) являются зарегистрированные значения магнитного поля в приёмниках. При этом на разных участках месторождения может быть снято разное число значений компонент вектора магнитной индукции. Более того, при равном количестве практических данных могут кардинально различаться местоположения их съёмки – в каких-то местах могут быть сгущения, а в каких-то разрядка. Также могут отличаться измеряемые в приёмниках поля величины. Вместе с тем рассматриваемые нейронные сети имеют единое количество нейронов входного слоя, которые содержат информацию о зарегистрированных сигналах. Поэтому необходим некоторый способ для унификации различного количества практических данных в вектор постоянной длины, который может быть задан на входной слой нейронной сети.

Одним из возможных вариантов является применение интерполяции с одинаковым для различных конфигураций снятых данных количеством степеней свобод. Для этих целей хороший результат может быть получен с использованием сглаживающего сплайна (линейный, эрмитов и т. п.), построенного на некоторой типичной для большого класса месторождений дискретизации дневной поверхности. Таким

образом, входным вектором для нейронной сети будет служить не вектор с компонентами снятого сигнала индукции, а унифицированный вектор (имеющий одинаковое количество элементов для различных конфигураций приёмников) сглаженного сигнала. Сглаживание применяется не только для унификации сигнала для «входного» слоя нейронной сети. При интерпретации данных электромагнитных зондирований с помощью классических методов, основанных на решении прямых задач, ряд исследователей также применяют сглаживание снятого сигнала, для улучшения сходимости инверсии. Следующий момент для успешного применения нейронной сети для интерпретации геофизических данных является главным. Именно архитектура в подавляющем большинстве случаев отвечает за адекватность и успешность результата нейронной сети. Это тот «черный ящик», который по снятым на поверхности данным покажет распределение намагниченности на глубине. Для задач магниторазведки возможно использование вариаций *полносвязной нейронной сети*, оптимальное количество слоев и количество нейронов в каждом из слоев которой может быть установлено экспериментально. Однако по аналогии с современными достижениями в области интеллектуального компьютерного зрения для этих целей лучшим образом подходят *сверточные нейронные сети*. При этом возможно использование как собственных архитектур, так и общепринятых, таких как *VGG16*, *MobileNet* и др., полносвязный слой которых переучен на обучающей выборке на данных магниторазведки методом *Transfer Learning*. При этом для анализа данных импульсной электроразведки добавляется еще одна компонента – время. В этом случае над нейронной сетью на одном временном интервале содержится нейронная сеть, «улавливающая» связи на последовательности времен – *рекуррентная нейронная сеть*.

Осталось рассмотреть способ получения обучающей и тестовой выборки для успешного обучения и валидации модели. Существует ряд эвристических и практических оценок того, какой необходим объём обучающей выборки для того, чтобы нейронная сеть смогла при обучении показывать приемлемый результат. Современные подходы наподобие *Transfer Learning* уже позволяют использовать в сотни и тысячи раз меньший объём выборки за счёт обучения основными сущностями и абстракциями на очень большом датасете (*dataset*), что

является серьёзной проблемой для интерпретации геофизических данных, так как свободно доступного датасета с сотнями тысяч геологических моделей и полученных для них сигналов в приёмниках нет и быть не может. Причинами этого являются государственный суверенитет в плане энергобезопасности и то, что достаточного количества месторождений на данный момент в мире просто не разведано. Остается единственный вариант – создавать датасет синтетически с дальнейшим его дополнением практическими данными. Для создания синтетического датасета необходимо выбрать типичную для широкого класса месторождений параметризацию в виде блочных структур. Для каждой такой модели с использованием прямого моделирования необходимо посчитать отклики в приёмниках. Полученный набор данных будет являться датасетом для обучения и тестирования. При этом для повышения точности результата нейронной сети данные в датасет можно добавлять «на лету», создавая новые модели и соответствующие для них отклики.

БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. *Вагин Д. В.* Оценивание параметров в обратных задачах: учебное пособие / Д. В. Вагин. – Новосибирск : Изд-во НГТУ, 2019. – 48 с.
2. *Амосов А. А.* Вычислительные методы для инженеров / А. А. Амосов, Ю. А. Дубинский, Н. В. Копченова, – Москва : Высшая школа, 1994. – 544 с.
3. *Самарский А. А.* Численные методы решения обратных задач математической физики : учебное пособие / А. А. Самарский, В. П. Николаевич. – Изд. 3-е. – Москва : Изд-во ЛКИ, 2009. – 408 с. – ISBN 978-5-382-00990-2.
4. *Тихонов А. Н.* Методы решения некорректных задач / А. Н. Тихонов, В. Я. Арсенин. – Изд. 2-е. – Москва : Наука, 1979. – 285 с.
5. *Персова М. Г.* Современные компьютерные технологии : конспект лекций / М. Г. Персова, Ю. Г. Соловейчик, П. А. Домников. – Новосибирск : Изд-во НГТУ, 2014. – 80 с. – ISBN 978-5-7782-2427-8.
6. *Симон Е. И.* Разработка методов интерпретации данных при зондированиях трёхмерной среды нестационарным электромагнитным полем : дис. ... канд. техн. наук: 05.13.18 / Е.И. Симон. – Новосибирск : Изд-во НГТУ, 2014. – 144 с.
7. *Гладков Л. А.* Генетические алгоритмы / Л. А. Гладков, В. В. Курейчик, В. М. Курейчик. – Москва : Физматлит, 2006. – 320 с. – ISBN 5-9221-0510-8.
8. *Плас Дж. Вандер.* Python для сложных задач: наука о данных и машинное обучение / Дж. Вандер Плас. – Санкт-Петербург : Питер, 2018. – 576 с. – ISBN 978-5-496-03068-7.
9. *Золотых Н. Ю.* Машинное обучение и анализ данных / Н. Ю. Золотых. – URL: <http://www.uic.unn.ru/~zny/ml>.

ОГЛАВЛЕНИЕ

Введение	3
1. Основные определения.....	4
2. Классификация обратных задач математической физики.....	9
3. Регуляризация обратных задач	11
Выбор параметра регуляризации	13
4. МНК функционал обратной задачи, метод Гаусса–Ньютона	15
Тестирование процедур	19
5. Параметризация в обратных задачах	22
Примеры решения обратных задач методом Гаусса–Ньютона.....	25
Задачи	32
6. Генетический алгоритм для решения обратных задач	34
Кроссинговер	37
Мутации	38
Селекция	38
Нечёткие генетические алгоритмы решения задач оптимизации и проектирования	39
Практическое применение.....	41
Пример программы подбора параметров с помощью ПГА.....	42
Задачи	46
7. Машинное обучение	47
Обучение без учителя	47
Искусственные нейронные сети для решения обратных задач.....	52
Библиографический список	57

**Вагин Денис Владимирович
Патрушев Илья Игоревич
Гриф Александр Михайлович**

ЦИФРОВЫЕ МОДЕЛИ И ОЦЕНИВАНИЕ ПАРАМЕТРОВ

Учебное пособие

Редактор *И.Л. Кескевич*
Выпускающий редактор *И.П. Брованова*
Корректор *И.Е. Семенова*
Дизайн обложки *А.В. Ладыжская*
Компьютерная верстка *С.И. Ткачева*

Налоговая льгота – Общероссийский классификатор продукции
Издание соответствует коду 95 3000 ОК 005-93 (ОКП)

Подписано в печать 09.04.2021. Формат 60 × 84 1/16. Бумага офсетная. Тираж 50 экз.
Уч.-изд. л. 3,48. Печ. л. 3,75. Изд. № 58. Заказ № 407. Цена договорная

Отпечатано в типографии
Новосибирского государственного технического университета
630073, г. Новосибирск, пр. К. Маркса, 20