

Министерство образования и науки Республики Казахстан
Евразийский национальный университет им. Л.Н.Гумилева

Телегенов Куат Аманжолович

Исследование изгиба балки на упругом основании
(научно-педагогическое направление)

Астана, 2014 г.

Министерство образования и науки Республики Казахстан

Евразийский национальный университет им. Л.Н.Гумилева

Допущен к защите:

Декан факультета

_____ Джайчибеков Н.Ж.

« ____ » _____ 2014 г.

Магистерская диссертация

ИССЛЕДОВАНИЕ ИЗГИБА БАЛКИ НА УПРУГОМ ОСНОВАНИИ

специальность: 6М060300 «Механика»
(научно-педагогическое направление)

Магистрант

К.А. Телегенов

Научный руководитель

к.т.н, профессор

К.С. Кишауов

Зав. кафедрой

А.Г. Ибраев

Астана, 2014 г.

СОДЕРЖАНИЕ

ВВЕДЕНИЕ	7
1 ОБЗОР БАЛКИ НА УПРУГОМ ОСНОВАНИИ	10
1.1 Балка как конструктивный элемент	10
1.2 Модели упругого основания	10
1.3 Балка на двухпараметрическом упругом основании	15
2 РЕШЕНИЕ ОСНОВНОГО ДИФФЕРЕНЦИАЛЬНОГО УРАВНЕНИЯ МЕТОДОМ КОНЕЧНЫХ ЭЛЕМЕНТОВ.....	20
2.1 Метод конечных элементов	20
2.2 Граничные условия	22
2.3 Метод Бубнова-Галеркина	23
2.4 Формулировка конечного элемента для балки на упругом основании.....	34
2.5 Решение методом конечных элементов.....	37
3 АНАЛИТИЧЕСКИЙ МЕТОД РЕШЕНИЯ УРАВНЕНИЯ БАЛКИ НА УПРУГОМ ОСНОВАНИИ.....	41
3.1 Формулировка проблемы	41
3.2 Решение граничной задачи	44
3.3 Преобразования при помощи сингулярных функций.....	45
3.4 Применение в компьютерных программных обеспечениях	46
4 РЕЗУЛЬТАТЫ РЕШЕННЫХ ЗАДАЧ	48
4.1 Распределенная нагрузка на основание Власова	48
4.2 Непрерывная балка	49
4.3 Примеры, решенные аналитическим решением.....	53
4.4 Двухпараметрическая модель на упругом основании	53
4.5 Балка на Винклеровском основании	56
4.6 Сравнительный анализ двух методов	60
ЗАКЛЮЧЕНИЕ	62
СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ	64
ПРИЛОЖЕНИЯ.....	66

Нормативные ссылки

В настоящем положении использованы ссылки на следующие нормативные документы:

Закон Республики Казахстан «Об образовании» от 27.07.2007 г.

ГОСО РК Послевузовское образование. Магистратура от 23.08.2012г. №1080.

Типовые правила проведения текущего контроля успеваемости, промежуточной и итоговой аттестации обучающихся в высших учебных заведениях №94 от 16.03 2011г.

В настоящем положении применяются термины и определения в соответствии с Законом Республики Казахстан «Об образовании» от 27.07.2007 г., ГОСО РК 5.04.033-2011.

Термины и определения

SAP2000 – система трехмерного статического и динамического анализа методом конечного элемента и проектирование сооружений. CSI SAP2000 следует в той же самой традиции, показывающей очень сложный, интуитивный и универсальный пользовательский интерфейс, приведенный в действие непревзойденным механизмом анализа и средствами проектирования для инженеров, работающих над транспортировкой, индустриальными, общественными работами, спортивными состязаниями, и другими средствами. От основанной графической среды моделирования трехмерного объекта, к большому разнообразию опций анализа и проектирования, полностью интегрированных через один пользовательский интерфейс, CSI SAP2000, оказался наиболее интегрированной, производительной и практической универсальной структурной программой на рынке.

ANSYS – универсальная программная система конечно-элементного (МКЭ) анализа, КЭ решения линейных и нелинейных, стационарных и нестационарных пространственных задач механики деформируемого твёрдого тела и механики конструкций (включая нестационарные геометрически и физически нелинейные задачи контактного взаимодействия элементов конструкций), задач механики жидкости и газа, теплопередачи и теплообмена, электродинамики, акустики, а также механики связанных полей. Моделирование и анализ в некоторых областях промышленности позволяет избежать дорогостоящих и длительных циклов разработки типа «проектирование — изготовление — испытания».

MATLAB – это высокоуровневый язык технических расчетов, интерактивная среда разработки алгоритмов и современный инструмент анализа данных. MATLAB по сравнению с традиционными языками программирования (C/C++, Java, Pascal, FORTRAN) позволяет на порядок сократить время решения типовых задач и значительно упрощает разработку новых алгоритмов. MATLAB представляет собой основу всего семейства продуктов MathWorks и является главным инструментом для решения широкого спектра научных и прикладных задач, в таких областях как: моделирование объектов и разработка систем управления, проектирование коммуникационных систем, обработка сигналов и изображений, измерение сигналов и тестирование, финансовое моделирование, вычислительная биология и др. Ядро MATLAB позволяет максимально просто работать с матрицами реальных, комплексных и аналитических типов данных. Содержит встроенные функции линейной алгебры (LAPACK, BLAS), быстрого Фурье преобразования (FFTW), функции для работы с полиномами, функции базовой статистики и численного решения дифференциальных уравнений. Все встроенные функции ядра MATLAB разработаны и оптимизированы специалистами и работают быстрее или так же, как их эквивалент на C/C++.

Сокращения

МКЭ – Метод конечных элементов

СЛАУ – Система линейных алгебраических уравнений

MATLAB – Matrix Laboratory

2Д – двухмерная

3Д – трехмерная

КЭ – конечный элемент

ВВЕДЕНИЕ

Конструкции, опирающиеся на упругое основание, имеют самое широкое применение в строительстве. Примерами упругого основания могут служить грунт или сваи, на которые опирается сооружение, близко расположенные друг от друга колонны, балки или ригели рам, на которых лежит какая-либо конструкция - балка, ферма и др. В настоящей работе под упругим основанием подразумевается в основном естественное грунтовое или свайное основание.

Методы расчета конструкций, лежащих на грунте, в зависимости от принятых моделей основания можно разбить на три группы

- методы, базирующиеся на винклеровой модели основания;
- методы, базирующиеся на теории упругого полупространства;
- методы, базирующиеся на комбинированных моделях упругого основания.

При правильном выборе численного значения коэффициента жесткости винклеровского основания и учета в необходимых случаях его переменности результаты расчета конструкций с использованием этой модели близко соответствуют опытным данным

Теория балок и плит, лежащих на упругом основании, представляет собой весьма обширный раздел современной строительной механики. В этой области выполнено огромное количество работ, многие из которых содержат ценные практические методы расчета балок и плит на упругом основании.

Однако существующие методы расчета еще далеко не совершенны и не дают ответа на множество разнообразных вопросов, выдвигаемых строительной практикой. Большая часть этих методов носит слишком сложный для практических вычислений характер; не могут считаться совершенными и те гипотезы, которые принимаются для работы, естественного грунта. Имеющиеся методы не позволяют рассчитывать сложные пространственные системы на упругом основании. Так, гипотеза коэффициента постели, рассматривающая упругое основание как систему отдельных, не связанных между собой пружин и позволяющая вследствие этого достаточно просто производить расчет конструкций на упругом основании, в ряде случаев приводит к результатам, резко расходящимся с действительностью.

Актуальность данной работы – гипотеза упругого изотропного полупространства, согласно современному грунтоведению, достаточно полно отражает физические свойства естественного основания. Однако методы расчета, базирующиеся на этой гипотезе, очень сложны, в результате чего практически приемлемые решения получены здесь только для весьма ограниченного круга задач.

Таким образом, перед современной теорией расчета фундаментов на упругом основании стоит ряд серьезных проблем, наиболее важными из которых являются:

- уточнение расчетных схем основания, в смысле сближения их с действительностью;

– разработка методов расчета сложных пространственных конструкций с учетом упругой податливости грунта;

– упрощение методов расчета сооружений на упругом основании в целях более широкого внедрения их в инженерную практику.

В отношении уточнения расчетов можно полагать, что учет упругопластической работы грунта основания позволит в дальнейшем приблизиться к действительной работе конструкций.

Вопросы расчета сложных пространственных конструкций на упругом основании, которые стоят перед инженерной практикой, могут быть, очевидно, наиболее эффективно решены при помощи приближенных методов. Эти же методы дают возможность упростить расчетные выражения.

В настоящей работе предлагается техническая теория расчета конструкций балки на упругом основании. Модель Власова с двумя параметрами решается при помощи методов конечных элементов и аналитическим методом.

Предлагаемый метод с успехом может быть применен и в горном деле при решении задач, относящихся к определению напряжений и деформаций в однослойных и многослойных пластах горных выработок, имеющих горизонтальное или наклонное расположение.

Объектом исследования являются балочный элемент на упругом основании.

Цель научно-исследовательской работы – упрощение методов расчета балки на упругом основании для последующего внедрения в инженерную практику.

Задачи исследования:

– Разработка и улучшение текущих методов расчета балки на упругом основании;

– Приближение расчетных схем изгиба балки к действительности;

– Разработка алгоритма решения системы уравнений изгиба балки.

Научная новизна работы заключается в разработанном алгоритме решения балки на упругом основании методом конечных элементов. Также использование сингулярных функций для решения балки на упругом основании.

Практическая значимость данной работы заключается в будущем внедрении данного решения в инженерную практику, что позволит упростить расчеты конструкции на упругом основании.

Структура и объем работы. Магистерская диссертация состоит из 65 страниц. Структура диссертационной работы - введение, три раздела, заключение, список используемой литературы из 36 наименований и 3 приложения. В работе имеются 29 иллюстраций и 4 таблиц.

Содержание работы изложено в трех разделах.

В первом разделе рассматривается методика решений задачи на упругом основании методом конечных элементов. Задача балки сводится к дифференциальному уравнению, затем данное дифференциальное уравнение сводится к системе линейных алгебраических уравнений и решается методом Гаусса в среде MATLAB.

Во втором разделе рассматривается методика решений задачи на упругом основании аналитическим методом при помощи сингулярных функций. Это упрощает решение сложных задач, с составной нагрузкой.

В третьем разделе детально рассматриваются решенные задачи и сравниваются с текущими решениями.

1 ОБЗОР БАЛКИ НА УПРУГОМ ОСНОВАНИИ

1.1 Балка как конструктивный элемент

Балка - это конструктивный элемент, представляющий собой горизонтальный либо наклонный брус, работающий предпочтительно на изгиб [1]. На практике, как правило, горизонтально расположенная балка воспринимает вертикальную поперечную весовую нагрузку, но в отдельных случаях нужно рассматривать воздействие и возможных горизонтальных поперечных сил (скажем, ветровую нагрузку либо при учёте потенциального землетрясения). Нагруженная балка, в свою очередь, воздействует на опоры, которыми могут являться колонны, подвесы, стены либо другие балки (перекладины). После этого нагрузка передаётся дальше и в результате, в большинстве случаев, понимается конструктивными элементами, работающими на сжатие — опорами. Отдельно дозволено выделить случай ферменной конструкции, в которой стретжни покоятся на горизонтальной балке.

Прочностные качества балки зависят от нескольких её характеристик:

- площадь и форма её поперечного сечения;
- длина балки;
- материал балки;
- способ её закрепления.

В современных сооружениях применяются, как правило, железные, железобетонные либо деревянные балки. Одним из самых распространённых типов поперечного сечения железной балки является двутавровое сечение. Двутавровые балки применяют при возведении каркасов зданий и мостов. Также используют тавровые балки, профили, балки с полым профилем (в частности, трубы), балки с угловым профилем сечения.

1.2 Модели упругого основания

Нагрузка от сооружения, передается через его подошву на основание. Под влиянием указанной нагрузки возникают осадки основания. Полный учет природных механических свойств грунтового основания, проявляющихся при их погружении, практически невозможен. Поэтому для определения деформативно-напряженного состояния оснований и сооружений используется расчетная модель основания, схематически описывающая природные механические процессы, протекающие в грунтах. Очевидно, что для получения достаточно достоверных результатов расчета модель должна правильно отображать главные механические свойства грунтового основания.

Исторически первой моделью грунтового основания является модель Фусса-Винклера, часто именуемая в технической литературе винклеровым основанием или гипотезой коэффициента постели. Расчетная механическая модель винклерового основания представляет собой ряд не связанных между собой упругих пружин, укрепленных на жестком основании как показано на рисунке 1. Штамп, приложенный к поверхности модели, при погружении

вдавливается на глубину, пропорциональную среднему удельному давлению, а при снятии нагрузки возвращается в исходное положение. Поверхность модели за пределами штампа не деформируется.

Механические свойства модели характеризуются коэффициентом жесткости K , называемым также часто коэффициентом постели основания. По физическому смыслу коэффициент жесткости означает величину усилия в кг, которое необходимо приложить к 1 см^2 поверхности основания, чтобы последнее осело на 1 см .

Для плит размерность коэффициента жесткости — кг/см^3 ; для балок при ширине подошвы b вычисляется погонный коэффициент постели

$$K_n = Kb \quad (1.1)$$

его размерность — кг/см^2 (или т/м^2). В случае абсолютно жесткого основания $K = \infty$ абсолютно податливого — $K = 0$.

При таких условиях реактивное давление основания p в каждой точке контакта подошвы фундамента сооружения с грунтом пропорционально осадке грунта в этой точке, т. е.

$$p = K\delta \quad (1.2)$$

Эта зависимость является фундаментальным уравнением всей теории расчета конструкций на упругом винклеровом основании.

Кроме того, в расчетах обычно принимаются следующие допущения:

- фундамент сохраняет связь с основанием независимо от знака перемещения, т. е. между балкой и основанием не должно быть разрывности;
- между фундаментом и поверхностью грунта при изгибе отсутствует трение;
- все деформации предполагаются достаточно малыми, благодаря чему можно пользоваться принципом наложения, суммируя деформации от различных воздействий.

Винклерова модель является самой простой из всех существующих расчетных схем грунтового основания. При правильном выборе значений коэффициента жесткости расчет, основанный на такой схематизации грунта, для большинства случаев практики дает удовлетворительное по точности приближенное решение.

С формальной точки зрения винклерова модель не является достаточно строгой. Действительно, наблюдения за натурными сооружениями и экспериментальные исследования работы оснований и фундаментов показывают, что осадка основания зависит не только от нагрузки в данной точке, но и от нагрузку соседних точек; что грунт оседает не только под фундаментом, но и по соседству с ним; что величина коэффициента постели зависит не только от вида грунта, но также от величины и формы загруженной площади и, наконец,

что грунт на растяжение не работает. В инженерной практике, статический анализ и проектирование балок на упругом основании разработаны на предположении, что силы реакции основания в каждой точке пропорциональны изгибу балки в данной точке. Вертикальные характеристики деформации фундамента моделируются при помощи одинаковых, независимых, дискретных и близко расположенных друг к другу линейно-упругих пружин. Коэффициент пропорциональности этих пружин известен как коэффициент жесткости грунтового основания, или постели k_s . Это простое и довольно грубое механическое представление грунтового основания впервые было введено Винклером в 1867 [2][3].

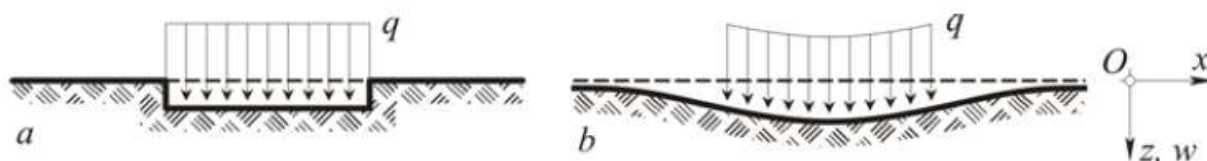


Рис. 1- Изгиб упругого основания при равномерном давлении: а - модель Винклера; б - реальное грунтовое основание

Модель Винклера, которая была первоначально разработана для анализа железнодорожных путей, является достаточно простой, но не точно отображает характеристики реального грунтового основания. Одним из существенных недостатков модели Винклера является отсутствием взаимодействия между соседними пружинами, следовательно, не учитывается касательное напряжение сдвига, которое происходит в основании грунта. Кроме того, возникает разрыв между нагруженной и ненагруженной частью поверхности основания грунта, но в действительности поверхность почвы не показывает никаких разрывов.

Модель Винклера можно интерпретировать как систему независимых друг от друга вертикальных пружин с жесткостью k . Потенциальная энергия деформации упругого основания может быть выражена как:

$$U_f = \int_0^l \frac{1}{2} kbw^2 dx \quad (1.3)$$

где b и l ширина и длина деформированной зоны и w является вертикальным смещением. Модель Винклера предполагает, что смещение появляется только в нагруженной зоне. Вне этой зоны прогибы равны нулю. Это предположение приводит к области прерывистого перемещения и это главный недостаток модели Винклера.

Преодолеть недостаток модели Винклера можно при помощи моделирования определенного взаимодействия между независимыми пружинами путем визуализации различных типов соединений. Под различными типами соединений подразумевается: изгибающиеся элементы (балки для одномерного и пластины для двумерного элементов), слои срезающей силы и

деформированная пред-натянутая мембрана [3]. Модель, предложенная Филоненко-Бородичом в 1940 году, основана на непрерывности модели Винклера при помощи соединения отдельных пружин к тонкой упругой мембране под действием постоянного напряжения. В модели предложенной Нетенуи взаимодействие между независимыми пружинами осуществляется путем введения упругой пластины для трехмерных и упругой балки для двумерных задач, которые деформируются только при изгибе. Другая модель основания, предложенная Пастернаком [4], моделируется при помощи создания взаимодействия между пружинами, путем соединения концов пружин к слою, состоящему из несжимаемых вертикальных элементов, которые деформируются только при помощи касательных сил [3][5]. Такая математическая модель содержит еще один постоянный параметр, который характеризует взаимодействие между пружинами, и называется двухпараметрической или механической моделью в соответствии с рисунком 2.

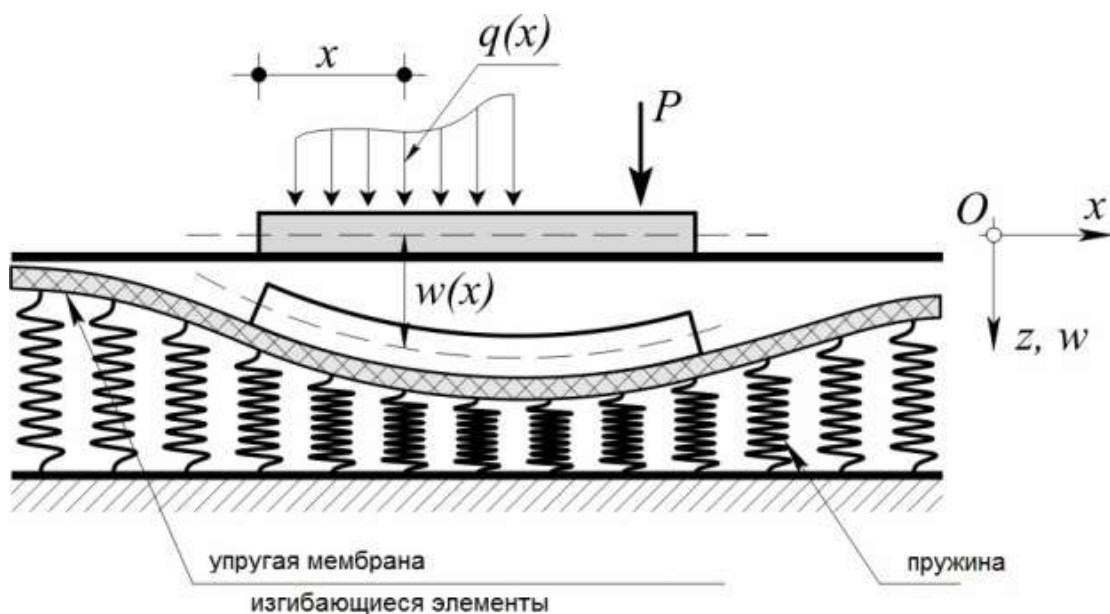


Рис. 2 - Балка на двухпараметрическом упругом основании

Другой подход к разработке и улучшению модели основания, начинается с трех сложных уравнений с частными производными определяющих поведение почвы в полу-бесконечном сплошной среде, а затем ввести упрощающие предположения относительно смещений и / или напряжений для решения. Это называется упрощенной моделью бесконечной среды.

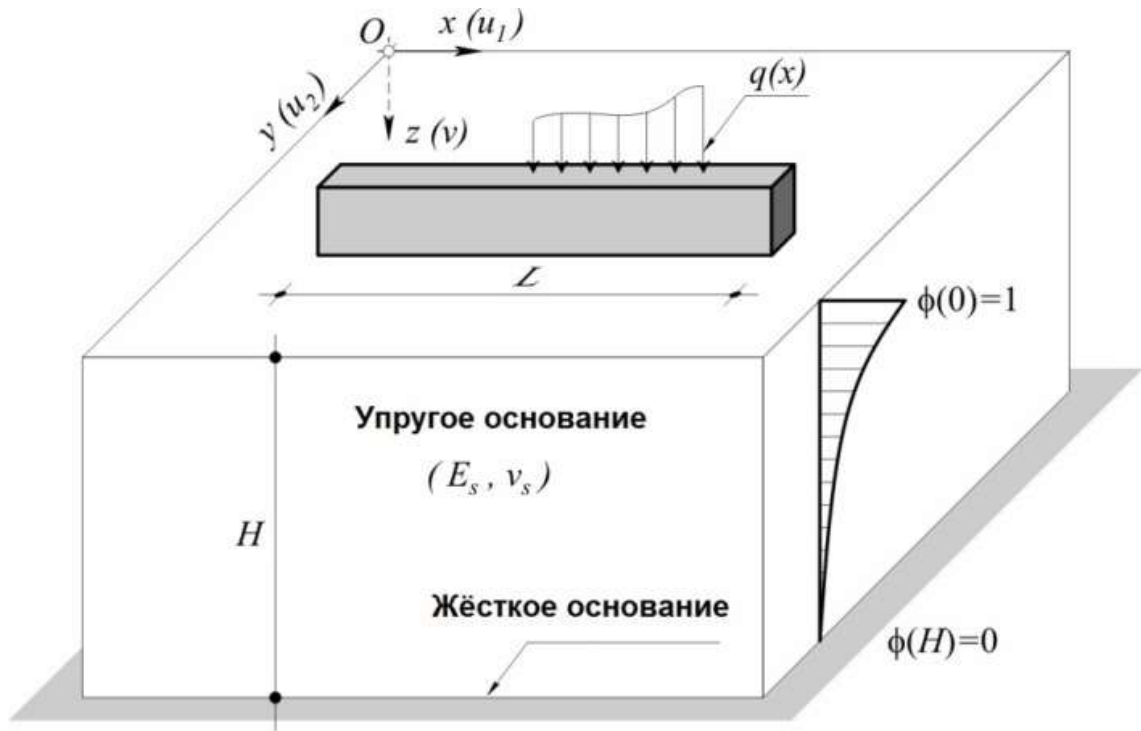


Рис. 3 - Модель основания Власова

Рейснер (Reissner) впервые применил упрощенную концепцию бесконечной среды, назвав ее «Упрощенная модель бесконечной среды Рейсснера» [6][7]. Власов и Леонтьев [8] адаптировавшие упрощенную концепция бесконечной среды, основанную на вариационном принципе, произвели двухпараметрическую модель основания. Данный метод рассматривает основание как упругий слой, и связь задается ограничением изгиба в основании, по функции соответствующей формы $\phi(z)$. Двухпараметрическая модель Власова как показано на рисунке 3. учитывает эффект пренебрегаемой энергией деформации сдвига в почве и поперечных сил путем введения произвольного параметра γ , для описания вертикального распределения деформаций в грунте. Однако авторы не предусматривает никакого механизма для расчета параметра γ . Позже Джонс и Ксенофонт [9] установили связь между параметром γ и характеристиками сдвига, но не предложили метода для расчета значения. После Джонса и Ксенофонта, Vallabhan и Das [10] определили параметр γ как зависимость от характеристик балки и основания используя итерационный метод. Они назвали ее модифицированная модель основания Власова [11].

Общее выражение для энергии деформации в двухпараметрических моделях представлена в форме:

$$U_f = \int_0^l \frac{1}{2} kbw^2 dx + \int_0^l \frac{1}{2} Gb \left(\frac{dw}{dx} \right)^2 dx \quad (1.4)$$

Вторая часть уравнения включает в себя второй параметр G , который представляет жесткость обобщенной пружины вращения. Существуют

различные интерпретации о физическом представлении G и соотношением к первому параметру k :

– Модель Филоненко-Бородича – параметр G представлен как внутренняя сила напряжения в виртуальном ряду, размещенном на поперечных пружинах, которые сдерживает вертикальные смещения пружин;

– Модель Hetenyi - ограничивает вертикальные перемещения, путем добавления воображаемой балки при изгибе. Второй параметр представляет жесткость данной балки;

– Модель Пастернака – параметр G представляет собой модуль сдвига виртуального слоя, который объединяет вертикальные пружинные элементы;

– Модель Власова-Леонтьева – параметры k и G получены на основе эластичного континуума, предполагаемого для поля смещений.

1.3 Балка на двухпараметрическом упругом основании

Многие инженерные конструкции, лежащие на грунте, можно рассматривать при их расчете как балки на упругом основании. Например, ленточные фундаменты, фундаменты подкрановых путей, доки, балочные плиты, бескаркасные жилые и гражданские здания и другие сооружения промышленного, гражданского, гидротехнического и дорожного строительства рассчитываются как балки, лежащие на упругом основании.

Теория расчета балок на упругом основании хорошо разработана в трудах советских и зарубежных авторов применительно к балке постоянного сечения, лежащей на упругом основании с постоянным коэффициентом жесткости. Однако общий, практически наиболее важный случай основания переменной жесткости исследован недостаточно; большинство из имеющихся методов расчета балок на упругом основании с переменным коэффициентом жесткости хотя и имеют теоретическую ценность, но для практического применения мало пригодны.

Для преодоления недостатков текущих моделей основания, а именно модели Винклера, рассмотрим модель упругого основания, основанная на вариационном методе.

Грунтовое основание рассматривается как упругая, изотропная и однородная бесконечная среда с постоянной толщиной H , и имеющий модуль упругости и коэффициент Пуассона, E_s и ν_s соответственно в соответствии рисунком 4. Толщину грунта H можно рассматривать как глубину от поверхности основания до предполагаемого жесткого слоя, или глубину, для которой деформации из-за нагрузок на балку не являются значимыми.

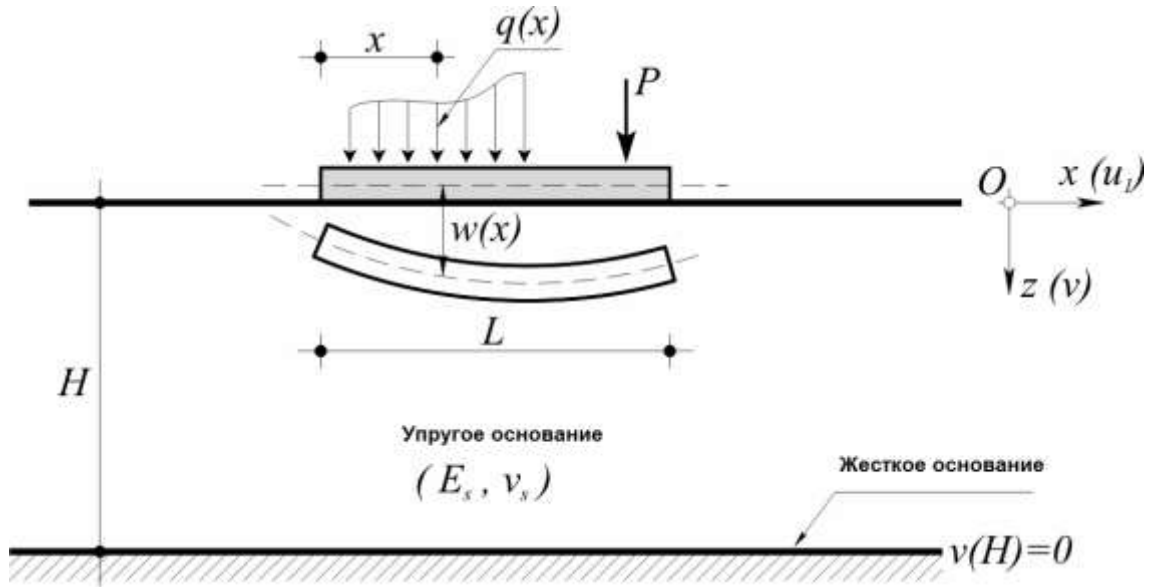


Рис. 4 - Балка на двухпараметрической модели Власова

Полагая условия плоской деформации, полная энергия деформации в системе балка-почва будет [10][11]:

$$\Pi = \frac{1}{2} \int_0^L E_b I_b \left(\frac{d^2 w}{dx^2} \right)^2 dx + \frac{1}{2} \int_{-\infty}^{+\infty} \int_0^H b (\sigma_x \varepsilon_x + \sigma_z \varepsilon_z + \tau_{xz} \gamma_{xz}) dz dx - \int_0^L q(x) w dx \quad (1.5)$$

где $E_b I_b$ жесткость при изгибе балки; L - длина балки; $w(x)$ - вертикальный сдвиг линии центра тяжести балки; $\sigma_x, \sigma_z, \tau_{xz}, \varepsilon_x, \varepsilon_z$ and γ_{xz} - напряжения и соответствующая деформация грунтового основания; $q(x)$ - распределенная нагрузка на балку.

Предположения относительно вертикального сдвига [8],

$$v(x, z) = w(x) \phi(z), \text{ такой что } \phi(0) = 1 \text{ и } \phi(H) = 0 \quad (1.6)$$

и никаких горизонтальных перемещений

$$u_1(x, z) = 0 \quad (1.7)$$

используя уравнения натяжения-сдвига упругости, внутренние напряжения грунтового основания могут быть выражены как:

$$\varepsilon_x = \frac{\partial u_1}{\partial x} = 0, \quad \varepsilon_z = \frac{\partial v}{\partial z} = w(x) \frac{\partial \phi}{\partial z}, \quad \gamma_{xz} = \frac{\partial u_1}{\partial z} + \frac{\partial v}{\partial x} = \frac{\partial w}{\partial x} \phi(z) \quad (1.8)$$

Подставляя выражение (1.8) в уравнение энергии деформации (1.5) и в определяющие соотношения плоской упругости

$$\begin{Bmatrix} \sigma_x \\ \sigma_z \\ \tau_{xz} \end{Bmatrix} = \frac{E_s(1-\nu_s)}{(1+\nu_s)(1-2\nu_s)} \begin{bmatrix} 1 & \frac{\nu_s}{1-\nu_s} & 0 \\ \frac{\nu_s}{1-\nu_s} & 1 & 0 \\ 0 & 0 & \frac{1-2\nu_s}{2(1-\nu_s)} \end{bmatrix} \begin{Bmatrix} \varepsilon_x \\ \varepsilon_z \\ \gamma_{xz} \end{Bmatrix} \quad (1.9)$$

Тогда полная энергия деформации в системе балка-почва может быть записана в следующем виде [11]

$$\begin{aligned} \Pi = & \frac{1}{2} \int_0^L E_b I_b \left(\frac{d^2 w}{dx^2} \right)^2 dx \\ & + \frac{1}{2} \int_{-\infty}^{+\infty} \int_0^H b \left[\frac{E_s(1-\nu_s)}{(1+\nu_s)(1-2\nu_s)} w^2(x) \left(\frac{d\phi}{dz} \right)^2 \right] dz dx \\ & - \int_0^L q(x) w dx \end{aligned} \quad (1.10)$$

Минимизация функции Π по отношению к w даст следующее основное уравнение для системы балка-почва:

$$E_b I_b \frac{d^4 w}{dx^4} - k_1 \frac{d^2 w}{dx^2} + k_s w = q(x) \quad (1.11)$$

$$k_s = \frac{E_s b (1-\nu_s)}{(1+\nu_s)(1-2\nu_s)} \int_0^H \left(\frac{d\phi}{dz} \right)^2 dz \quad (1.12)$$

является параметром сжатия основания (эквивалентно жесткости пружины Винклера или коэффициенту реакции земляного полотна) и

$$k_1 = \frac{E_s b}{2(1+\nu_s)} \int_0^H \phi^2 dz \quad (1.13)$$

параметр сдвига основания показали что, минимизация функции Π по отношению к ϕ даст

$$\phi(z) = \frac{\sinh \gamma \left(1 - \frac{z}{H}\right)}{\sinh \gamma} \quad (1.14)$$

где

$$(1.15)$$

$$\gamma^2 = H^2 \frac{1 - 2\nu_s}{2(1 - \nu_s)} \frac{\int_{-\infty}^{+\infty} \left(\frac{dw}{dx}\right)^2 dx}{\int_{-\infty}^{+\infty} w^2 dx}$$

Используя функция $\phi(z)$ из уравнения (1.14), новые формулы для параметров k_s и k_1 получают следующие формы [10]

$$k_s = \frac{E_s b(1 - \nu_s)}{(1 + \nu_s)(1 - 2\nu_s)H} \gamma \left(\frac{\sinh \gamma \cosh \gamma + \gamma}{2 \sinh^2 \gamma} \right) \quad (1.16)$$

$$k_1 = \frac{E_s bH}{2(1 + \nu_s)} \frac{1}{\gamma} \left(\frac{\sinh \gamma \cosh \gamma - \gamma}{2 \sinh^2 \gamma} \right) \quad (1.17)$$

Основное уравнение (1.11) балки на упругом основании зависит от вертикальной деформации, которая в свою очередь зависит от параметра вертикальной деформации почвы, γ , как показано на рисунке 5. Очевидно, что для $0 < \gamma < 2$, функция $\phi(z)$, приводит к линейным изменениям вертикальных перемещений сверху вниз, а при больших значениях $\gamma > 2$, быстрому рассеиванию вертикальных перемещений.

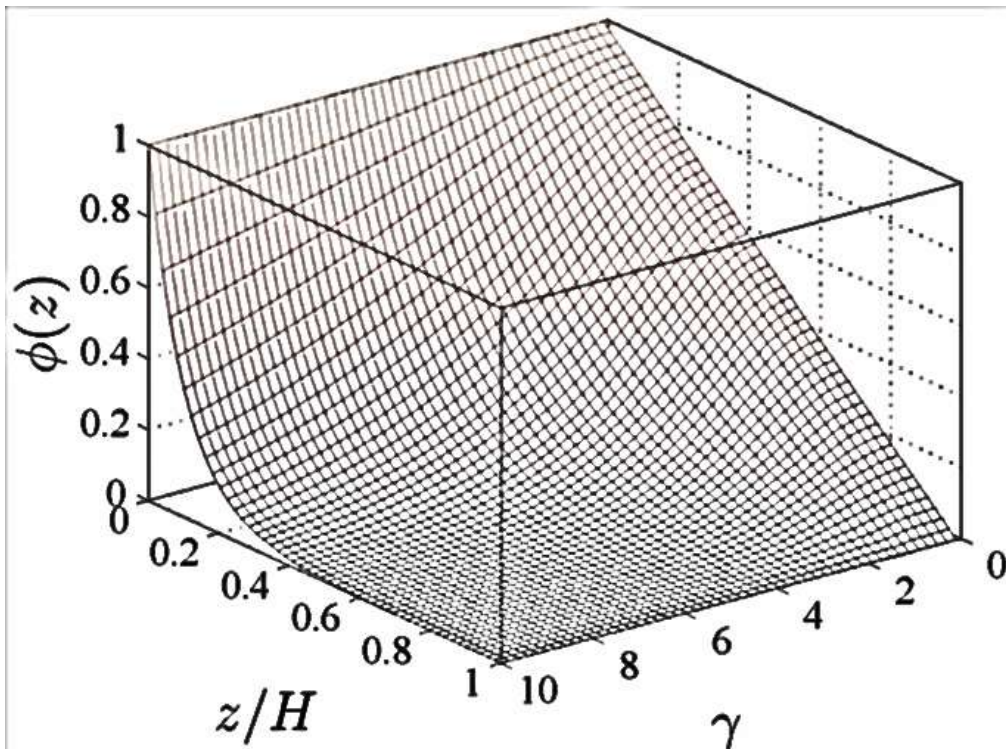


Рис. 5 - Функция $\phi(z)$ против γ и z/H

Для решения функции $\phi(z)$ изначально надо найти γ . Vallabhan and Das предложили итеративный метод расчета параметра γ . Предполагаемое начальное значение γ , позволяет вычислить параметры k_s и k_1 , используя уравнения (1.16) и (1.17). Далее, значение γ вычисляется по формуле (1.15) с использованием сдвига балки, найденным в предыдущем шаге. С этим вычисленным параметром

γ , новые значения k_s и k_1 подсчитываются при помощи уравнений (1.16) и (1.17). Итерационный процесс прекращается, когда изменения между двумя значениями γ меньше заданной погрешности.

Для решения основного уравнение системы балка-почвы используются два различных метода в следующих главах.

2 РЕШЕНИЕ ОСНОВНОГО ДИФФЕРЕНЦИАЛЬНОГО УРАВНЕНИЯ МЕТОДОМ КОНЕЧНЫХ ЭЛЕМЕНТОВ

2.1 Метод конечных элементов

Метод конечных элементов (МКЭ) — численный метод решения задач прикладной механики [12].

Широко используется для решения задач механики деформируемого твёрдого тела, теплообмена, гидродинамики и электромагнитных полей. С точки зрения вычислительной математики, идея метода конечных элементов заключается в том, что минимизация функционала вариационной задачи осуществляется на совокупности функций, каждая из которых определена на своей подобласти, для численного анализа системы позволяет рассматривать его как одну из конкретных ветвей диакоптики — общего метода исследования систем путём их расчленения. Возникновение метода конечных элементов связано с решением задач космических исследований в 1950-х годах (идея МКЭ была разработана советскими учёными ещё в 1936 году, но из-за неразвитости вычислительной техники метод не получил развитие). Этот метод возник из строительной механики и теории упругости, а уже затем было получено его математическое обоснование. Существенный толчок в своём развитии МКЭ получил в 1963 году после того, как было доказано то, что его можно рассматривать, как один из вариантов распространённого в строительной механике метода Рэлея-Ритца, который путём минимизации потенциальной энергии сводит задачу к системе линейных уравнений равновесия. После того, как была установлена связь МКЭ с процедурой минимизации, он стал применяться к задачам, описываемым уравнениями Лапласа или Пуассона. Область применения МКЭ значительно расширилась, когда было установлено (в 1968 году), что уравнения, определяющие элементы в задачах могут быть легко получены с помощью вариантов метода взвешенных невязок, таких как метод Галёркина, или метод наименьших квадратов. Это сыграло важную роль в теоретическом обосновании МКЭ, так как позволило применять его при решении многих типов дифференциальных уравнений. Таким образом, метод конечных элементов превратился в общий метод численного решения дифференциальных уравнений или систем дифференциальных уравнений.

С развитием вычислительных средств возможности метода постоянно расширяются, также расширяется и класс решаемых задач. Практически все современные расчёты на прочность проводят, используя метод конечных элементов.

Метод конечных элементов (МКЭ) является сеточным методом [13], предназначенным для решения задач микроуровня, для которого модель объекта задается системой дифференциальных уравнений в частных производных с заданными краевыми условиями.

$$\begin{aligned}LV + P &= 0 \\ Y(\Gamma) &= V\Gamma\end{aligned}\tag{2.1}$$

Здесь L - дифференциальный оператор (например, оператор Лапласа), V - фазовая переменная - неизвестная функция, которую следует найти, P - величина, независимая от V . $V(\Gamma) = VT$ - граничное условие первого рода (Дирихле), то есть на границе задано значение фазовой переменной.

Будем искать решение с помощью функции, имеющей следующий вид:

$$V^* = F + \sum_{m=1}^M A_m N_m \quad (2.2)$$

Здесь V^* - приближенное решение, F - функция, удовлетворяющая граничным условиям, N_m - пробные функции, которые на границе области должны быть равны нулю, A_m - неизвестные коэффициенты, которые необходимо отыскать из условия наилучшего удовлетворения дифференциальному оператору, M - количество пробных функций. Если подставить V^* в исходный дифференциальный оператор, то получим невязку, принимающую в различных точках области разное значение.

$$R = LV^* + P \quad (2.3)$$

Необходимо сформулировать условие, позволяющее минимизировать эту невязку по всей области. Одним из вариантов такого условия может быть следующее уравнение:

$$\int_S W_n R dS = 0 \quad (2.4)$$

Здесь W_n – некоторые весовые функции, в зависимости от выбора которых различают варианты метода взвешенных невязок, S – область пространства, в которой ищется решение. При выборе в качестве весовых функций дельта – функций будем иметь метод, который получил название метод поточечной коллокации, для кусочно-постоянных функций – метод коллокации по подобластям, но наиболее распространенным является метод Галеркина, в котором в качестве весовых функций выбираются пробные функции N . В этом случае, если количество пробных функций равно количеству весовых функций, после раскрытия определенных интегралов приходим к замкнутой системе алгебраических уравнений относительно коэффициентов A .

$$KA + Q = 0 \quad (2.5)$$

где коэффициенты матрицы K и вектора Q вычисляются по формулам:

$$K_{ij} = \int_S W_i L N_j dS$$

$$Q_i = \int_S W_i (LF + P) dS \quad (2.6)$$

После нахождения коэффициентов A и подстановки их в начальное уравнение, получаем решение исходной задачи. Недостатки метода взвешенных невязок очевидны: поскольку решение ищется сразу по всей области, то количество пробных и весовых функций должно быть значительным для обеспечения приемлемой точности, но при этом возникают трудности при вычислении коэффициентов K_{ij} и Q_i , особенно при решении плоских и объемных задач, когда потребуется вычисление двойных и тройных интегралов по областям с криволинейными границами. Поэтому на практике этот метод не использовался, пока не был изобретен метод конечных элементов (МКЭ). Идея метода заключается в следующем: в методе взвешенных невязок воспользоваться простыми пробными и весовыми функциями, но не во всей области S , а ее отдельных подобластях (конечных элементах), а точность решения задачи обеспечить использованием большого числа конечных элементов (КЭ), при этом КЭ могут быть простой формы и вычисление интегралов по ним не должно вызывать особых затруднений. Математически переход от метода взвешенных невязок к МКЭ осуществляется с использованием специальных пробных функций, которые также называются глобальными базисными функциями, обладающих следующими свойствами:

- в узле аппроксимации функции имеют значение равное единице;
- отличны от нуля только в КЭ, содержащих этот узел аппроксимации, во всей остальной области равны нулю.

2.2 Граничные условия

В теории дифференциальных уравнений, начальные и граничные условия — дополнение к основному дифференциальному уравнению (обыкновенному или в частных производных), задающее его поведение в начальный момент времени или на границе рассматриваемой области соответственно. Обычно дифференциальное уравнение имеет не одно решение, а целое их семейство. Начальные и граничные условия позволяют выбрать из него одно, соответствующее реальному физическому процессу или явлению. В теории обыкновенных дифференциальных уравнений доказана теорема существования и единственности решения задачи с начальным условием. Для уравнений в частных производных получены некоторые теоремы существования и единственности решений для определенных классов начальных и краевых задач.

Граничные условия на каждом краю могут быть одного из трёх типов:

- жесткая заделка
- шарнир
- свободный край

В зависимости от этих типов будут различные значения будут равны нулю, и приведены в таблице 1.

Таблица 1

Параметры граничных условий

Значения	Условия
$w(0) = 0$	При этом граничном условии, база балки не испытывает никакого изгиба в точке $x=0$.
$w'(0) = 0$	Также предполагается что балка в точке $x=0$ производная от изгиба равна нулю при защемлённом типе соединения.
$w''(L) = 0$	Это граничное условие предполагает что нет никакого изгибающего момента на свободном конце балки.
$w'''(L) = 0$	Это граничное условие предполагает что нет никаких сил сдвига на свободном конце балки.

2.3 Метод Бубнова-Галеркина

Метод успешно применялся замечательным русским кораблестроителем И.Г. Бубновым при расчете обшивки кораблей, и в дальнейшем получил развитие в трудах Б. Г. Галеркина.

Метод Бубнова-Галеркина не относится к вариационным, хотя в некоторых вариантах может интерпретироваться как метод возможных перемещений. Но этот метод является одним из наиболее часто используемых методов расчета в инженерной практике, в частности при расчете пластин.

В основе метода Бубнова–Галеркина лежит понятие ортогональности функций. Две функции $f(x)$ и $\varphi(x)$ называются ортогональными в интервале $a \leq x \leq b$, если выполняется условие

$$\int_a^b f(x) \cdot \varphi(x) dx = 0 \quad (2.7)$$

Например, из теории рядов Фурье известно, что функции $\sin nx$, $\sin mx$ и $\cos nx$, $\cos tx$ при $n \neq m$ и $\sin nx$, $\cos mx$ при произвольных целых n , m ортогональны в интервалах $-\pi \leq x \leq \pi$ и $0 \leq x \leq \pi$. Известны также системы классических ортогональных полиномов.

Кроме понятия ортогональности функций введем понятия линейной независимости и полноты системы функций.

Система функций $\varphi_1(x)$, $\varphi_2(x)$, $\varphi_3(x), \dots, \varphi_n(x)$ называется линейно независимой, если сумма $A_1 \cdot \varphi_1(x) + A_2 \cdot \varphi_2(x) + \dots + A_k \cdot \varphi_k(x) + \dots + A_n \cdot \varphi_n(x) = 0$ при любых x только тогда, когда все коэффициенты A_k ($k = 1, 2, 3, \dots, n$) равны нулю.

Иными словами, система функций считается линейно независимой, если ни одну из функций системы $\varphi_1(x)$, $\varphi_2(x), \dots, \varphi_k(x), \dots, \varphi_n(x)$ нельзя выразить линейной комбинацией других функций этой системы

$$\varphi_k(x) \neq \sum_i A_i \varphi_i(x) \quad (2.8)$$

при произвольных A_i , не все из которых равны нулю.

Если две или несколько функций системы являются линейно зависимыми, то и вся система функций будет линейно зависимой.

Действительно, пусть $\varphi_1(x) = A_2 \varphi_2(x) + \dots + A_k \varphi_k(x)$ $k < m$, тогда суммируя систему функций $\varphi_1(x)$, $\varphi_2(x), \dots, \varphi_k(x), \dots, \varphi_n(x)$ и положив $A_1 = -1$ и нулю все коэффициенты $A_m = 0$ при $t > k$, получим что $-\varphi_1(x) + A_2 \varphi_2(x) + A_2 \varphi_3(x) + \dots + A_k \varphi_k(x) + A_{k+1} \varphi_{k+1}(x) + \dots + A_m \varphi_m(x) = 0$, и, следовательно, система $\varphi_1(x)$, $\varphi_2(x)$, $\varphi_3(x), \dots, \varphi_n(x)$ - линейно зависимая система функций.

То, что принята линейная зависимость первых k членов системы функций, не существенно, так как при конечном числе функций всегда можно поменять порядок их нумерации.

К бесконечным линейно независимым системам функций относятся системы упомянутых выше синусов и косинусов. Другим примером линейно независимых систем функций являются системы полиномов разных порядков. Полиномом n -го порядка называется выражение

$$P_n(x) = a_0 + a_1 x + a_2 x^2 + \dots + a_n x^n = \sum_{k=0}^n a_k x^k \quad a_n \neq 0 \quad (2.9)$$

В частности, x^n нельзя выразить никакой линейной комбинацией полиномов порядка меньше n .

На основе любой линейно независимой системы функций порядка n можно построить другую систему n линейно независимых функций, взяв некоторые линейные комбинации из некоторых функций первоначальной системы. В

частности, можно построить систему линейно независимых, взаимно ортогональных функций. В пособии не рассматривается алгоритм построения ортогональных систем функций. С ним можно познакомиться в курсах математической физики, функционального анализа и некоторых других.

Введенные понятия ортогональности и линейной независимости системы функций имеют аналогию с понятиями ортогональности и линейной независимости векторов, а разложение функции в ряд по этой системе функций - с разложением векторов по линейно независимому базису векторов. В курсе линейной геометрии показывается:

- два непараллельных вектора в плоскости линейно независимы;
- любому вектору в плоскости можно построить вектор ортогональный данному;
- любой вектор в плоскости можно выразить через линейную комбинацию двух линейно независимых векторов;
- в трехмерном пространстве три некопланарных (не лежащих в одной плоскости) вектора являются независимыми;
- двум ортогональным векторам в плоскости можно построить третий, ортогональный им в трехмерном пространстве;
- любой вектор в трехмерном пространстве можно представить линейной комбинацией трех линейно независимых векторов;
- любые три линейно независимых вектора трехмерного пространства (два вектора двухмерного пространства) могут служить базисом этого пространства;
- условием ортогональности векторов является равенство нулю их скалярного произведения

$$(a,b) = a_1b_1 + a_2b_2 + a_3b_3 = \sum_{i=1}^3 a_i b_i = 0, \quad (2.10)$$

где a_i, b_i - коэффициенты разложения векторов в ортогональном базисе трехмерного пространства;

- на основе произвольного базиса пространства можно построить ортогональный базис этого пространства;
- любой вектор трехмерного (двухмерного) пространства можно представить в виде линейной комбинации базиса пространства, которая называется проекциями вектора на базис пространства;
- вектор, не лежащий в плоскости, нельзя восстановить по его линейной комбинации (проекциям) линейно независимых векторов в плоскости;
- двухмерное пространство называется подпространством трехмерного пространства;
- вектор в трехмерном пространстве можно разложить по базису этого пространства - системе трех линейно независимых векторов;
- разложение вектора, не лежащего в плоскости (вектор трехмерного пространства), по базису в плоскости называется проекцией вектора на плоскость или проекцией вектора в подпространство;

- по проекции вектора в подпространство нельзя восстановить вектор, лежащий в пространстве большей размерности;

- в курсах линейной алгебры и функционального анализа обобщают понятие двумерного и трехмерного пространств на понятие n -мерного и бесконечномерного пространств с соответствующими понятиями линейной независимости и ортогональности векторов, базиса и разложения произвольного вектора в базисе пространства и его проекции в подпространство;

- если по разложению произвольного вектора из данного множества векторов в базисе n -мерного пространства нельзя восстановить его первоначальную (точную) величину, то говорят, что данное пространство является неполным для данного множества векторов (они лежат в пространстве большей размерности);

- понятие полноты пространства особенно важно в бесконечномерном пространстве, так как, имея базис n -мерного пространства, всегда можно по определенному алгоритму построить базис пространства с размерностью $n + 1$, то построение по этому алгоритму базиса бесконечномерного пространства предполагает бесконечное число действий. В то же время, наличие бесконечномерного базиса векторного пространства не гарантирует его полноту.

Данное отступление в теорию n -мерных пространств потребовалось, чтобы ввести аналогичные понятия в пространстве (множестве) функций и, в частности, ввести понятие полноты системы функций.

Пусть имеется бесконечная система линейно независимых функций $\varphi_m(x)$ и функция $f(x)$, принадлежащая заданному классу функций. Если функция $f(x)$ в интервале $[a, b]$ может быть представлена в виде ряда

$$f(x) = \sum_{m=1}^{\infty} c_m \varphi_m(x) \quad (2.11)$$

то говорят, что функция $f(x)$ разложена в ряд Фурье по системе функций $\varphi_m(x)$, а само разложение - рядом Фурье функции $f(x)$. Если $\varphi_m(x)$ - система ортогональных функций, то коэффициенты ряда Фурье c_n определяются по формуле

$$c_m = \frac{\int_a^b f(x) \varphi_m(x) dx}{\int_a^b \varphi_m^2(x) dx} \quad (2.12)$$

В случае неортогональной системы функций коэффициенты c_n ряда Фурье определяются из бесконечной системы алгебраических уравнений

$$\sum_{m=1}^{\infty} B_{nm} c_m = d_n \quad (2.13)$$

$$\text{где } B_{nm} = B_{mn} = \int_a^b \varphi_m(x) \varphi_n(x) dx, \quad d_n = \int_a^b f(x) \varphi_n(x) dx.$$

В курсе математического анализа студентам дается теория тригонометрических рядов Фурье. Однако функцию можно раскладывать в ряды не только по системе тригонометрических функций, но и по любой другой полной системе линейно независимых функций. Для этой цели используются, например, функции Бесселя, или системы классических ортогональных полиномов и т.д.

Линейно независимая система функций $\varphi_m(x)$ называется полной на данном множестве функций в интервале $[a,b]$, если при разложении любой функции из этого множества функций в ряд Фурье по системе функций $\varphi_m(x)$ и любой точки $x \in [a,b]$ для точки x' из интервала $(x - \varepsilon < x' < x + \varepsilon)$ и как угодно малых ε и δ найдется такое $N > 0$, что будет выполняться условие

$$\left| f(x) - \sum_{m=1}^N c_m \varphi_m(x') \right| < \delta \quad (2.14)$$

Иными словами, для любой точки x интервала $[a,b]$ найдется точка x' в как угодно малой ε -окрестности точки x , в которой разложение функции $f(x)$ в ряд Фурье при удержании достаточного числа членов ряда N будет как угодно мало отличаться от значения функции в точке x .

Понятие полноты системы функций позволяет раскладывать в ряд Фурье по системе непрерывных функций функции другого класса. Например, раскладывать кусочно-непрерывные (имеющие конечные разрывы в конечном числе точек на данном интервале) функции в ряды по синусам и косинусам - системе непрерывных функций. Так, константу C можно разложить в ряд Фурье по системе функций $\varphi_m(x) = \sin mx$ в интервале $[0, \pi]$. Несмотря на то, что в точках $x = 0$ и $x = \pi$ разложение всегда принимает нулевое значение, т.е. не равным исходной константе C , это разложение дает значения, отличающиеся от единицы внутри интервала $[0, \pi]$ на как угодно малую величину ε , в том числе в точках, как угодно близких к точкам $x = 0$ и $x = \pi$, если в ряде Фурье удерживается достаточное число членов ряда.

Дифференциальные уравнения решаются же так. Пусть дано дифференциальное уравнение

$$L[f(x)] - g(x) = 0 \quad (2.15)$$

где L - некоторый дифференциальный оператор; $g(x)$ - заданная функция. Функция $f(x)$ удовлетворяет граничным условиям, число которых соответствует порядку дифференциального оператора

$$f(a) = A, \quad f(b) = B, \quad f'(a) = A_1, \quad f'(b) = B_1, \dots \quad (2.16)$$

Например, дифференциальное уравнение изгиба балки можно записать в виде

$$L[y(x)] - q(x) = \frac{d^2}{dx^2} \left(EJ_z \frac{d^2 y(x)}{dx^2} \right) - q(x) = 0 \quad (2.17)$$

где $y(x)$ - функция прогиба балки; EJ_z - изгибная жесткость балки (в общем случае переменная); $q(x)$ - распределенная поперечная нагрузка.

Для дифференциального уравнения изгиба балки, порядок которого равен четырем, задается четыре граничных условия. Так, если дана однопролетная шарнирно опертая балка с длиной пролета l , то граничные условия имеют вид:

$$y(0) = 0, \quad M_z(0) = 0, \quad y(l) = 0, \quad M(l) = 0, \quad \text{или в перемещениях}$$

$$y(0) = y''(0) = y(l) = y''(l) = 0.$$

Принимаем решение дифференциального уравнения в виде ряда

$$f(x) = \sum_{m=1}^{\infty} A_m f_m(x) \quad (2.18)$$

где $f_m(x)$ - произвольно выбранные функции, удовлетворяющие граничным условиям (2.2), но не удовлетворяющие в общем случае дифференциальному уравнению (2.1); A_m - произвольные коэффициенты. Тогда, в общем случае, ряд (2.4) не удовлетворяет дифференциальному уравнению (2.1)

$$L[f(x)] - g(x) = L \left[\sum_{m=1}^{\infty} A_m f_m(x) \right] - g(x) =$$

$$= \sum_{m=1}^{\infty} A_m L[f_m(x)] - g(x) = F(x) \neq 0. \quad (2.19)$$

Очевидно, что невязка полученного решения (отличие от нуля) существенно зависит от значений неопределенных коэффициентов A_m . Потребуем выполнения условий ортогональности функции $F(x)$ некоторой полной системе линейно независимых функций $\varphi_n(x)$

$$\int_a^b F(x) \varphi_n(x) dx = \int_a^b \left\{ \sum_{m=1}^{\infty} A_m L[f_m(x)] - g(x) \right\} \varphi_n(x) dx =$$

$$= \sum_{m=1}^{\infty} A_m \cdot \int_a^b L[f_m(x)] \cdot \varphi_n(x) dx - \int_a^b g(x) \cdot \varphi_n(x) dx = \sum_{m=1}^{\infty} B_{nm} A_m - C_n = 0, \quad (2.20)$$

где $B_{nm} = \int_a^b L[f_m(x)] \cdot \varphi_n(x) dx$, $C_n = \int_a^b g(x) \cdot \varphi_n(x) dx$, $n = 1, 2, 3 \dots$

Таким образом, получена система алгебраических уравнений

$$\sum_{m=1}^{\infty} B_{nm} A_m = C_n \quad n = 1, 2, 3, \quad (2.21)$$

из решения которой определяются коэффициенты A_m , такие что функция $F(x)$ ортогональна полной системе линейно независимых функций и, следовательно она тождественно равна нулю

$$F(x) = L[f(x)] - g(x) = L\left[\sum_{m=1}^{\infty} A_m f_m(x)\right] - g(x) = 0 \quad (2.22)$$

т.е. решение дифференциального уравнения (4.2.1) в виде ряда (2.4) с коэффициентами, определяемыми из системы уравнений (2.7), удовлетворяет дифференциальному уравнению и граничным условиям и, следовательно, является решением исходной задачи.

Так как система алгебраических уравнений в общем случае является полной бесконечной системой, решение которой в общем виде возможно лишь при определенных соотношениях между коэффициентами системы, то на практике обычно ограничиваются конечным числом членов ряда, получая таким образом приближенное решение.

Более простое решение получается, если системы функций $L[f_m(x)]$ и $\varphi_n(x)$ взаимно ортогональны в интервале интегрирования. Тогда метод Бубнова–Галеркина приводит к системе независимых алгебраических уравнений для каждого неизвестного коэффициента A_m и, следовательно, можно получить точное решение задачи. На практике довольно часто в качестве функций $f_m(x)$ и $\varphi_n(x)$ принимают одни и те же функции $f_m(x) = \varphi_n(x)$, но даже если сами функции системы $\varphi_n(x)$ взаимно ортогональны, эта система может оказаться не ортогональной после воздействия оператора L на функции системы, т.е.

$$\int_a^b \varphi_m(x) \varphi_n(x) dx \Big|_{m \neq n} = 0 \quad \int_a^b L[\varphi_m] \cdot \varphi_n(x) dx \Big|_{m \neq n} \neq 0 \quad (2.23)$$

В то же время, применение различных систем - системы функций $f_m(x)$, удовлетворяющих граничным условиям задачи, и полной системы линейно независимых функций $\varphi_n(x)$ бывает удобным, так как часто для удовлетворения граничных условий приходится применять системы довольно сложные функций.

Для применения данного метода для решения задач теории упругости Рассмотрим алгоритм решения дифференциальных уравнений в трехмерном и двухмерном пространствах на примере решения пространственной и плоской задач теории упругости.

Пространственная задача

Решение задачи теории упругости в перемещениях принимаем в виде рядов, аналогичных рядам, используемым при решении задач методом Ритца–Тимошенко:

$$u(x, y, z) = u_0(x, y, z) + \sum_{m=1}^{\infty} A_m u_m(x, y, z) \quad (2.24)$$

$$v(x, y, z) = v_0(x, y, z) + \sum_{m=1}^{\infty} B_m v_m(x, y, z)$$

$$w(x, y, z) = w_0(x, y, z) + \sum_{m=1}^{\infty} C_m w_m(x, y, z)$$

где $u_0(x, y, z)$, $v_0(x, y, z)$, $w_0(x, y, z)$ - функции, удовлетворяющие неоднородным граничным условиям; $u_m(x, y, z)$, $v_m(x, y, z)$, $w_m(x, y, z)$ - функции, удовлетворяющие однородным граничным условиям на поверхности; A_m , B_m , C_m - неизвестные коэффициенты, которые определяются на основе метода Бубнова–Галеркина из условия ортогональности системы уравнений равновесия теории упругости в перемещениях полной системе линейно независимых функций $\varphi_n(x)$.

В методе Бубнова–Галеркина функции $u_0(x, y, z)$, $v_0(x, y, z)$, $w_0(x, y, z)$ должны удовлетворять всем граничным условиям задачи, а не только кинематическим, а функции $u_m(x, y, z)$, $v_m(x, y, z)$, $w_m(x, y, z)$ - всем соответствующим однородным граничным условиям задачи.

Чтобы получить алгоритм решения пространственной задачи теории упругости запишем уравнения равновесия в перемещениях в операторной форме

$$\begin{cases} L_{11}u(x, y, z) + L_{12}v(x, y, z) + L_{13}w(x, y, z) + X = 0, \\ L_{21}u(x, y, z) + L_{22}v(x, y, z) + L_{23}w(x, y, z) + Y = 0, \\ L_{31}u(x, y, z) + L_{32}v(x, y, z) + L_{33}w(x, y, z) + Z = 0, \end{cases} \quad (2.25)$$

где

$$L_{11} = (\lambda + \mu) \frac{\partial^2}{\partial x^2} + \mu \nabla^2 = (\lambda + 2\mu) \frac{\partial^2}{\partial x^2} + \mu \frac{\partial^2}{\partial y^2} + \mu \frac{\partial^2}{\partial z^2};$$

$$L_{12} = (\lambda + \mu) \frac{\partial^2}{\partial x \partial y}; \quad L_{13} = (\lambda + \mu) \frac{\partial^2}{\partial x \partial z};$$

$$L_{21} = (\lambda + \mu) \frac{\partial^2}{\partial y \partial x}; \quad L_{22} = \mu \frac{\partial^2}{\partial x^2} + (\lambda + 2\mu) \frac{\partial^2}{\partial y^2} + \mu \frac{\partial^2}{\partial z^2};$$

$$L_{23} = (\lambda + \mu) \frac{\partial^2}{\partial y \partial z}; \quad L_{31} = (\lambda + \mu) \frac{\partial^2}{\partial z \partial x};$$

$$L_{32} = (\lambda + \mu) \frac{\partial^2}{\partial z \partial y}; \quad L_{33} = \mu \frac{\partial^2}{\partial x^2} + \mu \frac{\partial^2}{\partial y^2} + (\lambda + 2\mu) \frac{\partial^2}{\partial z^2}.$$

При подстановке решения (2.25) в уравнения равновесия в общем случае произвольных коэффициентов A_m , B_m , C_m уравнения равновесия не будут выполняться. Коэффициенты A_m , B_m , C_m определим из условий ортогональности решения системы дифференциальных уравнений равновесия полной системе линейно независимых функций.

Пусть $\varphi_n(x, y, z)$ - полная система линейно независимых функций в области, определяемой границами рассматриваемого тела. Тогда из условия ортогональности уравнений равновесия, определяемого решением (2.25),

полной системе линейно независимых функций, получим (здесь и далее аргументы функций для простоты опускаем)

$$\begin{aligned} \iiint_{\Omega} [L_{11}u + L_{12}v + L_{13}w] \cdot \varphi_n d\Omega = \iiint_{\Omega} [L_{11}u_0 + L_{12}v_0 + L_{13}w_0] \cdot \varphi_n d\Omega + \\ + \iiint_{\Omega} \left[L_{11} \sum_{m=1}^{\infty} A_m u_m + L_{12} \sum_{m=1}^{\infty} B_m v_m + L_{13} \sum_{m=1}^{\infty} C_m w_m + X \right] \cdot \varphi_n d\Omega \end{aligned} \quad (2.26)$$

Интегрируя ряды почленно и перенося интегралы от известных функций в правую часть равенства, получаем

$$\begin{aligned} \sum_{m=1}^{\infty} \iiint_{\Omega} [A_m L_{11} u_m + B_m L_{12} v_m + C_m L_{13} w_m] \cdot \varphi_n d\Omega = \\ = - \iiint_{\Omega} [L_{11} u_0 + L_{12} v_0 + L_{13} w_0] \cdot \varphi_n d\Omega - \iiint_{\Omega} X \cdot \varphi_n d\Omega \end{aligned} \quad (2.27)$$

Выполняя аналогичные действия с другими уравнениями равновесия в результате интегрирования, получаем систему алгебраических уравнений

$$\begin{cases} \sum_{m=1}^{\infty} (a_{nm}^1 A_m + b_{nm}^1 B_m + c_{nm}^1 C_m) = d_n^1, \\ \sum_{m=1}^{\infty} (a_{nm}^2 A_m + b_{nm}^2 B_m + c_{nm}^2 C_m) = d_n^2, \\ \sum_{m=1}^{\infty} (a_{nm}^3 A_m + b_{nm}^3 B_m + c_{nm}^3 C_m) = d_n^3, \end{cases} \quad n = 1, 2, 3, \dots \quad (2.28)$$

где $a_{nm}^k = \iiint_{\Omega} (L_{k1} u_m) \cdot \varphi_n d\Omega$; $b_{nm}^k = \iiint_{\Omega} (L_{k2} v_m) \cdot \varphi_n d\Omega$; $c_{nm}^k = \iiint_{\Omega} (L_{k3} w_m) \cdot \varphi_n d\Omega$;

$$d_n^k = - \iiint_{\Omega} [L_{k1} u_0 + L_{k2} v_0 + L_{k3} w_0] \cdot \varphi_n d\Omega - \iiint_{\Omega} q_k \cdot \varphi_n d\Omega;$$

$$q_1 = X; \quad q_2 = Y; \quad q_3 = Z. \quad (2.29)$$

Здесь $k = 1, 2, 3$ - номера уравнения равновесия.

Система алгебраических уравнений и коэффициенты системы (2.29) получены в предположении, что используется одна система линейно независимых функций $\varphi_n(x, y, z)$ для всех уравнений равновесия теории упругости в перемещениях. Можно использовать для каждого из уравнений равновесия свою систему линейно независимых функций $\varphi_{kn}(x, y, z)$ ($k = 1, 2, 3$). Использование двух или трех полных систем линейно независимых функций может оказаться более удобным в зависимости от вида функций $u_m(x, y, z)$, $v_m(x, y, z)$, $w_m(x, y, z)$, применяемых для удовлетворения граничных условий. В частности, если системы функций $u_m(x, y, z)$, $v_m(x, y, z)$, $w_m(x, y, z)$,

удовлетворяющие граничным условиям задачи, каждая из которых является полной системой линейно независимых функций, они могут использоваться и в качестве систем функций для ортогонализации системы уравнений равновесия при определении коэффициентов A_m, B_m, C_m решения (3.1). В этом случае обычно принимают соответственно:

$\varphi_{1n}(x,y,z) = u_n(x,y,z); \quad \varphi_{2n}(x,y,z) = v_n(x,y,z); \quad \varphi_{3n}(x,y,z) = w_n(x,y,z)$, а коэффициенты системы алгебраических уравнений определяются по формулам:

$$\begin{aligned} a_{nm}^k &= \iiint_{\Omega} (L_{k1}u_m) \cdot u_n d\Omega \\ b_{nm}^k &= \iiint_{\Omega} (L_{k2}v_m) \cdot u_n d\Omega \\ c_{nm}^k &= \iiint_{\Omega} (L_{k3}w_m) \cdot u_n d\Omega \\ d_n^k &= -\iiint_{\Omega} [L_{k1}u_0 + L_{k2}v_0 + L_{k3}w_0] \cdot u_n d\Omega - \iiint_{\Omega} q_k \cdot u_n d\Omega \end{aligned} \quad (2.30)$$

Метод Бубнова–Галеркина в случае, когда для решения и его ортогонализации используются одни и те же системы линейно независимых функций $\varphi_{1n}(x,y,z) = u_n(x,y,z)$, $\varphi_{2n}(x,y,z) = v_n(x,y,z)$, $\varphi_{3n}(x,y,z) = w_n(x,y,z)$, можно интерпретировать как метод возможных перемещений и отнести к разряду вариационных методов.

Отметим также, что, как и при решении в рядах другими методами, решение методом Бубнова–Галеркина приводит к бесконечной системе алгебраических уравнений, и, чтобы избежать трудностей с их решением, на практике ограничиваются решением с конечным числом членов ряда, получая приближенные решения. Перед пользователем в этом случае стоит проблема оценки точности получаемого решения, что является достаточно сложной задачей и в данном пособии не рассматривается.

В случае, если удастся подобрать систему функций $\varphi_{in}(x,y,z)$, ортогональных к системе уравнений равновесия твердого деформируемого тела, т.е. если $a_{mn}^f = b_{mn}^f = c_{mn}^f = 0$ при $m \neq n$ ($f = u, v, w$), бесконечная система распадается на систему отдельных уравнений (или группы систем из трех алгебраических уравнений). В этом случае можно говорить, что получено точное решение задачи, так как расчет может быть проведен с необходимым для точности решения числом членов ряда.

Плоская задача

При решении плоской задачи теории упругости используются дифференциальные уравнения в перемещениях, которые в операторном виде имеют вид:

$$\begin{cases} L_{11}u(x,y) + L_{12}v(x,y) + \frac{2(1-\nu_1^2)}{E_1}X(x,y) = 0, \\ L_{21}u(x,y) + L_{22}v(x,y) + \frac{2(1-\nu_1^2)}{E_1}Y(x,y) = 0, \end{cases} \quad (2.31)$$

где $L_{11..} = 2\frac{\partial^2 \dots}{\partial x^2} + (1-\nu_1)\frac{\partial^2 \dots}{\partial y^2}$; $L_{12..} = L_{21..} = (1+\nu_1)\frac{\partial^2 \dots}{\partial x \partial y}$; $L_{22..} = (1-\nu_1)\frac{\partial^2 \dots}{\partial x^2} + 2\frac{\partial^2 \dots}{\partial y^2}$.

Решение для функций перемещений принимается в виде рядов

$$\begin{cases} u(x,y) = u_0(x,y) + \sum_{m=1}^{\infty} A_m u_m(x,y), \\ v(x,y) = v_0(x,y) + \sum_{m=1}^{\infty} B_m v_m(x,y), \end{cases} \quad (2.32)$$

где $u_0(x,y)$, $v_0(x,y)$ - функции, удовлетворяющие неоднородным граничным условиям задачи; $u_m(x,y)$, $v_m(x,y)$ - функции, удовлетворяющие соответствующим однородным граничным условиям; A_m , B_m - неизвестные коэффициенты, определяемые из условий ортогональности уравнений равновесия при подстановке в них решения полной системы линейно независимых функций φ_n .

Выполняя действия, аналогичные проведенным при решении пространственной задачи теории упругости методом Бубнова–Галеркина, получим систему алгебраических уравнений

$$\begin{cases} \sum_{m=1}^{\infty} (a_{nm}^u A_m + b_{nm}^u B_m) = d_n^u, \\ \sum_{m=1}^{\infty} (a_{nm}^v A_m + b_{nm}^v B_m) = d_n^v, \end{cases} \quad n = 1, 2, 3 \dots \quad (2.33)$$

где $a_{nm}^k = \int_A (L_{k1} u_m) \cdot \varphi_n dA$; $b_{nm}^k = \int_A (L_{k2} v_m) \cdot \varphi_n dA$; $d_n^k = - \int_A \left(L_{k1} u_0 + L_{k2} v_0 + \frac{2(1-\nu_1^2)}{E_1} q_k \right) \cdot \varphi_n dA$;
 $k = 1, 2$.

Если для решения задачи и условий ортогонализации используются одни и те же полные системы линейно независимых функций $\varphi_{1n}(x,y) = u_n(x,y)$, $\varphi_{2n}(x,y) = v_n(x,y)$, то коэффициенты системы алгебраических уравнений определяются по формулам:

$$\begin{aligned} a_{nm}^k &= \int_A (L_{k1} u_m) \cdot u_n^k dA \\ b_{nm}^k &= \int_A (L_{k2} v_m) \cdot u_n^k dA \end{aligned} \quad (2.34)$$

$$d_n^k = - \int_A \left(L_{k1} u_0 + L_{k2} v_0 + \frac{2(1-\nu_1^2)}{E_1} q_k \right) \cdot u_n^k dA$$

Здесь $k = 1, 2$ - номер уравнения равновесия плоской задачи теории упругости; $u_n^1 = u_n$, $u_n^2 = v_n$; $q_1 = X$; $q_2 = Y$.

Если рассматривается прямоугольная область плоской задачи, то функции $u_m(x, y)$, $v_m(x, y)$ обычно принимаются в виде произведения функций независимых аргументов:

$$\begin{aligned} u_m(x, y) &= U_{m1}^x(x) \cdot U_{m2}^y(x) \\ v_m(x, y) &= V_{m1}^x(x) \cdot V_{m2}^y(x) \end{aligned} \quad (2.35)$$

где $m = 1, 2, 3, \dots$; $m1 = 1, 2, \dots, m$; $m2 = m - m1 + 1$ ($m1 + m2 = m$).

Иными словами решение ищется в виде двойных рядов:

$$u(x, y) = U_0^x(x) \cdot U_0^y(y) + \sum_{m=1}^{\infty} \sum_{m2=1}^m U_{m1}^x(x) \cdot U_{m2}^y(y) \quad (2.36)$$

Функции $U_{m2}^x(x)$, $V_{m2}^x(x)$ и $U_{m2}^y(x)$, $V_{m2}^y(x)$ удовлетворяют граничным условиям на поперечных ($x = x_1$, $x = x_2$) и на продольных ($y = y_1$, $y = y_2$) краях прямоугольной области соответственно.

Если рассматривается область, ограниченная прямыми x_1, x_2 ($x_1 \leq x \leq x_2$) и криволинейными продольными краями, то функции могут быть представлены в виде:

$$\begin{aligned} u_m(x, y) &= U_{m1}^x(x) \cdot U_{m2}^{xy}(x, y) \\ v_m(x, y) &= V_{m1}^x(x) \cdot V_{m2}^{xy}(x, y) \end{aligned} \quad (2.37)$$

где $U_{m2}^x(x)$, $V_{m2}^x(x)$ - функции, удовлетворяющие граничным условиям на прямолинейных (поперечных) краях области ($x = x_1$ и $x = x_2$); $U_{m2}^{xy}(x, y)$, $V_{m2}^{xy}(x, y)$ - функции, удовлетворяющие граничным условиям на криволинейных краях области.

Если продольные криволинейные края пластинки, ограниченные кривыми $y_1 = r_1(x)$, $y_2 = r_2(x)$, жестко защемлены, и, следовательно, на криволинейных краях пластинки $y = r_1(x)$, $y = r_2(x)$ перемещения равны нулю $u = v = 0$, то функции $U_{m2}^{xy}(x, y)$, $V_{m2}^{xy}(x, y)$ можно принять в виде

$$U_{m2}^{xy}(x, y) = V_{m2}^{xy}(x, y) = \sin m\pi \frac{y - r_1(x)}{r_2(x) - r_1(x)} \quad (2.38)$$

2.4 Формулировка конечного элемента для балки на упругом основании

Предположения и ограничения, такие же, как и в элементарной теории балок, а также с источников [14][15].

- Элемент имеет длину l , и имеет два узла, по одному на каждом конце;
- Элемент соединен с другими элементами только на узлах;
- Нагрузки на элемент происходят только в узлах.

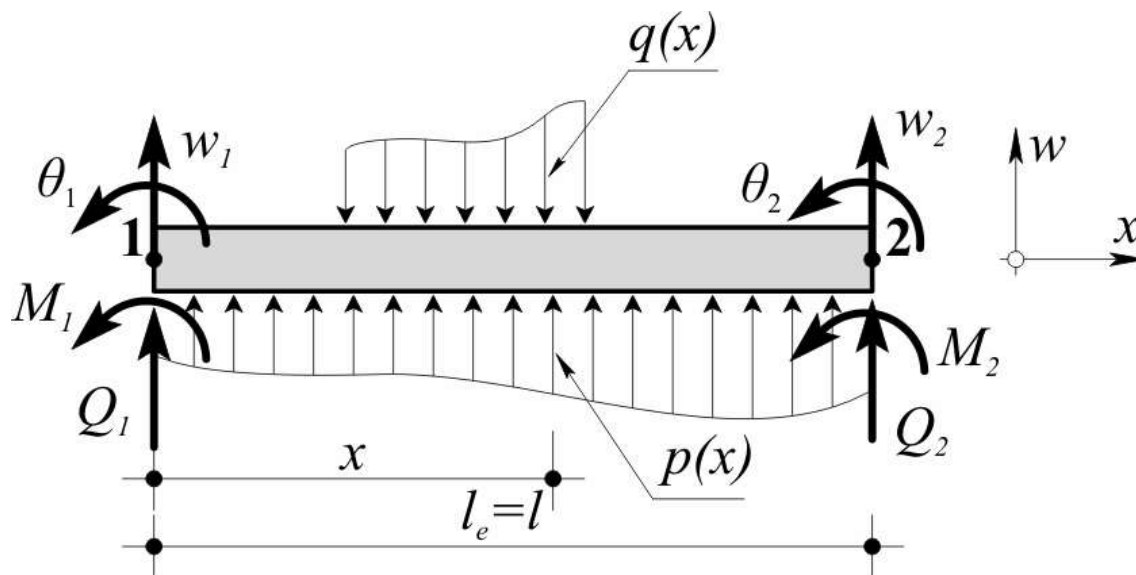


Рис.6 - Нотации в конечном элементе

Рисунок 6 показывает конечный элемент балки на двухпараметрическом упругом основании, где $\{w_d\} = \{w_1 \ \theta_1 \ w_2 \ \theta_2\}^T$ элементы степени свободы, а $\{S_e\} = \{Q_1 \ M_1 \ Q_2 \ M_2\}^T$ являются нагрузками на узлы элемента.

Следует отметить, что Q_1 и Q_2 с вектора нагрузки $\{S_e\}$ не просто поперечные силы сдвига балки; также они включают сопротивление сдвигу связанное с модулем двухпараметрического основания [16]. Сила Q_i ($i=1, 2$) является обобщенной силой сдвига и определяется:

$$Q_i = V_i + V_i^* \quad (2.39)$$

где $V_i = E_b I_b \frac{d^3 w(x)}{dx^3}$ вклад обычный срезающей силы из элементарной теории балок, $V_i^* = -k_1 \frac{dw(x)}{dx}$ вклад срезающей силы от двухпараметрического упругого основания (отрицательный знак возникает потому что, положительное отклонение требует противоположных поперечных сил в основании).

Учитывая четыре граничных условий и одномерность характера проблемы с точки зрения независимой переменной, предполагаем вид функции смещения:

$$w_e(x) = a_0 + a_1 x + a_2 x^2 + a_3 x^3 \quad (2.40)$$

Выбор кубической функции для описания смещения не является произвольным. С учетом того, что было указано четыре граничных условий,

можно определить не более четырех констант в предполагаемой функции смещения. Вторая производная предполагаемой функции перемещения $w_e(x)$ линейна, следовательно, изгибающий момент изменяется линейно по длине элемента. Это соответствует предположению, что нагрузки применяются только на узлах элемент.

Применяя граничные условия и находя значения для констант уравнения (2.40), а затем подставляя результаты обратно в уравнение (2.40) можно получить интерполяционную форму смещения

$$w_e(x) = N_1(x)w_1 + N_2(x)\theta_1 + N_3(x)w_2 + N_4(x)\theta_2 = [N_i]^T \{d_e\} \quad (2.41)$$

где $N_i(x)$, ($i=1, \dots, 4$) являются известными формами функции типа Эрмита, которые описывают распределение смещений узловых значений в векторе узловых перемещений $\{d_e\}$:

$$N_1(x) = 1 - 3\frac{x^2}{l^2} + 2\frac{x^3}{l^3}, N_2(x) = x - 2\frac{x^2}{l} + \frac{x^3}{l^2} \quad (2.42)$$

$$N_3(x) = 3\frac{x^2}{l^2} - 2\frac{x^3}{l^3}, N_4(x) = \frac{x^2}{l} + \frac{x^3}{l^2}$$

Многочлен (2.41) представляет собой приближенное решение основного уравнения (1.11), это дает результат остатка (ошибка или несоответствие):

$$\varepsilon(x) = E_b I_b \frac{d^4 w_e(x)}{dx^4} - k_1 \frac{d^2 w_e(x)}{dx^2} + k_s w_e(x) - q(x) \neq 0 \quad (2.43)$$

Сводя к минимуму этот остаток приведет к аннулированию балансирующей функции Галёркина, где вес рассматривается для каждой из четырех функций $N_i(x)$. Таким образом, конечные матрицы жесткости выглядят следующим образом [17]:

$$[k_e] = EI \int_0^l N_i''(x)N_j''(x) dx = \frac{E_b I_b}{l^3} \begin{bmatrix} 12 & 6l & -12 & 6l \\ 6l & 4l^2 & -6l & 2l^2 \\ -12 & -6l & 12 & -6l \\ 6l & 2l^2 & -6l & 4l^2 \end{bmatrix} \quad (2.44)$$

$$[k_e^s] = k_s \int_0^l N_i(x)N_j(x) dx = \frac{k_s l}{420} \begin{bmatrix} 156 & 22l & 54 & -13l \\ 22l & 4l^2 & 13l & -3l^2 \\ 54 & 13l & 156 & -22l \\ -13l & -3l^2 & -22l & 4l^2 \end{bmatrix} \quad (2.45)$$

$$[k_e^t] = k_1 \int_0^l N_i'(x)N_j'(x) dx = \frac{k_1}{30l} \begin{bmatrix} 36 & 3l & -36 & 3l \\ 3l & 4l^2 & -3l & -l^2 \\ -36 & -3l & 36 & -3l \\ 3l & -l^2 & -3l & 4l^2 \end{bmatrix} \quad (2.46)$$

где

$$([k_e] + [k_e^s] + [k_e^t])\{d_e\} = \{S_e\} - \{R_e\} \quad (2.47)$$

матричная запись основного дифференциального уравнения (1.11); $[k_e]$ матрица жесткости изгиба балки; $[k_e^s]$ матрица жесткости первого параметра грунта; $[k_e^t]$ матрица жесткости второго параметра грунта; $i, j = 1, \dots, 4$.

Вектор $\{R_e\}$ зависит от распределенной нагрузки, и для равномерной нагрузки $q(x) = q$, принимает вид:

$$\{R_e\} = \left\{ \frac{ql}{2} \quad \frac{ql^2}{12} \quad -\frac{ql}{2} \quad -\frac{ql^2}{12} \right\}^T \quad (2.48)$$

Собрав все матрицы жесткости полученным выше, получим глобальную матрицу жесткости системы балка-основание:

$$[K]\{W\} = \{F\} \quad (2.49)$$

где $[K]$ глобальная матрица жесткости, $\{W\}$ глобальное узловое перемещение и $\{F\} = \{S_e\} - \{R_e\}$ вектор эквивалентной нагрузки на систему. Необходимо применить граничные условия для решения систем уравнений.

2.5 Решение методом конечных элементов

Для решения системы линейных уравнений применим метод Гаусса. Метод Гаусса – классический способ решения системы линейных алгебраических уравнений (СЛАУ). Данный метод исключает последовательно переменные, при помощи элементарных преобразований вся система уравнений сводится к такой же системе треугольного вида, из которой далее, начиная с последних (по индексу), находятся все неизвестные системы [18].

Алгоритм метода Гаусса представлен на рисунке 7.

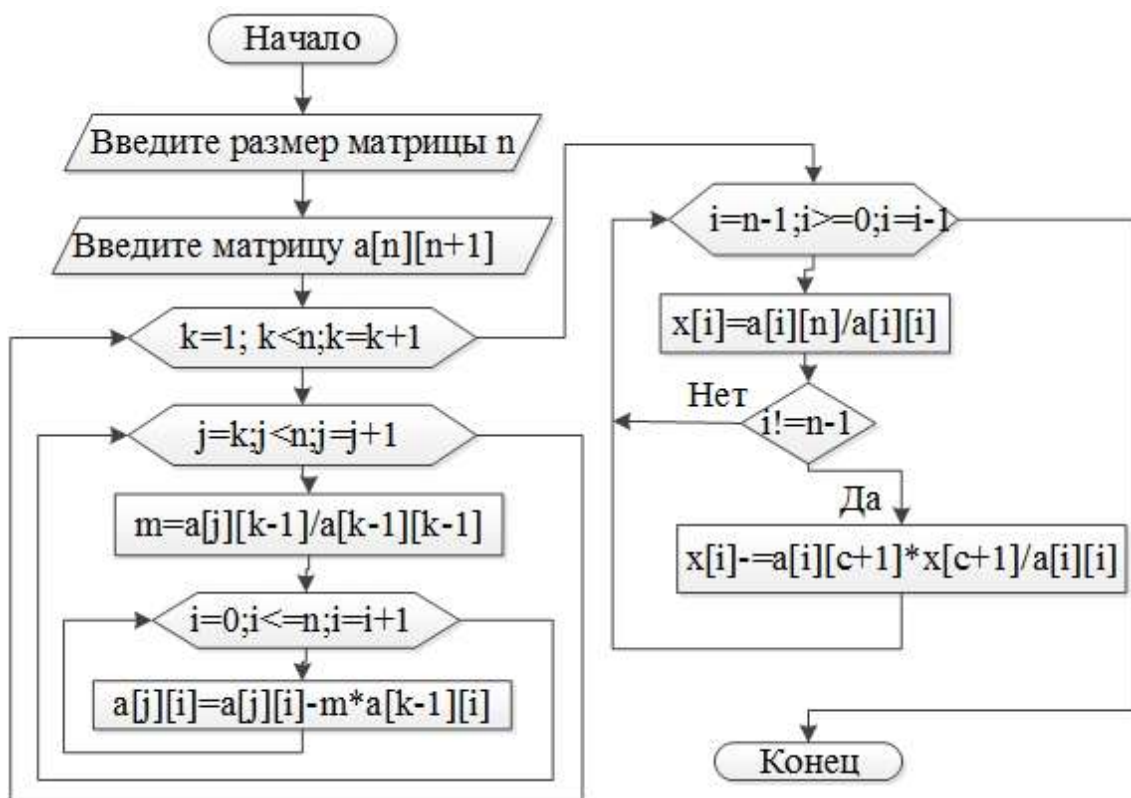


Рис. 7 – Алгоритм Гаусса для решения СЛАУ

Алгоритм решения СЛАУ методом Гаусса подразделяется на два этапа.

На первом осуществляется так называемый прямой ход, когда путём элементарных преобразований над строками систему приводят к ступенчатой или треугольной форме, либо устанавливают, что система несовместна. А именно, среди элементов первого столбца матрицы выбирают ненулевой, перемещают его на крайнее верхнее положение перестановкой строк и вычитают получившуюся после перестановки первую строку из остальных строк, домножив её на величину, равную отношению первого элемента каждой из этих строк к первому элементу первой строки, обнуляя тем самым столбец под ним. После того, как указанные преобразования были совершены, первую строку и первый столбец мысленно вычёркивают и продолжают пока не останется матрица нулевого размера. Если на какой-то из итераций среди элементов первого столбца не нашёлся ненулевой, то переходят к следующему столбцу и проределывают аналогичную операцию.

На втором этапе осуществляется так называемый обратный ход, суть которого заключается в том, чтобы выразить все получившиеся базисные переменные через небазисные и построить фундаментальную систему решений, либо, если все переменные являются базисными, то выразить в численном виде единственное решение системы линейных уравнений. Эта процедура начинается с последнего уравнения, из которого выражают соответствующую базисную переменную (а она там всего одна) и подставляют в предыдущие уравнения, и так далее, поднимаясь по «ступенькам» вверх. Каждой строчке соответствует ровно одна базисная переменная, поэтому на каждом шаге, кроме последнего (самого верхнего), ситуация в точности повторяет случай последней строки.

Метод Гаусса требует $O(n^3)$ арифметических операций.

Достоинствами данного метода являются:

- Для матриц ограниченного размера менее трудоёмкий по сравнению с другими методами.
- Позволяет однозначно установить, совместна система или нет, и если совместна, найти её решение.
- Позволяет найти максимальное число линейно независимых уравнений — ранг матрицы системы.

Используя алгоритм показанный на рисунке 7, был реализован код в среде MATLAB. Язык MATLAB является высокоуровневым интерпретируемым языком программирования, включающим основанные на матрицах структуры данных, широкий спектр функций, интегрированную среду разработки, объектно-ориентированные возможности и интерфейсы к программам, написанным на других языках программирования. Программы, написанные на MATLAB, бывают двух типов — функции и скрипты. Функции имеют входные и выходные аргументы, а также собственное рабочее пространство для хранения промежуточных результатов вычислений и переменных. Скрипты же используют общее рабочее пространство. Как скрипты, так и функции не компилируются в машинный код и сохраняются в виде текстовых файлов. Существует также возможность сохранять так называемые pre-parsed программы — функции и скрипты, обработанные в вид, удобный для машинного исполнения. В общем случае такие программы выполняются быстрее обычных, особенно если функция содержит команды построения графиков. Основной особенностью языка MATLAB являются его широкие возможности по работе с матрицами.

```
function [x] = GaussianEliminate(A,b)
N = length(b);
for column=1:(N-1)
[dummy,index] = max(abs(A(column:end,column)));
index=index+column-1;
temp = A(column,:);
A(column,:) = A(index,:);
A(index,:) = temp;
temp = b(column)
b(column)= b(index);
b(index) = temp;
for row =(column+1):N
d = A(row,column)/A(column,column);
A(row,column:end) = A(row,column:end)-d*A(column,column:end) ;
b(row) = b(row)-d*b(column);
end
end
for row=N:-1:1
```

```
x(row) = b(row);  
for i=(row+1):N  
x(row) = x(row)-A(row,i)*x(i);  
end  
x(row) = x(row)/A(row,row);  
end  
x = x'  
return
```

Данный код может быть вызван в программе с параметрами основной матрицы и матрицы свободных членов, как пример $x = \text{GaussianEliminate}(A, b)$;

3 АНАЛИТИЧЕСКИЙ МЕТОД РЕШЕНИЯ УРАВНЕНИЯ БАЛКИ НА УПРУГОМ ОСНОВАНИИ

3.1 Формулировка проблемы

Вывод уравнений основан на вариации полного потенциального функционала энергии. Этот метод широко используется в работах [19][8][20][21] для постановки задачи или нахождения параметров земляного полотна. Многие авторы используют различные модели для представления поведения балки: классическая теория балки представлена в [22][17][23]; первый порядок теории деформации сдвига балок [24][25]. Предположения и уравнения классической теории балки используются в представленном методе:

- Балка и почвенные материалы линейно упруги, однородны и изотропны;
- Смещения малы по сравнению с толщиной балки;
- Осевые деформации малы по сравнению с размерностями;
- Поперечные нормальные деформации и касательные напряжения пренебрежимо малы;
- Поперечные сечения являются плоскими и перпендикулярно к продольной оси до и после деформации (гипотеза Бернулли).

Положительные направления нагрузки, смещения и внутренние силы совпадают с положительными координатами в соответствии с рисунком 8.

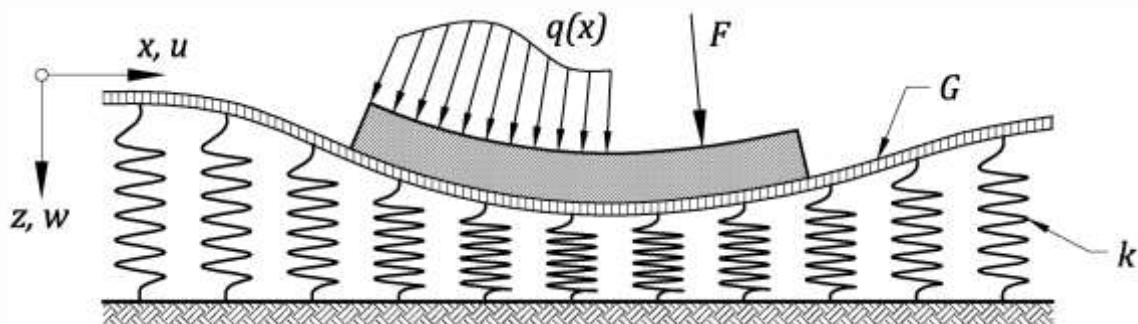


Рис. 8 – Балка на двухпараметрическом упругом основании

Следующие соотношения и уравнения основаны на предыдущих предположениях:

- Поле смещений

$$\begin{aligned} u(x, y, z) &= -z \frac{\partial w}{\partial x} \\ w(x, y, z) &= w(x, z) \end{aligned} \quad (3.1)$$

- Уравнение совместности

$$k \approx \frac{\partial^2 w}{\partial x^2} \quad (3.2)$$

где k является изгибом балки

– Поле деформаций

$$\begin{aligned} \varepsilon_{xx} &= -zk \\ \gamma_{xz} &= 0 \end{aligned} \quad (3.3)$$

– Напряженно-деформированное отношение

$$\sigma_{xx} = -zEk \quad (3.4)$$

где E является модулем упругости балки.

Энергия упругой деформации:

$$U_b = \frac{1}{2} \int_V \sigma : \varepsilon dV = \int_0^l \frac{1}{2} EI \left(\frac{d^2 w}{dx^2} \right)^2 dx \quad (3.5)$$

где I это момент инерции сечения, l длина балки.

Энергия упругой деформации упругого основания:

$$U_f = \int_0^l \frac{1}{2} kbw^2 dx + \int_0^l \frac{1}{2} Gb \left(\frac{dw}{dx} \right)^2 dx \quad (3.6)$$

При рассмотрении одномерной проблемы, ширина деформируемой зоны основания b , должна быть приравнена к ширине балки.

Тогда полная энергия упругой деформации станет:

$$U = U_b + U_f = \frac{1}{2} \int_0^l \left[EI \left(\frac{d^2 w}{dx^2} \right)^2 + Gb \left(\frac{dw}{dx} \right)^2 + kbw^2 \right] dx \quad (3.7)$$

Потенциал нагрузки

$$W = \int_0^l qbw dx \quad (3.8)$$

где q это интенсивность нагрузки.

Полный потенциальный функционал энергии:

$$\Pi(w) = U - W = \frac{1}{2} \int_0^l \left[EI \left(\frac{d^2 w}{dx^2} \right)^2 + Gb \left(\frac{dw}{dx} \right)^2 + kbw^2 \right] dx - \int_0^l qbw dx \quad (3.9)$$

Первая вариация уравнения 3.9 даст:

$$\delta \Pi(w) = \int_0^l EI \frac{d^2 w}{dx^2} \delta \left(\frac{d^2 w}{dx^2} \right) dx + \int_0^l Gb \frac{dw}{dx} \delta \left(\frac{dw}{dx} \right) dx + \int_0^l kbw \delta w dx - \int_0^l qb \delta w dx \quad (3.10)$$

Упрощения уравнения (3.10) даст:

$$\begin{aligned} \delta \Pi(w) = & \left[EI \frac{d^2 w}{dx^2} \delta \left(\frac{dw}{dx} \right) - EI \frac{d^3 w}{dx^3} \delta w + Gb \frac{dw}{dx} \delta w \right]_0^l + \\ & + \int_0^l \left(EI \frac{d^4 w}{dx^4} - Gb \frac{d^2 w}{dx^2} + kbw - qb \right) \delta w dx \end{aligned} \quad (3.11)$$

Экстремум условие для уравнения (3.11) дает $\delta \Pi(w) = 0$ или:

$$\begin{aligned} & \left[EI \frac{d^2 w}{dx^2} \delta \left(\frac{dw}{dx} \right) \right]_0^l + \left[\left(-EI \frac{d^3 w}{dx^3} + Gb \frac{dw}{dx} \right) \delta w \right]_0^l + \\ & + \int_0^l \left(EI \frac{d^4 w}{dx^4} - Gb \frac{d^2 w}{dx^2} + kbw - qb \right) \delta w dx = 0 \end{aligned} \quad (3.12)$$

Первая часть уравнения (3.12) описывает:

– Обязательные граничные условия - $\delta[w]_0^l$ и $\delta[dw/dx]_0^l$;

– Естественные граничные условия - $[EId^2 w/dx^2]_0^l$ и $[-EId^3 w/dx^3 + Gbdw/dx]_0^l$.

Произвольное изменение смещения $\delta w \neq 0$ дает нетривиальное решение экстремальной задачи, которое является уравнение равновесия балки на упругом основании:

$$EI \frac{d^4 w}{dx^4} - Gb \frac{d^2 w}{dx^2} + kbw - qb = 0 \quad (3.13)$$

Вариационная формулировка данной граничной задачи может быть использована для аналитического, численного или приближенного решения этой проблемы.

3.2 Решение граничной задачи

Уравнение (3.13) является обыкновенным дифференциальным уравнением и имеет решение:

$$w(x) = e^{\frac{\sqrt{\frac{bG - \sqrt{b}\sqrt{bG^2 - 4EI k}}{EI}}}{\sqrt{2}} x} C_1 + e^{-\frac{\sqrt{\frac{bG - \sqrt{b}\sqrt{bG^2 - 4EI k}}{EI}}}{\sqrt{2}} x} C_2 + e^{\frac{\sqrt{\frac{bG + \sqrt{b}\sqrt{bG^2 - 4EI k}}{EI}}}{\sqrt{2}} x} C_3 + e^{-\frac{\sqrt{\frac{bG + \sqrt{b}\sqrt{bG^2 - 4EI k}}{EI}}}{\sqrt{2}} x} C_4 + \bar{w}(x) \quad (3.14)$$

где C_1, C_2, C_3 и C_4 произвольные постоянные интегрирования, зависящие от граничных условий и $\bar{w}(x)$ является частным решением, которое определяется нагрузкой.

Решение (3.14) описывает отклонение кривой в непрерывной области балки. Постоянные интегрирования обеспечивают непрерывность деформаций и равновесия балки. Наличие нескольких нагрузок требует деления балки на непрерывные области. На концах этих регионов непрерывность и условия равновесия должны быть удовлетворены. Эти условия будут предоставлять данные для получения постоянных интегрирования для каждого из этих регионов.

С математической точки зрения решение системы обыкновенных дифференциальных уравнений не является проблемой для каждого соответствующего непрерывного региона. Применение граничных условий, условий непрерывности и равновесия дает систему линейных уравнений с неизвестными постоянными интегрирования. Этот метод приводит к необходимости решения большой системы уравнений в присутствии нескольких нагрузок. Недостатком этого метода является сложная математика, но данный метод позволяет решить проблемы с разрывными структурными элементами, такие как: промежуточные опоры или петли, различных сечения балок или модули земляного полотна, и т.д.

Метод начальных параметров, представленный в [26] позволяет избежать решения больших систем уравнений, чтобы получить постоянные интегрирования. Метод предполагает, два из граничных условий на левом конце балки известны. Далее мы переходим к следующей точке нагрузки. Эта нагрузка

воздействует на решение для следующей точки. Добавление эффекта нагрузки на общее решение дает измененную кривую между рассматриваемой и следующей нагрузкой. Этот метод приводит к решению системы только двух уравнений [23].

Перечисленные выше методы приводят к решению систем уравнений, для получения постоянных интегрирования. Применение этих методов для практических задач приводит к сложному решению.

Метод суперпозиции представленный в [22] позволяет избежать этих осложнений. Метод решает простые задач бесконечно-длинных балок с различными простыми нагрузками для построения окончательного решения произвольных балок, нагрузок и опор.

В [27] авторы рассматривают обобщенные решения для балок Эйлера-Бернулли и Тимошенко с разрывом первого рода на упругом основании. Этот метод дает теорию для получения решения многих сложных проблем со структурными разрывами, такие как промежуточные опоры или петли, сечения различных балки или модули земляного полотна, и т.д.

Преобразование Лапласа используется в [28] для решения уравнения равновесия. Использование данного преобразования с ресурсами математической компьютерной программы Maple, приводит к упрощению решения граничной задачи.

Приближенные и численные методы, такие как Ритца, Галёркина, конечных разностей, конечных элементов и дифференциальных квадратур широко используются для решения основных уравнений многими учеными [17][20][19][29]. Все эти методы превосходят по сравнению с аналитическими методами для решения сложных задач с различными граничными условиями или нагрузками. Численные методы также имеют недостатки, такие как: трудно изучить влияние параметров проблемы на решение; необходимость специализированного компьютерной обеспечения для получения решения этой проблемы. Существует несколько коммерческих программ для решения проблем балок на упругом основании методом конечных элементов. Реализация или развитие собственного компьютерного кода требует особых навыков от пользователя.

3.3 Преобразования при помощи сингулярных функций

Описанные ранее аналитические решения вызывают много осложнений, связанных с комплексной нагрузкой. Значительное снижение математической работы может быть достигнуто путем представления нагрузки в виде суммы сингулярных функций. Таким образом, нагрузка представлена одной функцией и функции внутренних сил и прогибов могут быть рассчитаны путем интеграции функции нагрузки. Таким образом, это позволит избежать разделение балки на отдельные области. Идея этого метода представлена Clebsch [30] и стала популярной благодаря Macaulay [31]. Подробное описание данного метода может быть найдено в книге [32].

Представление нагрузки единой функцией может быть выполнено следующими разрывными и сингулярными функциями:

– Разрывные функции - представляют распределенные нагрузки

$$\langle x-a \rangle^n = \begin{cases} 0 & \text{при } x < a \\ (x-a)^n & \text{при } x \geq a \end{cases} \quad n = 0, 1, 2, \dots \quad (3.15)$$

где a это расстояние по оси x к началу разрывной функции. При $n = 0$ функция представляет собой равномерную нагрузку, при $n = 1$ это линейная нагрузка и так далее. Конкретный случай единичной ступенчатой функцией, когда $n = 0$ известна как функции Хевисайда и обозначается как $H(x-a)$.

– Сингулярными функциями - представляют силы и пары

$$\langle x-a \rangle^n = \begin{cases} 0 & \text{при } x \neq a \\ \pm \infty & \text{при } x = a \end{cases} \quad n = -1, -2, -3, \dots \quad (3.16)$$

В случае когда $n = -1$, функция известна как дельта функция Дирака и обозначается как $\delta(x-a)$.

Разрывные функции подчиняются тем же правилам что и обычные функции для интегрирования и дифференцирования

$$\int \langle x-a \rangle^n dx = \frac{1}{1+n} \langle x-a \rangle^{n+1} \quad \text{для } n \geq 0 \quad (3.17)$$

и

$$\frac{d}{dx} \langle x-a \rangle^n = n \langle x-a \rangle^{n-1} \quad (3.18)$$

Использование вышеуказанных функций различные нагрузки могут быть представлены единой функцией. Конечные нагрузки могут быть применены в качестве естественных граничных условий.

3.4 Применение в компьютерных программных обеспечениях

Программы для вычислительной алгебры, такие как MATLAB, Mathcad, Maple и Mathematica являются программными пакетами высокого уровня со встроенными функциями для алгебраических вычислений, включая манипуляции с разрывными и сингулярными функциями.

Предложенный подход для аналитического решения представляет получения уравнений при помощи преобразование нагрузок с сингулярными функциями и решение этих уравнений ресурсами современной компьютерной

алгебры как обыкновенные дифференциальные уравнения. Порядок структурного анализа балки на упругом основании выполняется следующим образом:

- Преобразование нагрузок в виде суммы разрывных и сингулярных функций;
- Расчет общего решения данной граничной задачи;
- Определение граничных условий на концах балки и расчет постоянных
- интегрирования.

Функция *dsolve* в Wolfram Mathematica позволяет решать дифференциальное уравнение для функции, и имеет синтаксис написания: *DSolve[eqn, y, x]*. Также позволяет решать ряд дифференциальных уравнений. Пример простого решения показан на рисунке 9.

```
In[1]:= DSolve[y' [x] + y[x] == a Sin[x], y[x], x]
Out[1]= {{y[x] -> e-x C[1] +  $\frac{1}{2}$  a (-Cos[x] + Sin[x])}}
```

Include a boundary condition:

```
In[2]:= DSolve[{y' [x] + y[x] == a Sin[x], y[0] == 0}, y[x], x]
Out[2]= {{y[x] -> - $\frac{1}{2}$  a e-x (-1 + ex Cos[x] - ex Sin[x])}}
```

Рис. 9 – Пример решения в Wolfram Mathematica

4 РЕЗУЛЬТАТЫ РЕШЕННЫХ ЗАДАЧ

4.1 Распределенная нагрузка на основание Власова

Несмотря на то что, применяемая упрощенная механика сплошной среды, улучшенной модели Власова, на первый взгляд может выглядеть трудоемкой и трудно используемой для замкнутой формы решения, численная модель довольно проста и может быть легко реализована в специфичных приложениях программного обеспечения. Примером такого программного обеспечения может быть среда MATLAB. Использование среды MATLAB позволяет получить результаты для балки на упругом основании для основания модели Власова. Данный метод сравнивается с результатами, полученными с помощью анализа плоской деформации при помощи пакета конечных элементов Plaxis (2Д-МКЭ).

Распределенная равномерная нагрузка $q = 200$ кПа длиной $l = 2$ м лежит на основании, имеющий глубину $H = 4l$, модуль деформации $E_s = 255$ МПа и коэффициент Пуассона $\nu_s = 0.3$. Геометрия модели расширена до $3l$ вне зоны загрузки.

Используя среду MATLAB для построения графиков, полученный результат сравнен с 2Д-МКЭ моделью и представлен в рисунках 10-12. Наиболее важное наблюдение, который можно сделать - медленное рассеивание смещения (рисунок 10а) и напряжения (рисунок 11а и 12а), как в горизонтальном, так и вертикальном направлениях, это происходит потому что в улучшенной модели Власова, совместимость смещение происходит только по вертикали. Тем не менее, как видно, данная модель производит консервативные, но разумные числовые значения для вертикальных перемещений и напряжений основания грунта.

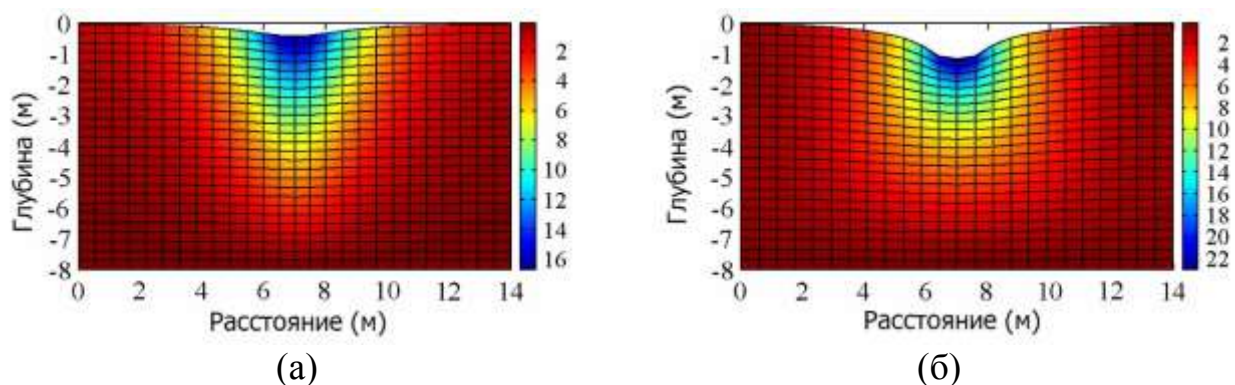


Рис. 10 - Деформированные сетки и вертикальное смещение (мм) через грунт основания. (а) улучшенная модель Власова; (б) 2Д-МКЭ.

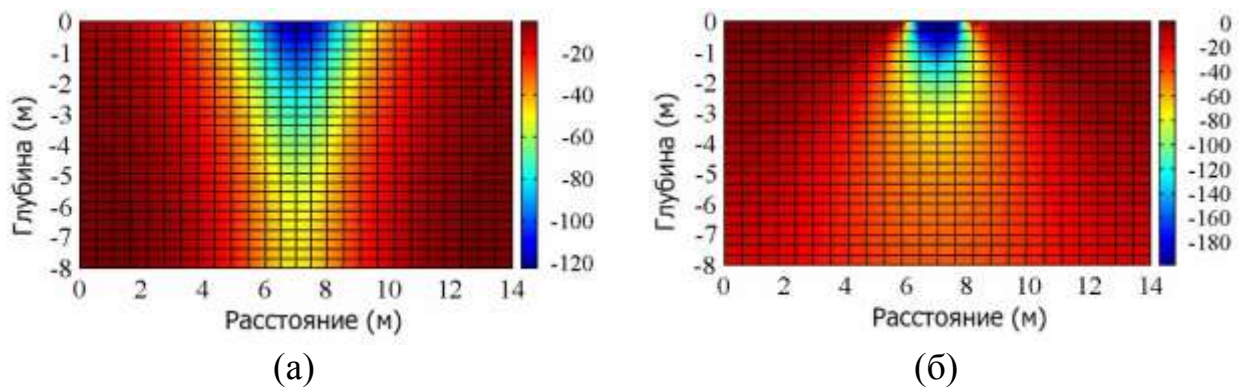


Рис. 11 - Вертикальные напряжения (кПа) через грунт основания. (а) улучшенная модель Власова; (б) 2Д-МКЭ.

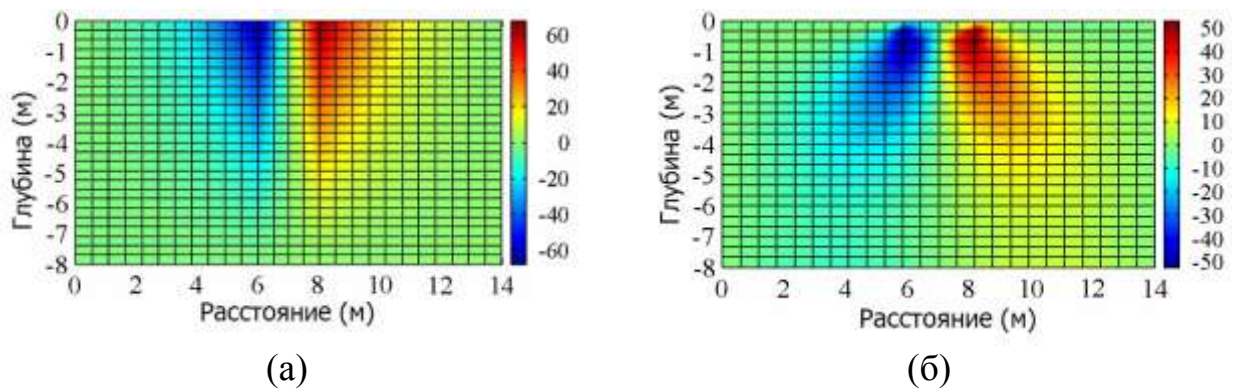


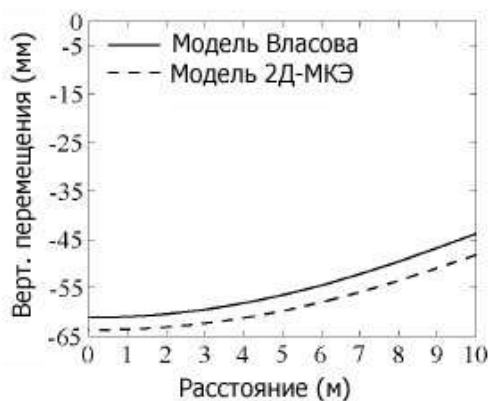
Рис. 12 - Касательные напряжения (кПа) через грунт основания. (а) улучшенная модель Власова; (б) 2Д-МКЭ.

4.2 Непрерывная балка

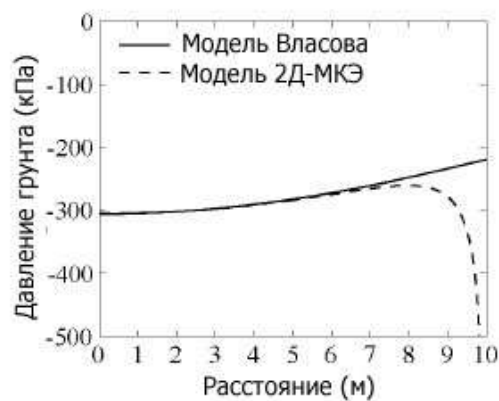
Незакрепленная балка длиной $L = 20$ м, шириной $b = 0,5$ м и высотой $h = 1,0$ м с модулем упругости $E = 27000$ МПа расположена на основании глубиной $H = 5$ м, модуль упругости деформации основания $E_s = 20$ МПа и коэффициент Пуассона $\nu_s = 0,25$. Рассмотрим три случая отдельных нагрузок на балку:

- случай (А), равномерно распределенная нагрузка $Q = 150$ кН/м, действующая вертикально вдоль всей длины балки;
- случай (Б), вертикальная точечная нагрузка $P = 500$ кН на середине балки;
- случай (В), две равные вертикальные точечные нагрузки $P = 250$ кН на концах балки.

Из-за условий симметрии будет анализирована только половина балки.

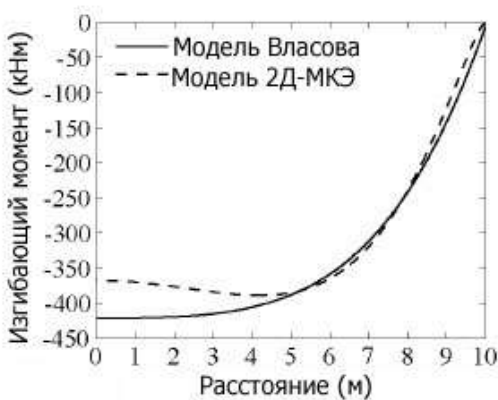


(а)

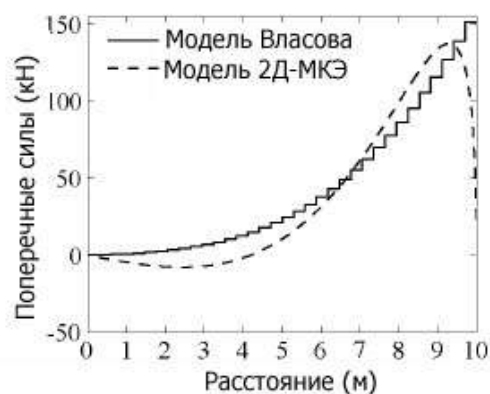


(б)

Рис. 13 – Результаты для случая (А) (а) вертикальные перемещения; (б) реактивное давление грунта



(в)



(г)

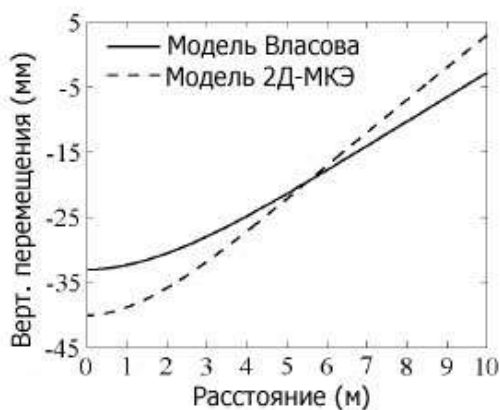
Рис. 14 – Результаты для случая (А) (в) изгибающий момент; (г) поперечные силы

В общей сложности 905 15-узловых треугольных элементов с интерполяцией четвертого порядка для перемещений и двенадцать точек Гаусса для численного интегрирования были использованы для определения сетки в 2Д-МКЭ модели. В обоих случаях балка моделируется с изгибаемым элементом как показано в таблице 2. Используемая геометрия модели распространяется на длину L вне области балки.

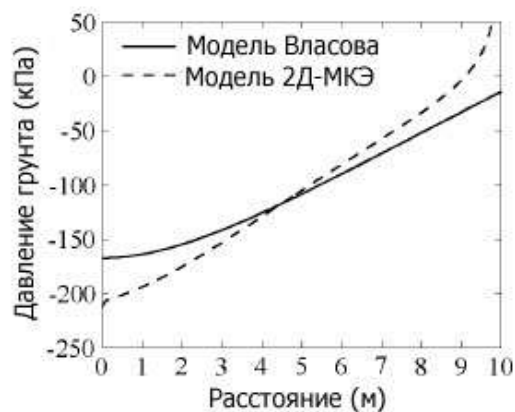
Таблица 2

Количество узлов и элементов при моделировании балки

	Улучшенная модель Власова	2Д-МКЭ модель
Тип элемента	Линейны с двумя точками	Линейный с пятью точками
Количество узлов	35	469
Количество элементов	34	117

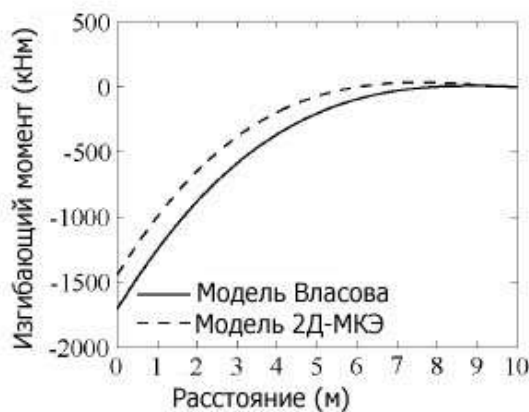


(а)

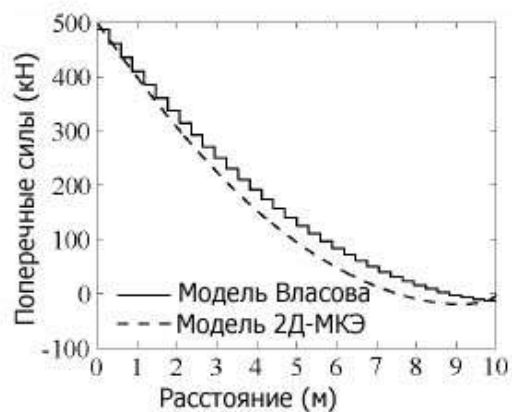


(б)

Рис. 15 – Результаты для случая (Б) (а) вертикальные перемещения; (б) реактивное давление грунта



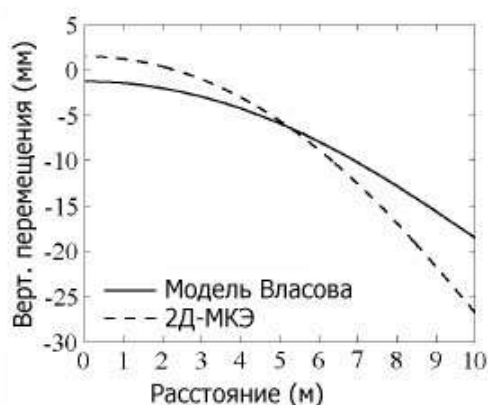
(в)



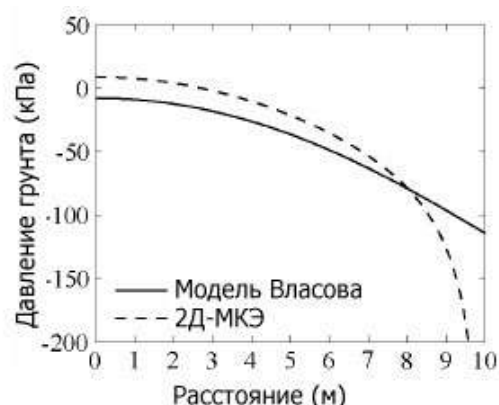
(г)

Рис. 16 – Результаты для случая (Б) (в) изгибающий момент; (г) поперечные силы

Результаты улучшенной модели Власова и 2Д-МКЭ представлены для сравнения на рисунках 13-18 для всех рассматриваемых случаев нагрузки. Можно отметить, что решения для трех случаев нагрузки имеют почти одинаковую форму и сходимость, как и для основания так и для балки. Однако полное сравнение этих двух методов не совсем правильно, потому что в решении 2Д конечных элементов предполагает полную совместимость смещений на границе балки и почвы, когда, как только вертикальное смещение существует в модели Власова.

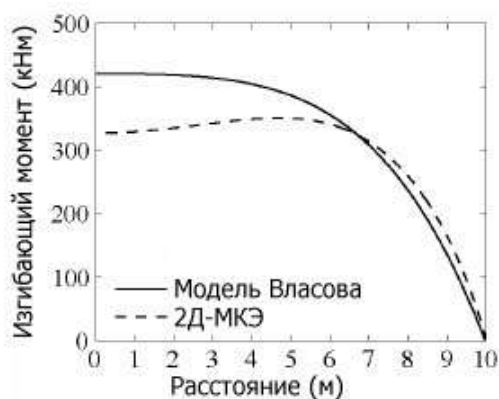


(a)

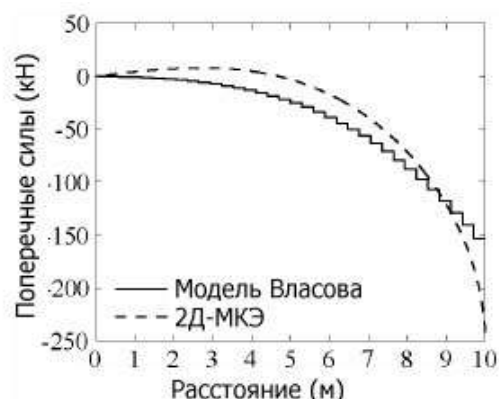


(б)

Рис. 17 – Результаты для случая (В) (а) вертикальные перемещения; (б) реактивное давление грунта



(a)



(б)

Рис. 18 – Результаты для случая (В) (в) изгибающий момент; (г) поперечные силы

Результаты финальных вычисленных значений параметров почвы представлены в таблице 3. Это демонстрирует универсальность метода для решения балки на упругом основании, не имея необходимости устанавливать значения параметров основания.

Таблица 3

Параметры упругого основания

Нагрузка	k_s (кН/м ²)	k_1 (кН)	γ параметр	Количество итераций
Случай А	2400.80	6558.25	0.352	3
Случай Б	2401.57	6515.22	0.418	3
Случай В	2437.25	5953.29	0.953	4

Рассматриваемые примеры показывают что улучшенная модель Власова для основания была применена к статическому структурному анализу балок, опирающихся на упругом основании. Формулировка двухузлового элемента изгиба балки на двухпараметрическом упругом основании, на основе функции

кубической формы регулярного элемента изгиба балки с добавлением эффекта основания в качестве элемента матриц жесткости, также представлены. Чтобы продемонстрировать универсальность представленной модели фундамента, результаты сравнивались с решениями, полученными от метода двумерных конечных элементов плоской деформации (2Д-МКЭ).

В качестве общего замечания, полученное решение несколько консервативно, но достаточно близко к более сложным решениям конечных элементов.

4.3 Примеры, решенные аналитическим решением

Представленные решения классических задач показывают эффективность предложенного подхода. Решения получены с использованием системы компьютерной алгебры Mathematica. Mathematica — система компьютерной алгебры, используемая во многих научных, инженерных, математических и компьютерных областях. Разрабатывается компанией Wolfram Research. Возможности данной программы включают в себя: аналитические преобразования, численные расчеты, теория чисел, линейную алгебру и др. Кроме того, Mathematica — это интерпретируемый язык функционального программирования. Можно сказать, что система Mathematica написана на языке Mathematica, хотя некоторые функции, особенно относящиеся к линейной алгебре, в целях оптимизации были написаны на языке C. Mathematica поддерживает и процедурное программирование с применением стандартных операторов управления выполнением программы (циклы и условные переходы), и объектно-ориентированный подход. Mathematica допускает отложенные вычисления. Также в систему Mathematica можно задавать правила работы с теми или иными выражениями.

4.4 Двухпараметрическая модель на упругом основании

Рассмотрим задачу, данную в [33] с модулем земляного полотна полученным модифицированным методом Власова, $k = 4803.14 \text{ кН/м}^3$ и $G = 13032.44 \text{ кН/м}$. Длина и ширина балки 20 и 0.5 м соответственно, изгибная жесткость балки имеет значение $EI = 1125000 \text{ кНм}^2$. Балка загружена в центре силой $F = 1000 \text{ кН/м}$.

Нагрузка представлена в виду сингулярной функции

$$q(x) = F\delta(x - a) \quad (3.19)$$

где a это расстояние от силы F до левого конца балки. Уравнение равновесие можно записать как:

$$EI \frac{d^4 w}{dx^4} - Gb \frac{d^2 w}{dx^2} + kbw = bF\delta(x - a) \quad (3.20)$$

Естественные граничные условия, используемые для расчета постоянных интегрирования

$$\begin{aligned}
 -EI \frac{d^2 w}{dx^2}(0) &= 0 \\
 -EI \frac{d^3 w}{dx^3}(0) + Gb \frac{dw}{dx}(0) &= 0 \\
 -EI \frac{d^2 w}{dx^2}(l) &= 0 \\
 -EI \frac{d^3 w}{dx^3}(l) + Gb \frac{dw}{dx}(l) &= 0
 \end{aligned} \tag{3.21}$$

В соответствии с основными уравнениями и классической теории балки, функции внутренних силы и реакция земляного полотна представляют:

$$\begin{aligned}
 R(x) = kbw(x) &\rightarrow \text{Реакция основания} \\
 S(x) = Gb \frac{dw(x)}{dx} &\rightarrow \text{Поперечные силы постели грунта} \\
 M(x) = -EI \frac{d^2 w(x)}{dx^2} &\rightarrow \text{Изгибающий момент балки} \\
 V(x) = -EI \frac{d^3 w(x)}{dx^3} &\rightarrow \text{Поперечные силы балки}
 \end{aligned} \tag{3.22}$$

Для решения основного уравнения данной задачи в Wolfram Mathematica необходимо ввести следующий код:

```

DSolve[{1125000*w''''[x] - 6516.22*w''[x] + 2401.57*w[x] == 500 * DiracDelta[x - 10], -1125000*w''[0] == 0, -1125000*w''''[0] + 6516.22*w'[0] == 0, -1125000*w''[20] == 0, -1125000*w''''[20] + 6516.22*w'[20] == 0}, w, x].

```

Полный код приведен в приложениях Б и В. Данный код решает основное дифференциальное уравнение с учетом заданных граничных условий. Функции внутренних сил могут быть найдены при помощи (3.22). Диаграммы реакции основания, и внутренних сил показаны на рисунках 19-21.

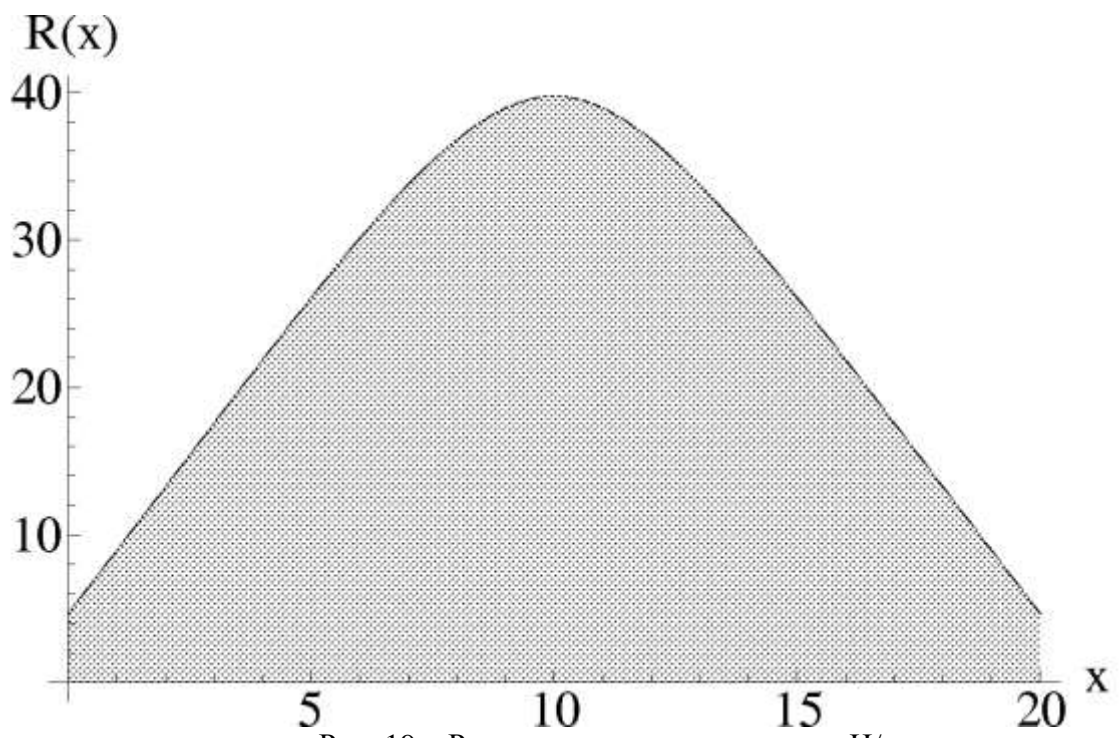


Рис. 19 – Реакция земляного полотна кН/м

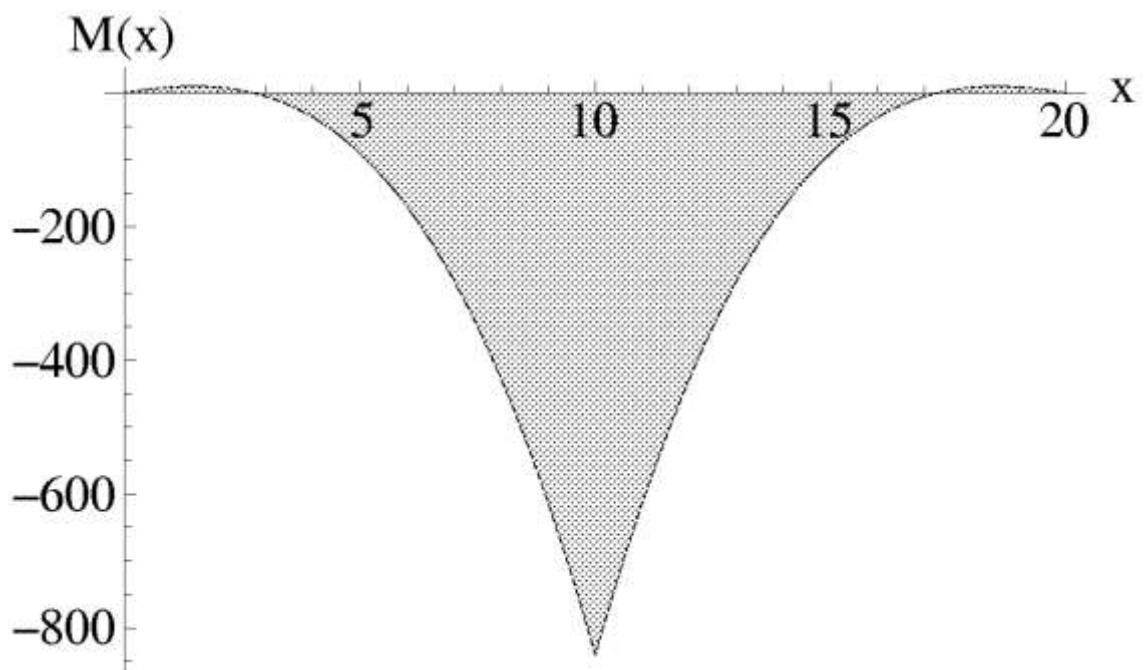


Рис. 20 – Изгибающий момент балки, кНм

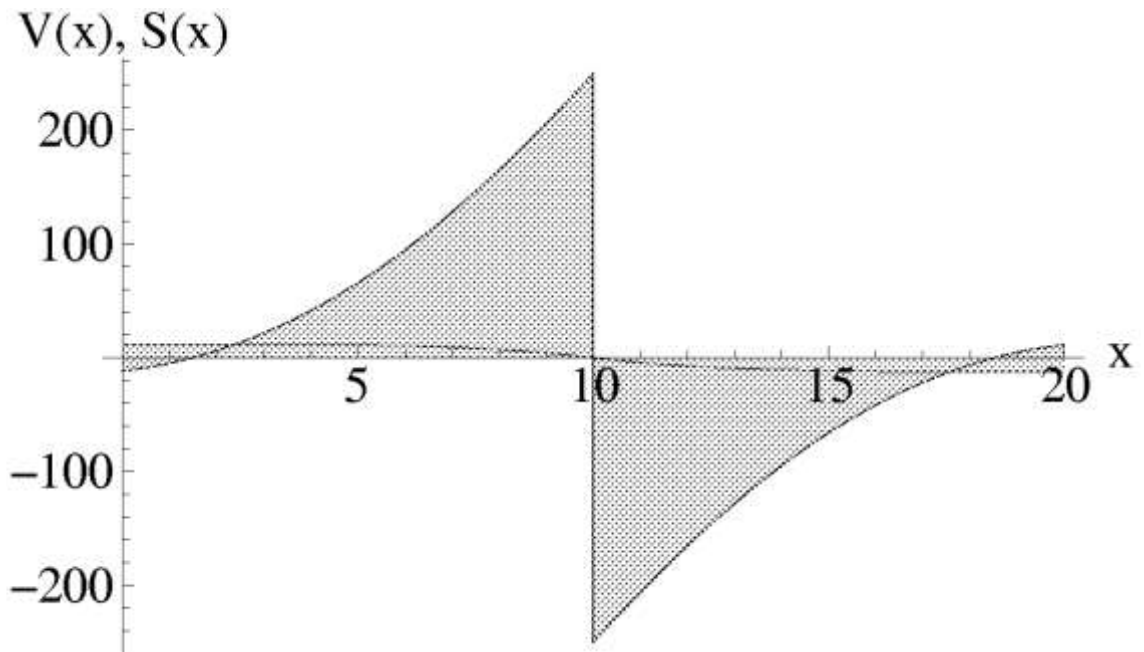


Рис. 21 – Поперечные силы балки и постели грунта, кН

4.5 Балка на Винклеровском основании

Рассмотрим балку со свободными концами с размерами и нагрузками, как показано на рисунке 22 [22]. Следует найти прогиб на концах, а также прогиб, момент, и сдвиг в точке C и середине. Модуль упругости материала балки $E = 1.5 \times 10^6$ фунт/дюйм² и модуль основания $k = 200$ фунт/дюйм³. Балка имеет прямоугольное поперечное сечение с габаритами в 10/8 дюймов.

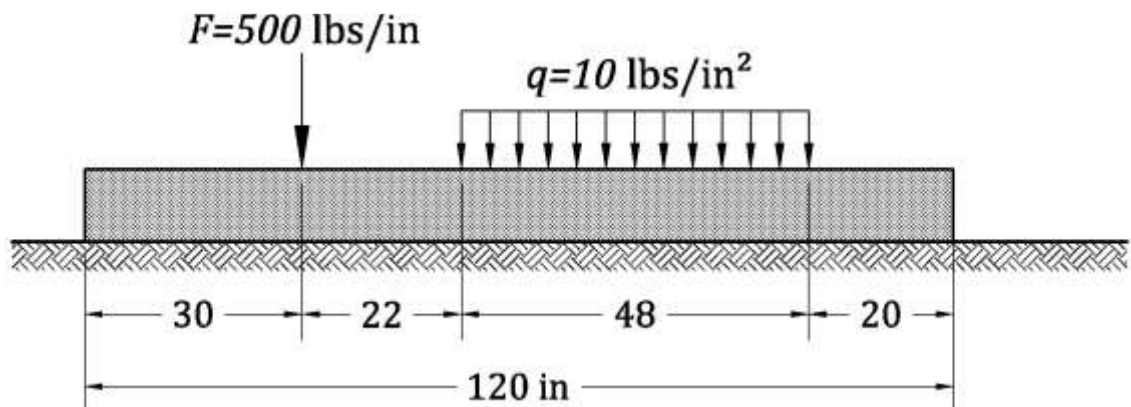


Рис. 22 – Балка с приложенной нагрузкой

Найдем значение $I = 10 \times 8^2 / 12 = 426.7$ дюйм⁴, отсюда $EI = 640000000$. Нагрузки могут быть представлены сингулярными функциями в виде:

$$q(x) = F\delta(x-a) + qH(x-b) - qH(x-c) \quad (3.23)$$

где a это расстояние от силы F к левому концу балки; b это расстояние от левого края балки к началу распределенной нагрузки q ; c это расстояние от левого края

балки до конца распределенной нагрузки. Дифференциальное уравнение можно получить из уравнения (3.20), применив $G = 0$.

$$EI \frac{d^4 w}{dx^4} + kbw = F\delta(x - a) + qH(x - b) - qH(x - c) \quad (3.24)$$

Граничных условия и внутренние силы такие же, как и в предыдущем примере.

Код для системы компьютерной алгебры:

```
DSolve [{640000000*w''''[x] + 2000*w[x] == 5000*DiracDelta[x - 30] +
100*(HeavisideTheta[x - 52]) - 100*(HeavisideTheta[x - 100]), -640000000*w''[0] ==
0, -640000000*w''[120] == 0, -640000000*w''[120] == 0},
w, x].
```

Изгибающие моменты и поперечные силы показаны на рисунках 23 и 24.

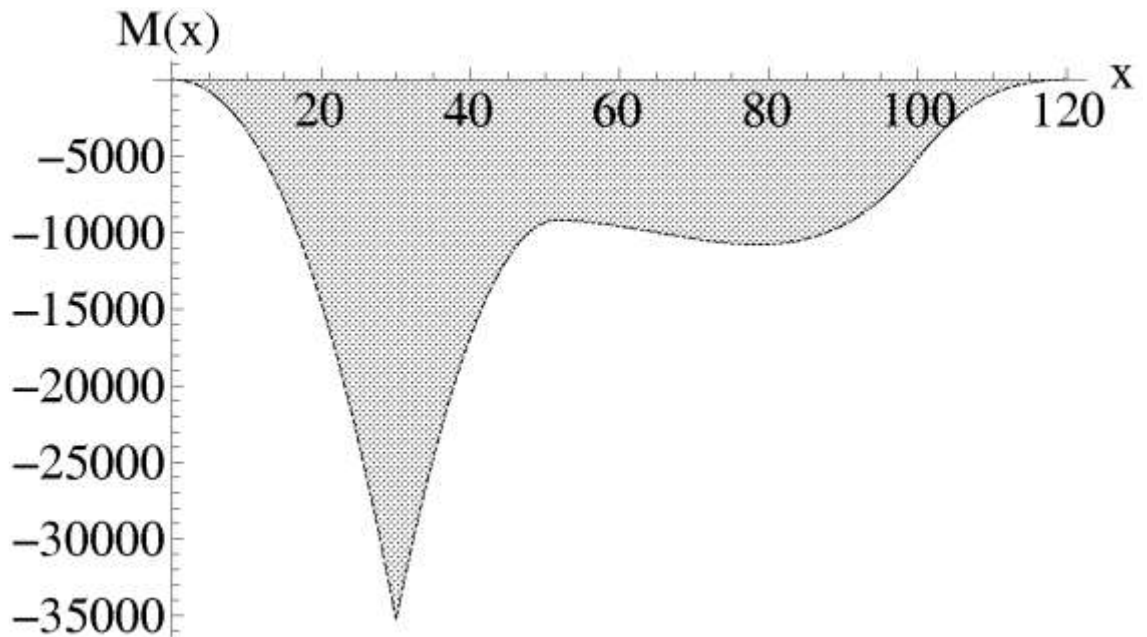


Рис. 23 – Изгибающий момент

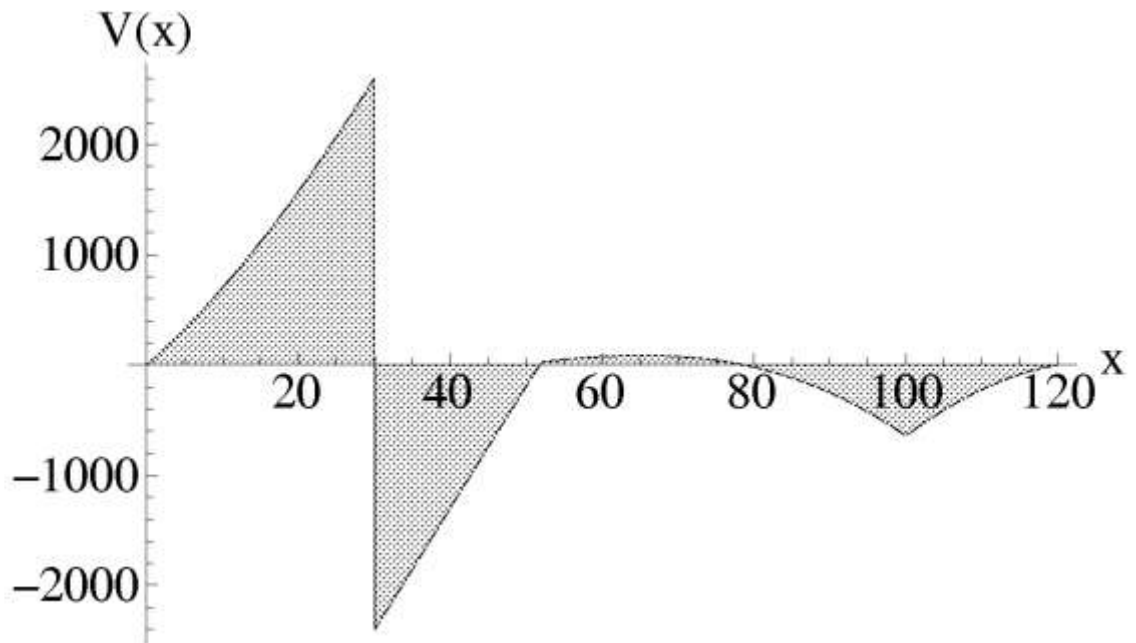


Рис. 22 – Поперечные силы

Изгибающий момент в центре балки ($x = 60$), исходя из рисунка 24 равен 9626.4 lbs , что вполне сопоставимо с результатом из [22] который имеет значение 9623.5 lbs . Их разница составляет 0.03 процента. Результат изгибающего момента и поперечных сил из [22] приведен на рисунке 25.

c. *Bending Moment and Shearing Force at C*

	M_C in. lbs.	Q_C lbs.
On the infinite beam:		
P	-2547.9	-643.2
q	13251.3	630.2
P_{0A}	-1700.0	18.0
M_{0A}	-4.7	-0.5
P_{0B}	1168.2	12.4
M_{0B}	-543.4	58.4
On the beam of finite length:		
	9623.5	75.3

Рис. 25 – Значения изгибающего момента и поперечных сил

Также рассмотрим другой пример для верификации решения [34]. Рассмотрим балку на упругом основании со свободными концами. Геометрические размеры, механические свойства и нагрузки показаны на рисунке 26.

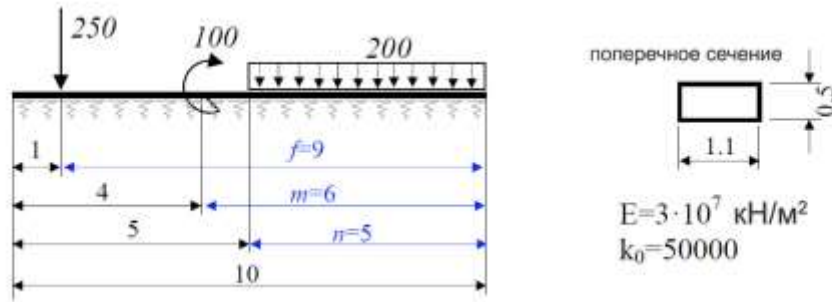


Рис. 26 – Балка с приложенной нагрузкой

Коэффициент основания будет $k = k_0 * 1.1 = 55000$.

Путем сравнения эпюр внутренних сил и реакции грунта, на рисунках 27-29, можно сказать что сравниваемое решение аналогично с уже доступными решениями.

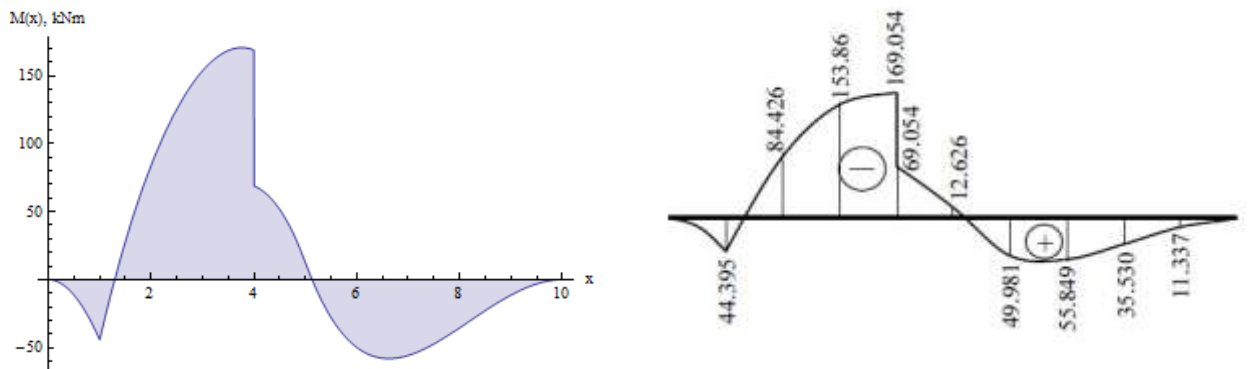


Рис. 27 – Сравнение изгибающего момента

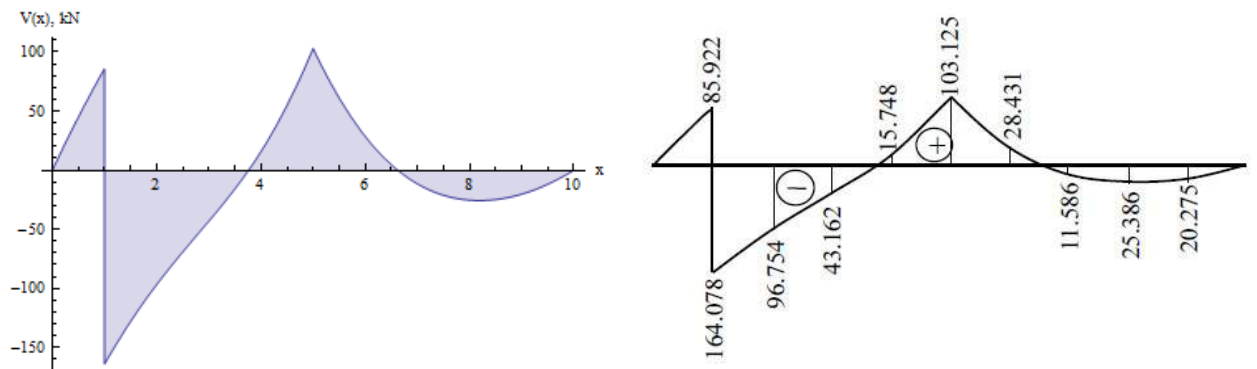


Рис. 28 – Сравнение поперечных сил

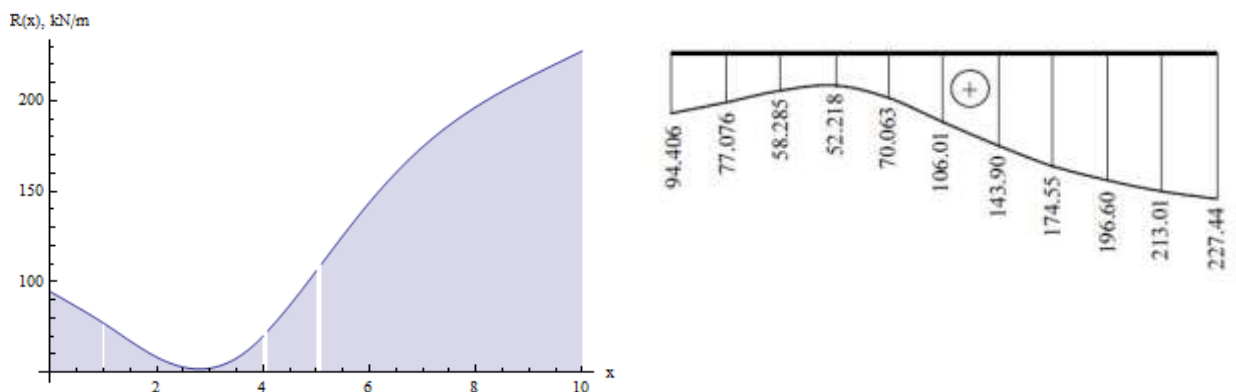


Рис. 29 – Сравнение реакции грунта

Приведенные примеры также решаются коммерческой конечной компьютерных программ элемент SAP2000 и ANSYS. Метод конечных элементов основан на аппроксимации. Когда сетка конечных элементов уточняется полученные результаты достаточно близки к результатам, полученным с помощью аналитического решения. Даже с приближенных результатов, конечные решения элементов имеют много преимуществ как нелинейного моделирования среды почвы, презентации точек разрыва в фундамент и лучевой жесткости, пространственного описания проблемы в грунт-конструкция взаимодействия и т.д.

Приведенные примеры показывают, некоторые из преимуществ предлагаемого подхода для аналитического решения балки на упругом основании:

- Применение различных нагрузках в произвольной точке или области на балки;
 - Применение граничных условий на концах балки;
 - Подход может быть выполнена на короткой, средней или длинной балки;
 - Жесткость балки и грунта может изменяться плавно по длине балки;
 - Реализация подхода в системы компьютерной алгебры является простым и не требуют специальных навыков программирования от пользователя;
 - Представленное решение может быть реализовано в любой математической системе, которые могут решать обыкновенные дифференциальные уравнения;
 - Полученное аналитическое решение может быть использовано для анализа влияния различных параметров на поведение структуры.
- Изложенный подход имеет и некоторые недостатки, такие как:
- Применение различных нагрузок и сложных вариаций балки и жесткости основы усложняет вычисления на компьютерных программах;
 - Структурные разрывы, такие как прыжки в балке или жесткости фундамента жесткости, петли и т.д. не могут быть применены для данного метода;
 - Не возможно связать значения смещений к точка вдоль длины балки.

4.6 Сравнительный анализ двух методов

Таблица 4

Сравнение двух методов

Преимущества МКЭ	Преимущества аналитического метода
<ul style="list-style-type: none"> – Универсальность, возможность решать практически любые краевые задачи; – Быстрая сходимоссть; – Позволяет работать с геометрически более сложными областями; 	<ul style="list-style-type: none"> – Применение различных нагрузках в произвольной точке или области на балки; – Применение граничных условий на концах балки;

<ul style="list-style-type: none"> – Гибкость и разнообразие сеток; – Нерегулярная сетка, дает возможность уменьшить количество вычислений; – Метод является проекционным, то есть устойчив. 	<ul style="list-style-type: none"> – Подход может быть выполнена на короткой, средней или длинной балки; – Жесткость балки и грунта может изменяться плавно по длине балки; – Реализация подхода в системы компьютерной алгебры является простым и не требуют специальных навыков программирования от пользователя; – Представленное решение может быть реализовано в любой математической системе, которые могут решать обыкновенные дифференциальные уравнения; <p>Полученное аналитическое решение может быть использовано для анализа влияния различных параметров на поведение структуры</p>
<p>Недостатки МКЭ</p>	<p>Недостатки аналитического метода</p>
<ul style="list-style-type: none"> – Время, необходимое для расчетов, а также требования к аппаратным средствам компьютера и объему носителей информации в несколько раз превышают аналогичные требования для МКР. – Трудно изучить влияние параметров проблемы на решение; необходимость специализированого компьютерной обеспечения для получения решения этой проблемы. – Реализация или развитие собственного компьютерного кода требует особых навыков от пользователя 	<ul style="list-style-type: none"> – Применение различных нагрузок и сложных вариаций балки и жесткости основы усложняет вычисления на компьютерных программах; – Структурные разрывы, такие как прыжки в балке или жесткости фундамента жесткости, петли и т.д. не могут быть применены для данного метода

ЗАКЛЮЧЕНИЕ

В настоящей работе предлагается техническая теория расчета конструкций балки на упругом основании. Модель Власова с двумя параметрами решается при помощи методов конечных элементов и аналитическим методом.

Формулировка изгиба двух узловой балки на двухпараметрическом упругом основании, основана на кубической функции с учетом параметров основания в виде матрицы жесткости. Данные матрицы могут быть использованы в современных компьютерных программах обеспечения для получения результата. Формулировка двухузловой элемента изгиба балки на двухпараметрическом упругом основании, на основе функции кубической формы регулярного элемента изгиба балки с добавлением эффекта основания в качестве элемента матриц жесткости, также представлены. Чтобы продемонстрировать универсальность представленной модели фундамента, результаты сравнивались с решениями, полученными от метода двумерных конечных элементов плоской деформации (2Д-МКЭ). В качестве общего замечания, полученное решение несколько консервативно, но достаточно близко к более сложным решениям конечных элементов.

Для аналитического решения, приведенные примеры показывают, некоторые из преимуществ предлагаемого подхода для аналитического решения балки на упругом основании:

- Применение различных нагрузок в произвольной точке или области на балки;
- Применение граничных условий на концах балки;
- Подход может быть выполнена на короткой, средней или длинной балки;
- Жесткость балки и грунта может изменяться плавно по длине балки;
- Реализация подхода в системы компьютерной алгебры является простым и не требуют специальных навыков программирования от пользователя;
- Представленное решение может быть реализовано в любой математической системе, которые могут решать обыкновенные дифференциальные уравнения;
- Полученное аналитическое решение может быть использовано для анализа влияния различных параметров на поведение структуры.

Изложенный подход имеет и некоторые недостатки, такие как:

- Применение различных нагрузок и сложных вариаций балки и жесткости основы усложняет вычисления на компьютерных программах;
- Структурные разрывы, такие как прыжки в балке или жесткости фундамента жесткости, петли и т.д. не могут быть применены для данного метода.

Проанализирован механизм взаимодействия балки с массивом и на его основании разработана матрица жесткости изгиба для двух узловой балки на двухпараметрическом упругом основании, что позволяет использовать данную матрицу в современных компьютерных программах для получения результата.

Выполнено аналитическое решение с использованием системы компьютерной алгебры Wolfram Mathematica, при этом сложные нагрузки преобразованы при помощи синуглярных функций, что позволяет упростить решение сложных задач.

Предложен итеративный метод расчета параметра вертикальной деформации почвы с учетом сдвига, исходя из значения параметра сжатия основания и параметра сдвига основания, что дает найти точное значение параметра, характеризующего быстроту затухания осадок по глубине основания.

Проведено сравнение расчета по улучшенной модели Власова с методом конечных элементов плоской деформации и установлено, что скорость сходимости результатов решения предложенным методом в 3-4 раза быстрее.

Разработанные коды можно использовать для дальнейшего анализа балки на упругом основании. Это может быть анализ балки на упругом основании с тремя параметрами, также возможна работа для анализа балки на упругом основании с учетом податливости грунта, и неоднородным упругим основанием.

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

- 1 Балка [Online]. URL: <http://ru.wikipedia.org/wiki/Балка>.
- 2 Филоненко-Бородич М.М. Некоторые приближенные теории грунтового основания // Ученые записки МГУ. 1940. № 46.
- 3 Kerr A.D. Elastic and Viscoelastic Foundation Models // J. Appl. Mech. 1964. Vol. 31, № 3. P. 491–498.
- 4 Пастернак П.Л. Основы нового метода расчета фундаментов на упругом основании при помощи двух коэффициентов постели. Москва: Госстройиздат, 1954.
- 5 Pronk A.C. The Pasternak Foundation: An Attractive Alternative for the Winkler Foundation. // Proc. 5th Int. Conf. Concr. Pavement Des. Rehabil. 1993. P. 349–357.
- 6 Reissner E. Deflection of plates on viscoelastic foundation // J. Appl. Mech. 1958. Vol. 25. P. 144–145.
- 7 Reissner E. Note on The Formulation of the Problem of the Plate on an Elastic Foundation // Acta Mech. № 4. P. 88–91.
- 8 Власов В.З., Леонтьев Н.Н. Балки, плиты и оболочки на упругом основании. Гос. изд-во физико-математической лит-ры, 1960.
- 9 Jones, R. Xenophontos J. The Vlasov Foundation Model // Int. J. Mech. Sci. 1977. Vol. 19. P. 317–323.
- 10 Vallabhan, C. V. G. Das Y.C. Parametric Study of Beams on Elastic Foundations // J. Eng. Mech. Div. 1988. № 114. P. 2072–2082.
- 11 Vallabhan, C. V. G. Das Y.C. Modified Vlasov Model for Beams on Elastic Foundations // J. Geotech. Eng. 1991. № 117. P. 956–966.
- 12 Метод конечных элементов [Online]. URL: http://ru.wikipedia.org/wiki/метод_конечных_элементов.
- 13 Трудоношина, В.А. Уварова М.Ю. Метод конечных элементов.
- 14 Cook R.D. Concepts and applications of finite element analysis. Wiley India Pvt. Limited, 2007.
- 15 Hutton D. V. Fundamentals of Finite Element Analysis. New York: McGraw Hill, 2004.
- 16 Zhaohua, F. Cook D.R. Beam Elements on Two-Parameter Elastic Foundation // J.Eng. Mech. № 109. P. 1390–1402.
- 17 Teodoru, I.B. Muşat, V. Vrabie M. A Finite Element Study of the Bending Behavior of Beams on Two-Parameter Elastic Foundation // Bull. Polytech. Inst. of Jassy. 2009.
- 18 Ильин, В. А. Позняк Э.Г. Линейная алгебра: Учебник для вузов. 6th ed. Москва: ФИЗМАТЛИТ, 2004. P. 280.
- 19 Kolar, V. Nemes I. Modeling of soil-structure interaction // Elsevier. 1989.
- 20 Melerski E.S. Design of beams, circular plates and cylindrical tanks on elastic foundation. 2nd ed. Taylor & Francis Group, 2006.
- 21 Avramidis I.E., Morfidis K. Bending of beams on three-parameter elastic foundation // Int. J. Solids Struct. 2006. Vol. 43, № 2. P. 357–375.
- 22 Hetenyi M. Beams on elastic foundation. University of Michigan Press, 1946.

- 23 Selvadurai A.P.S. Elastic Analysis of soil-foundation interaction // Elsevier. 1979.
- 24 Aydogan M. Stiffness-Matrix Formulation of Beams with Shear Effect on Elastic Foundation // J. Str. Eng. 1995. Vol. 121.
- 25 Yin J.. Closed-Form Solution for Reinforced Timoshenko Beam on Elastic Foundation // J. Eng. Mech. 200AD. Vol. 126. P. 868–874.
- 26 Крылов А.Н. О расчете балок, лежащих на упругом основании. 2nd ed. Ленинград: Академия Наук СССР, 1931.
- 27 Yavari, A. Sarkani, S. Reddy J.N. Generalized solution of beams with jump discontinuities on elastic foundations // Arch. Appl. Mech. 2001. Vol. 71. P. 625–639.
- 28 Betten J. Beam on an Elastic Foundation // Maplesoft Appl. Cent.
- 29 Chen C. Solution of beam on elastic foundation by DQEM // J. Eng. Mech. 1998. Vol. 124.
- 30 Clebsch A. Theorie der Elasticit̄at Fester K̄orper // B.B. Teubner. 1862.
- 31 Macaulay W.H. A note on the deflection of beams // Messenger Math. 1919. Vol. XLVIII.
- 32 Gere J.M. Mechanics of materials. Thomson Learning Inc., 2004.
- 33 Teodoru I.B. EBBEF2p – A Computer Code for Analysing Beams on Elastic Foundations // Intersections. 2009. Vol. 6.
- 34 Lubomirova Parvanova S. Lecture notes: Structural Analysis II [Online]. URL: http://www.uacg.bg/filebank/att_1324.pdf.
- 35 Кишауов К.С., Телегенов К.А. Расчет балки на упругом основании методом конечных элементов. Сборник материалов Международных XIII Байконурских чтений. Жезказган, 2013
- 36 Кишауов К.С., Телегенов К.А. Динамическая устойчивость цилиндрической оболочки с упругим наполнителем (при действии нагрузок и температур). Сборник материалов Международных XIII Байконурских чтений. Жезказган, 2013

Код на MATLAB для решения балки на упругом основании методом конечных элементов.

```

clear all
L=20; % длина, м было 10
LR=[3 3]; % 1-жёсткое защемление, 2-свободное опирание, 3-свободный край
M=[]; % изгибающие моменты, м, кНм [a, load]
F=[[10,-500]]; % сосредоточенные силы, м, кН [a,load]
q=[]; % распределённая нагрузка, м, кН/м [start,end,load]
c=4803000.14; % жёсткость упругого основания, Н/м^3
E=27000; % модуль упругости, МПа
Gr={'жёсткое защемление','свободное опирание','свободный край'};
fprintf('Длина балочки L=%d м\n',L)
disp('Граничные условия')
fprintf('Слева: %s\nСправа: %s\n',Gr{LR(1)},Gr{LR(2)})
fprintf('Жёсткость упругого основания c=%d Н/м^3\n',c)
fprintf('Модуль упругости E=%d МПа\n',E)
if ~isempty(M),
    disp('Изгибающие моменты')
    disp(' k      zk      Mk')
    fprintf('%2.0f      %5.2f      %6.2f\n',[1:size(M,1);M'])
end
if ~isempty(F),
    disp('Сосредоточенные силы')
    disp(' k      zk      Fk')
    fprintf('%2.0f      %5.2f      %6.2f\n',[1:size(F,1);F'])
end
if ~isempty(q),
    disp('Распределённая нагрузка')
    disp([' k      zk1      ' ...
          'zk2      qk'])
    fprintf('%2.0f      %5.2f      %5.2f      %6.2f\n',[1:size(q,1);q'])
end
b=50;h=100; %input for beam parameters in cm
Jx=b*h^3/12;
s=1; % max размер по вертикали
if ~isempty(M),
    s=max([s;abs(M(:,2))]);
end
if ~isempty(F),
    s=max([s;abs(F(:,2))]);
end
if ~isempty(q),
    s=max([s;abs(q(:,3))]);
end
s002=0.02*s; % масштаб по вертикали
L002=0.02*L; % масштаб по горизонтали
figure
plot([0,L,L,0],[-s002,-s002,s002,s002],'k') % балочка
xlim([-L002,L+L002]) % пределы по осям
ylim([-1.2*s,1.2*s])
hold on
switch LR(1), % рисуем граничное условие слева
case 1, % жёсткое защемление
    plot([0,0],[-0.1*s,0.1*s],'k') % заделка слева
    for k=0:9,
        plot([0,-L002],[(5-k)*s002,(3-k)*s002],'k');
    end
case 2, % шарнир
    plot(0,-2*s002,'ok') % левая опора

```

```

    plot([0 -L002 +L002 0],...
        [-2*s002 -6*s002 -6*s002 -2*s002], 'k')
    plot([-1.5*L002 1.5*L002],...
        [-6*s002 -6*s002], 'k')
end
switch LR(2), % рисуем граничное условие справа
case 1, % жёсткое защемление
    plot([L,L],[-0.1*s,0.1*s], 'k') % заделка справа
    for k=0:9,
        plot([L,L+L002],[ (5-k)*s002, (3-k)*s002], 'k');
    end
case 2, % шарнир
    plot(L,-2*s002, 'ok') % правая опора
    plot([L L-L002 L+L002 L],...
        [-2*s002 -6*s002 -6*s002 -2*s002], 'k')
    plot([L-1.5*L002 L+1.5*L002],...
        [-6*s002 -6*s002], 'k')
end
for k=1:size(M,1), % изгибающие моменты
    x=M(k,1);
    v=M(k,2);
    plot([x-0.1*L,x,x+0.1*L],[-v,-v,v,v], 'k');
    arrow([x-0.05*L,-v],[x-0.1*L,-v]);
    arrow([x+0.05*L,v],[x+0.1*L,v]);
    text(x-L002,abs(v)+0.1*s,['\itM_{'} num2str(k) ''])
end
for k=1:size(F,1), % поперечные силы
    x=F(k,1);
    v=F(k,2);
    plot([x,x],[-v,0], 'k');
    arrow([x,-0.05*s*sign(v)],[x,0]);
    text(x-L002,-(abs(v)+0.1*s)*sign(v),['\itF_{'} num2str(k) ''])
end
for k=1:size(q,1), % распределённые нагрузки
    x1=q(k,1);
    x2=q(k,2);
    v=q(k,3);
    plot([x1,x1,x2,x2],[0,-v,-v,0], 'k');
    arrow([x1,-0.05*s*sign(v)],[x1,0]);
    arrow([x2,-0.05*s*sign(v)],[x2,0]);
    text((x1+x2)/2,-(abs(v)+0.1*s)*sign(v),['\itq_{'} num2str(k) ''])
end
text(0.38*L,-s,['\itc\rm=' num2str(c) ' \itH\rm/\itm^3'])
hold off
set(get(gcf,'CurrentAxes'),...
    'FontName','Times New Roman Cyr','FontSize',12)
title('\bfЭскиз балочки')
xlabel('\itz\rm, м')
ylabel('\itM\rm, кНм, \itF\rm, кН, \itq\rm, кН/м')

alpha=c*b*1e-3; % коэффициент жёсткости, Н/м^2
Leq=(4*E*Jx*1e-2/alpha)^0.25; % приведённая длина, м
fprintf('Коэффициент жёсткости alpha=c*b=%d Н/м^2\n',alpha)
fprintf(['Приведённая длина Leq>='...
    '(4*EJx*alpha)^(1/4)=%12.5f м\n'],Leq)
fprintf('Отношение L/Leq=%12.8f\n',L/Leq)
zk=unique([0;L]); % точки переключения
if ~isempty(M)
    zk=unique([zk;M(:,1)]);
end
if ~isempty(F)
    zk=unique([zk;F(:,1)]);
end
if ~isempty(q)

```

```

zk=unique([zk;q(:,1);q(:,2)]);
end
nk=length(zk)-1; % число интервалов
fprintf('Число интервалов переключения nk=%d\n',nk)
disp('Точки переключения, m')
fprintf('%d ',zk)
fprintf('\n')

nu=4;
fprintf('Общее число неизвестных n=%d\n',nu)
seq1=cell(nu,1); % уравнения
seq2=seq1; % развёрнутые уравнения
A=zeros(nu); % матрица системы уравнений
b=zeros(nu,1); % вектор правых частей
switch LR(1), % граничное условие слева
case 1, % жёсткое защемление
seq1{1}='EJx*w0';
seq1{2}='EJx*theta0';
A(1,1)=1;
A(2,2)=1;
case 2, % шарнир
seq1{1}='EJx*w0';
seq1{2}='M0';
A(1,1)=1;
A(2,3)=1;
otherwise % свободный край
seq1{1}='M0';
seq1{2}='Q0';
A(1,3)=1;
A(2,4)=1;
end
seq2{1}=seq1{1};
seq2{2}=seq1{2};
switch LR(2), % граничное условие справа
case 1, % жёсткое защемление
seq1{3}=['EJx*w(' num2str(L) ')'];
seq1{4}=['EJx*theta(' num2str(L) ')'];
seq2{3}=['EJx*w0*Y1(L/Leq)+EJx*theta0*Leq*Y2(L/Leq)+'...
'M0*Leq^2*Y3(L/Leq)+Q0*Leq^3*Y4(L/Leq)'];
seq2{4}=['-4*EJx*w0/Leq*Y4(L/Leq)+EJx*theta0*Y1(L/Leq)+'...
'M0*Leq*Y2(L/Leq)+Q0*Leq^2*Y3(L/Leq)'];
Y1=cosh(L/Leq)*cos(L/Leq); % Y1(L/Leq)
Y2=0.5*(cosh(L/Leq)*sin(L/Leq)+...
sinh(L/Leq)*cos(L/Leq)); % Y2(L/Leq)
Y3=0.5*sinh(L/Leq)*sin(L/Leq); % Y3end
Y4=0.25*(cosh(L/Leq)*sin(L/Leq)-...
sinh(L/Leq)*cos(L/Leq)); % Y4(L/Leq)
A(3,1)=Y1;
A(3,2)=Leq*Y2;
A(3,3)=Leq^2*Y3;
A(3,4)=Leq^3*Y4;
A(4,1)=-4/Leq*Y4;
A(4,2)=Y1;
A(4,3)=Leq*Y2;
A(4,4)=Leq^2*Y3;
if ~isempty(M),
for k1=1:size(M,1),
seq2{3}=[seq2{3} '+Leq^2*M'...
num2str(k1) '*Y3((L-'...
num2str(M(k1,1)) ')/Leq)'];
seq2{4}=[seq2{4} '+Leq*M' num2str(k1) ...
'*Y2((L-' num2str(M(k1,1)) ')/Leq)'];
end
Y2=0.5*(cosh((L-M(:,1))/Leq).*sin((L-M(:,1))/Leq)+...

```

```

    sinh((L-M(:,1))/Leq).*cos((L-M(:,1))/Leq));
Y3=0.5*sinh((L-M(:,1))/Leq).*sin((L-M(:,1))/Leq);
b(3)=b(3)+Leq^2*sum(M(:,2).*Y3);
b(4)=b(4)+Leq*sum(M(:,2).*Y2);
end
if ~isempty(F),
    for k1=1:size(F,1),
        seq2{3}=[seq2{3} '+Leq^3*F' num2str(k1) ...
            '*Y4((L-' num2str(F(k1,1)) ') /Leq)'];
        seq2{4}=[seq2{4} '+Leq^2*F' num2str(k1) ...
            '*Y3((L-' num2str(F(k1,1)) ') /Leq)'];
    end
    Y3=0.5*sinh((L-F(:,1))/Leq).*sin((L-F(:,1))/Leq);
    Y4=0.25*(cosh((L-F(:,1))/Leq).*sin((L-F(:,1))/Leq)-...
        sinh((L-F(:,1))/Leq).*cos((L-F(:,1))/Leq));
    b(3)=b(3)+Leq^3*sum(F(:,2).*Y4);
    b(4)=b(4)+Leq^2*sum(F(:,2).*Y3);
end
if ~isempty(q),
    for k1=1:size(q,1),
        seq2{3}=[seq2{3} '-Leq^4/4*q' num2str(k1) ...
            '*Y1((L-' num2str(q(k1,1)) ') /Leq)-1)+' ...
            'Leq^4/4*q' num2str(k1) '*Y1((L-' ...
            num2str(q(k1,2)) ') /Leq)-1)'];
        seq2{4}=[seq2{4} '+Leq^3*q' num2str(k1) ...
            '*Y4((L-' num2str(q(k1,1)) ') /Leq)-Leq^3*q' ...
            num2str(k1) '*Y4((L-' num2str(q(k1,2)) ') /Leq)'];
    end
    Y1=cosh((L-q(:,1))/Leq).*cos((L-q(:,1))/Leq);
    Y4=0.25*(cosh((L-q(:,1))/Leq).*sin((L-q(:,1))/Leq)-...
        sinh((L-q(:,1))/Leq).*cos((L-q(:,1))/Leq));
    b(3)=b(3)-Leq^4/4*sum(q(:,3).*Y1-1);
    b(4)=b(4)+Leq^3*sum(q(:,3).*Y4);
    Y1=cosh((L-q(:,2))/Leq).*cos((L-q(:,2))/Leq);
    Y4=0.25*(cosh((L-q(:,2))/Leq).*sin((L-q(:,2))/Leq)-...
        sinh((L-q(:,2))/Leq).*cos((L-q(:,2))/Leq));
    b(3)=b(3)+Leq^4/4*sum(q(:,3).*Y1-1);
    b(4)=b(4)-Leq^3*sum(q(:,3).*Y4);
end
case 2, % шарнир
seq1{3}=['EJx*w(' num2str(L) ')'];
seq1{4}=['M(' num2str(L) ')'];
seq2{3}=['EJx*w0*Y1(L/Leq)+EJx*theta0*Leq*Y2(L/Leq)+' ...
    'M0*Leq^2*Y3(L/Leq)+Q0*Leq^3*Y4(L/Leq)'];
seq2{4}=['-4*EJx*w0/Leq^2*Y3(L/Leq)' ...
    '-4*EJx*theta0/Leq*Y4(L/Leq)+' ...
    'M0*Y1(L/Leq)+Q0*Leq*Y2(L/Leq)'];
Y1=cosh(L/Leq)*cos(L/Leq); % Y1(L/Leq)
Y2=0.5*(cosh(L/Leq)*sin(L/Leq)+...
    sinh(L/Leq)*cos(L/Leq)); % Y2(L/Leq)
Y3=0.5*sinh(L/Leq)*sin(L/Leq); % Y3end
Y4=0.25*(cosh(L/Leq)*sin(L/Leq)-...
    sinh(L/Leq)*cos(L/Leq)); % Y4(L/Leq)
A(3,1)=Y1;
A(3,2)=Leq*Y2;
A(3,3)=Leq^2*Y3;
A(3,4)=Leq^3*Y4;
A(4,1)=-4/Leq^2*Y3;
A(4,2)=-4/Leq*Y4;
A(4,3)=Y1;
A(4,4)=Leq*Y2;
if ~isempty(M),
    for k1=1:size(M,1),
        seq2{3}=[seq2{3} '+Leq^2*M' num2str(k1) '*Y3((L-' ...

```

```

        num2str(M(k1,1)) ')/Leq '];
    seq2{4}=[seq2{4} '+M' num2str(k1) '*Y3((L-' ...
        num2str(M(k1,1)) ')/Leq '];
end
Y1=cosh((L-M(:,1))/Leq).*cos((L-M(:,1))/Leq);
Y3=0.5*sinh((L-M(:,1))/Leq).*sin((L-M(:,1))/Leq);
b(3)=b(3)+Leq^2*sum(M(:,2)).*Y3;
b(4)=b(4)+sum(M(:,2)).*Y1;
end
if ~isempty(F),
    for k1=1:size(F,1),
        seq2{3}=[seq2{3} '+Leq^3*F' num2str(k1) '*Y4((L-' ...
            num2str(F(k1,1)) ')/Leq '];
        seq2{4}=[seq2{4} '+Leq*F' num2str(k1) '*Y2((L-' ...
            num2str(F(k1,1)) ')/Leq '];
    end
    Y2=0.5*(cosh((L-F(:,1))/Leq).*sin((L-F(:,1))/Leq)+...
        sinh((L-F(:,1))/Leq).*cos((L-F(:,1))/Leq));
    Y4=0.25*(cosh((L-F(:,1))/Leq).*sin((L-F(:,1))/Leq)-...
        sinh((L-F(:,1))/Leq).*cos((L-F(:,1))/Leq));
    b(3)=b(3)+Leq^3*sum(F(:,2)).*Y4;
    b(4)=b(4)+Leq*sum(F(:,2)).*Y2;
end
if ~isempty(q),
    for k1=1:size(q,1),
        seq2{3}=[seq2{3} '-Leq^4/4*q' num2str(k1) ...
            *(Y1((L-' num2str(q(k1,1)) ')/Leq)-1)+Leq^4/4' ...
            *q' num2str(k1) *(Y1((L-' ...
            num2str(q(k1,2)) ')/Leq)-1)'];
        seq2{4}=[seq2{4} '+Leq^4*q' ...
            num2str(k1) '*Y1((L-' ...
            num2str(q(k1,1)) ')/Leq)-Leq^4*q' ...
            num2str(k1) '*Y3((L-' ...
            num2str(q(k1,2)) ')/Leq)'];
    end
    Y1=cosh((L-q(:,1))/Leq).*cos((L-q(:,1))/Leq);
    Y3=0.5*sinh((L-q(:,1))/Leq).*sin((L-q(:,1))/Leq);
    b(3)=b(3)-Leq^4/4*sum(q(:,3)).*(Y1-1);
    b(4)=b(4)+Leq^2*sum(q(:,3)).*Y3;
    Y1=cosh((L-q(:,2))/Leq).*cos((L-q(:,2))/Leq);
    Y3=0.5*sinh((L-q(:,2))/Leq).*sin((L-q(:,2))/Leq);
    b(3)=b(3)+Leq^4/4*sum(q(:,3)).*(Y1-1);
    b(4)=b(4)-Leq^2*sum(q(:,3)).*Y3;
end
otherwise % свободный край
seq1{3}=['M(' num2str(L) ')'];
seq1{4}=['Q(' num2str(L) ')'];
seq2{3}=['-4*EJx*w0/Leq^2*Y3(L/Leq)-' ...
    '4*EJx*theta0/Leq*Y4(L/Leq)+' ...
    'M0*Y1(L/Leq)+Q0*Leq*Y2(L/Leq)'];
seq2{4}=['-4*EJx*w0/Leq^3*Y2(L/Leq)-' ...
    '4*EJx*theta0/Leq^2*Y3(L/Leq)-' ...
    '4*M0/Leq*Y4(L/Leq)+Q0*Y1(L/Leq)'];
Y1=cosh(L/Leq).*cos(L/Leq); % Y1(L/Leq)
Y2=0.5*(cosh(L/Leq).*sin(L/Leq)+...
    sinh(L/Leq).*cos(L/Leq)); % Y2(L/Leq)
Y3=0.5*sinh(L/Leq).*sin(L/Leq); % Y3end
Y4=0.25*(cosh(L/Leq).*sin(L/Leq)-...
    sinh(L/Leq).*cos(L/Leq)); % Y4(L/Leq)
A(3,1)=-4/Leq^2*Y3;
A(3,2)=-4/Leq*Y4;
A(3,3)=Y1;
A(3,4)=Leq*Y2;
A(4,1)=-4/Leq^3*Y2;

```

```

A(4,2)=-4/Leq^2*Y3;
A(4,3)=-4/Leq*Y4;
A(4,4)=Y1;
if ~isempty(M),
    for k1=1:size(M,1),
        seq2{3}=[seq2{3} '+M' num2str(k1) ...
            '*Y1((L-' num2str(M(k1,1)) ') /Leq)'];
        seq2{4}=[seq2{4} '-4/Leq*M' num2str(k1) ...
            '*Y4((L-' num2str(M(k1,1)) ') /Leq)'];
    end
    Y1=cosh((L-M(:,1))/Leq).*cos((L-M(:,1))/Leq);
    Y4=0.25*(cosh((L-M(:,1))/Leq).*sin((L-M(:,1))/Leq)-...
        sinh((L-M(:,1))/Leq).*cos((L-M(:,1))/Leq));
    b(3)=b(3)+sum(M(:,2).*Y1);
    b(4)=b(4)-4/Leq*sum(M(:,2).*Y4);
end
if ~isempty(F),
    for k1=1:size(F,1),
        seq2{3}=[seq2{3} '+Leq*F' ...
            num2str(k1) '*Y2((L-' num2str(F(k1,1)) ') /Leq)'];
        seq2{4}=[seq2{4} '+F' num2str(k1) ...
            '*Y1((L-' num2str(F(k1,1)) ') /Leq)'];
    end
    Y1=cosh((L-F(:,1))/Leq).*cos((L-F(:,1))/Leq);
    Y2=0.5*(cosh((L-F(:,1))/Leq).*sin((L-F(:,1))/Leq)+...
        sinh((L-F(:,1))/Leq).*cos((L-F(:,1))/Leq));
    b(3)=b(3)+Leq*sum(F(:,2).*Y2);
    b(4)=b(4)+sum(F(:,2).*Y1);
end
if ~isempty(q),
    for k1=1:size(q,1),
        seq2{3}=[seq2{3} '+Leq^2*q' ...
            num2str(k1) '*Y3((L-' num2str(q(k1,1)) ') /Leq)-' ...
            'Leq^2*q' num2str(k1) '*Y3((L-' num2str(q(k1,2)) ') /Leq)'];
        seq2{4}=[seq2{4} '+Leq*q' num2str(k1) ...
            '*Y2((L-' num2str(q(k1,1)) ...
            ') /Leq)-Leq*q' num2str(k1) '*Y2((L-' ...
            num2str(q(k1,2)) ') /Leq)'];
    end
    Y2=0.5*(cosh((L-q(:,1))/Leq).*sin((L-q(:,1))/Leq)+...
        sinh((L-q(:,1))/Leq).*cos((L-q(:,1))/Leq));
    Y3=0.5*sinh((L-q(:,1))/Leq).*sin((L-q(:,1))/Leq);
    b(3)=b(3)+Leq^2*sum(q(:,3).*Y3);
    b(4)=b(4)+Leq*sum(q(:,3).*Y2);
    Y2=0.5*(cosh((L-q(:,2))/Leq).*sin((L-q(:,2))/Leq)+...
        sinh((L-q(:,2))/Leq).*cos((L-q(:,2))/Leq));
    Y3=0.5*sinh((L-q(:,2))/Leq).*sin((L-q(:,2))/Leq);
    b(3)=b(3)-Leq^2*sum(q(:,3).*Y3);
    b(4)=b(4)-Leq*sum(q(:,3).*Y2);
end
end
disp(['Система уравнений для неизвестных начальных параметров:'])
for k=1:nu,
    disp([seq1{k} '=0'])
end
disp('Распишем её:')
for k=1:nu,
    disp([seq2{k} '=0'])
end
disp('Посчитаем коэффициенты:')
for k=1:nu,
    fprintf('%d*EJx*w0%+d*EJx*theta0%+d*M0%+d*Q0',A(k,1:4))
    fprintf('%+d=0\n',b(k))
end

```

```

disp('Решаем систему уравнений')
np=-A\b;
fprintf(['EJx*w0=%9.5f кНм^3\nEJx*theta0=%9.5f кНм^2\n'...
'M0=%9.5f кНм\nQ0=%9.5f кН\n'],np(1:4))

disp('Перемещения на участках:')
z=[];
EJwz=[];
for k=1:nk,
    EJwz1=['EJx*w0*Y1(z/Leq)+EJx*theta0*Leq*Y2(z/Leq)+'...
'M0*Leq^2*Y3(z/Leq)+Q0*Leq^3*Y4(z/Leq)'];
    EJwz2=sprintf(['%d*Y1(z/Leq)+%d*Leq*Y2(z/Leq)+%d*...
'Leq^2*Y3(z/Leq)+%d*Leq^3*Y4(z/Leq)'],np);
    z0=linspace(zk(k),zk(k+1)); % аргументы для графика
    Y1=cosh(z0/Leq).*cos(z0/Leq); % Y1(z/Leq)
    Y2=0.5*(cosh(z0/Leq).*sin(z0/Leq)+...
sinh(z0/Leq).*cos(z0/Leq)); % Y2(z/Leq)
    Y3=0.5*sinh(z0/Leq).*sin(z0/Leq); % Y3(z/Leq)
    Y4=0.25*(cosh(z0/Leq).*sin(z0/Leq)-...
sinh(z0/Leq).*cos(z0/Leq)); % Y4(z/Leq)
    EJwz0=np(1)*Y1+np(2)*Leq*Y2+np(3)*Leq^2*Y3+np(4)*Leq^3*Y4;
    if ~isempty(M)
        nM=find(M(:,1)<z0(k+1)); % изгибающие моменты слева
        for k1=1:length(nM),
            Y3=0.5*sinh((z0-M(nM(k1),1))/Leq).*...
sin((z0-M(nM(k1),1))/Leq);
            EJwz1=[EJwz1 '+Leq^2*M' num2str(nM(k1)) '*Y3((z-'...
num2str(M(nM(k1),1)))/Leq)'];
            EJwz2=[EJwz2 sprintf('%+d',M(nM(k1),2))...
'*Leq^2*Y3((z-' num2str(M(nM(k1),1)))/Leq)'];
            EJwz0=EJwz0+M(nM(k1),2)*Leq^2*Y3;
        end
    end
    if ~isempty(F)
        nF=find(F(:,1)<z0(k+1)); % сосредоточенные силы слева
        for k1=1:length(nF),
            Y4=0.25*(cosh((z0-F(nF(k1),1))/Leq).*sin((z0-F(nF(k1),1))/Leq)-...
sinh((z0-F(nF(k1),1))/Leq).*cos((z0-F(nF(k1),1))/Leq)); % Y4(z/Leq)
            EJwz1=[EJwz1 '+Leq^3*F' num2str(nF(k1)) '*Y4((z-' ...
num2str(F(nF(k1),1)))/Leq)'];
            EJwz2=[EJwz2 sprintf('%+d',F(nF(k1),2))...
'*Leq^3*Y4((z-' num2str(F(nF(k1),1)))/Leq)'];
            EJwz0=EJwz0+F(nF(k1),2)*Leq^3*Y4;
        end
    end
    if ~isempty(q)
        nq=find(q(:,1)<z0(k+1)); % начало q слева
        for k1=1:length(nq),
            Y1=cosh((z0-q(nq(k1),1))/Leq).*cos((z0-q(nq(k1),1))/Leq);
            EJwz1=[EJwz1 '-Leq^4/4*q' num2str(nq(k1)) ...
'*(Y1((z-' num2str(q(nq(k1),1)))/Leq)-1)'];
            EJwz2=[EJwz2 sprintf('%+d',-q(nq(k1),3))...
'*Leq^4/4*(Y1((z-' num2str(q(nq(k1),1)))/Leq)-1)'];
            EJwz0=EJwz0-Leq^4/4*q(nq(k1),3)*(Y1-1);
        end
    end
    nq=find(q(:,2)<z0(k+1)); % конец q слева
    for k1=1:length(nq),
        Y1=cosh((z0-q(nq(k1),2))/Leq).*cos((z0-q(nq(k1),2))/Leq);
        EJwz1=[EJwz1 '+Leq^4/4*q' num2str(nq(k1)) ...
'*(Y1((z-' num2str(q(nq(k1),2)))/Leq)-1)'];
    end
end

```



```

EJwz2=[EJwz2 sprintf('%+d',q(nq(k1),3)) ...
'*Leq^4/4*(Y1((z-' num2str(q(nq(k1),2)) ')/Leq)-1)'];
EJwz0=EJwz0+Leq^4/4*q(nq(k1),3)*(Y1-1);
end
end
fprintf(['x e [%d,%d]: EJx*w%d(z)=%s=%s\n'],...
zk(k),zk(k+1),k,EJwz1,EJwz2);
z=[z z0]; % накапливаем аргументы
EJwz=[EJwz EJwz0]; % накапливаем функцию
end
figure
plot(z,EJwz/(E*Jx)*1e8,'k')
set(get(gcf,'CurrentAxes'),...
'FontName','Times New Roman Cyr','FontSize',12)
title('\bfЭпюра прогибов')
xlabel('\itz\rm, м')
ylabel('\itw\rm(\itz\rm), мм')

disp('Изгибающие моменты на участках:')
Mz=[];
for k=1:nk,
Mz1=['-4/Leq^2*EJx*w0*Y3(z/Leq)-4/Leq*EJx*theta0*Y4(z/Leq)+'...
'M0*Y1(z/Leq)+Q0*Leq*Y2(z/Leq)'];
Mz2=sprintf(['%d*(-4/Leq^2)*Y3(z/Leq) %+d*(-4/Leq)*Y4(z/Leq)'+...
'%d*Y1(z/Leq) %+d*Leq*Y2(z/Leq)'],np);
z0=linspace(zk(k),zk(k+1)); % аргументы для графика
Y1=cosh(z0/Leq).*cos(z0/Leq); % Y1(z/Leq)
Y2=0.5*(cosh(z0/Leq).*sin(z0/Leq)+...
sinh(z0/Leq).*cos(z0/Leq)); % Y2(z/Leq)
Y3=0.5*sinh(z0/Leq).*sin(z0/Leq); % Y3(z/Leq)
Y4=0.25*(cosh(z0/Leq).*sin(z0/Leq)-...
sinh(z0/Leq).*cos(z0/Leq)); % Y4(z/Leq)
Mz0=-4/Leq^2*np(1)*Y3-4/Leq*np(2)*Y4+np(3)*Y1+np(4)*Leq*Y2;
if ~isempty(M)
nM=find(M(:,1)<zk(k+1)); % изгибающие моменты слева
for k1=1:length(nM),
Y1=cosh((z0-M(nM(k1),1))/Leq).*cos((z0-M(nM(k1),1))/Leq);
Mz1=[Mz1 '+M' num2str(nM(k1)) ...
'*Y1((z-' num2str(M(nM(k1),1)) ')/Leq)'];
Mz2=[Mz2 sprintf('%+d',M(nM(k1),2)) ...
'*Y1((z-' num2str(M(nM(k1),1)) ')/Leq)'];
Mz0=Mz0+M(nM(k1),2)*Y1;
end
end
if ~isempty(F)
nF=find(F(:,1)<zk(k+1)); % сосредоточенные силы слева
for k1=1:length(nF),
Y2=0.5*(cosh((z0-F(nF(k1),1))/Leq).*sin((z0-F(nF(k1),1))/Leq)+...
sinh((z0-F(nF(k1),1))/Leq).*cos((z0-F(nF(k1),1))/Leq));
Mz1=[Mz1 '+Leq*F' num2str(nF(k1)) ...
'*Y2((z-' num2str(F(nF(k1),1)) ')/Leq)'];
Mz2=[Mz2 sprintf('%+d',F(nF(k1),2)) '*Leq*Y2((z-' ...
num2str(F(nF(k1),1)) ')/Leq)'];
Mz0=Mz0+F(nF(k1),2)*Leq*Y2;
end
end
if ~isempty(q)
nq=find(q(:,1)<zk(k+1)); % начало q слева
for k1=1:length(nq),
Y3=0.5*sinh((z0-q(nq(k1),1))/Leq).*sin((z0-q(nq(k1),1))/Leq);
Mz1=[Mz1 '+Leq^2*q' ... num2str(nq(k1)) '*Y3((z-' ...
num2str(q(nq(k1),1)) ')/Leq)'];

```

```

Mz2=[Mz2 sprintf('%+d',q(nq(k1),3)) ...
'*Leq^2*Y3((z-' num2str(q(nq(k1),1)) ')/Leq)'];
Mz0=Mz0+Leq^2*q(nq(k1),3)*Y3;
end
nq=find(q(:,2)<z(k+1)); % конец q слева
for k1=1:length(nq),
Y3=0.5*sinh((z0-q(nq(k1),2))/Leq).*sin((z0-q(nq(k1),2))/Leq);
Mz1=[Mz1 '-Leq^2*q' ... num2str(nq(k1)) '*Y3((z-' ...
num2str(q(nq(k1),2)) ')/Leq)'];
Mz2=[Mz2 sprintf('%+d',-q(nq(k1),3)) ...
'*Leq^2*Y3((z-' num2str(q(nq(k1),2)) ')/Leq)'];
Mz0=Mz0-Leq^2*q(nq(k1),3)*Y3;
end
end
fprintf('x e [%d,%d]: M%d(z)=%s=%s\n',zk(k),zk(k+1),k,Mz1,Mz2);
Mz=[Mz Mz0]; % накапливаем функцию
end
figure
plot(z,Mz,'k')
set(get(gcf,'CurrentAxes'),...
'FontName','Times New Roman Cyr','FontSize',12)
title('\bfЭпюра изгибающих моментов')
xlabel('\itz\rm, м')
ylabel('\itM\rm(\itz\rm), кНм')

disp('Перерезывающие силы на участках:')
Qz=[];
for k=1:nk,
Qz1=['-4/Leq^3*EJx*w0*Y2(z/Leq)'...
'-4/Leq^2*EJx*theta0*Y3(z/Leq)-4/Leq'...
'*M0*Y4(z/Leq)+Q0*Y1(z/Leq)'];
Qz2=sprintf(['%d*(-4/Leq^3)*Y2(z/Leq)%d'...
'*(-4/Leq^3)*Y3(z/Leq)%d*(-4/Leq)*Y4(z/Leq)'...
'%d*Y1(z/Leq)'],np);
z0=linspace(zk(k),zk(k+1)); % аргументы для графика
Y1=cosh(z0/Leq).*cos(z0/Leq); % Y1(z/Leq)
Y2=0.5*(cosh(z0/Leq).*sin(z0/Leq)+...
sinh(z0/Leq).*cos(z0/Leq)); % Y2(z/Leq)
Y3=0.5*sinh(z0/Leq).*sin(z0/Leq); % Y3(z/Leq)
Y4=0.25*(cosh(z0/Leq).*sin(z0/Leq)-...
sinh(z0/Leq).*cos(z0/Leq)); % Y4(z/Leq)
Qz0=-4/Leq^3*np(1)*Y3-4/Leq^2*np(2)*Y4-4/Leq*np(3)*Y1+np(4)*Y2;
if ~isempty(M)
nM=find(M(:,1)<z(k+1)); % изгибающие моменты слева
for k1=1:length(nM),
Y4=0.25*(cosh((z0-M(nM(k1),1))/Leq).*sin((z0-M(nM(k1),1))/Leq)-...
sinh((z0-M(nM(k1),1))/Leq).*cos((z0-M(nM(k1),1))/Leq));
Qz1=[Qz1 '-4/Leq*M' num2str(nM(k1)) ...
'*Y4((z-' num2str(M(nM(k1),1)) ')/Leq)'];
Qz2=[Qz2 sprintf('%+d',M(nM(k1),2)) '*(-4/Leq)'...
'*Y4((z-' num2str(M(nM(k1),1)) ')/Leq)'];
Qz0=Qz0-4/Leq*M(nM(k1),2)*Y4;
end
end
if ~isempty(F)
nF=find(F(:,1)<z(k+1)); % сосредоточенные силы слева
for k1=1:length(nF),
Y1=cosh((z0-F(nF(k1),1))/Leq).*cos((z0-F(nF(k1),1))/Leq);
Qz1=[Qz1 '+F' num2str(nF(k1)) ...
'*Y1((z-' num2str(F(nF(k1),1)) ')/Leq)'];
Qz2=[Qz2 sprintf('%+d',F(nF(k1),2)) ...

```

```

        '*Y1((z-' num2str(F(nF(k1),1)) ')/Leq)'];
    Qz0=Qz0+F(nF(k1),2)*Y1;
end
end
if ~isempty(q)
    nq=find(q(:,1)<z(k+1)); % начало q слева
    for k1=1:length(nq),
        Y2=0.5*(cosh((z0-q(nq(k1),1))/Leq).*sin((z0-q(nq(k1),1))/Leq)+...
            sinh((z0-q(nq(k1),1))/Leq).*cos((z0-q(nq(k1),1))/Leq));
        Qz1=[Qz1 '+Leq*q' num2str(nq(k1)) ...
            '*Y2((z-' num2str(q(nq(k1),1)) ')/Leq)'];
        Qz2=[Qz2 sprintf('%+d',q(nq(k1),3)) ...
            '*Leq*Y1((z-' num2str(q(nq(k1),1)) ')/Leq)'];
        Qz0=Qz0+Leq*q(nq(k1),3)*Y2;
    end
    nq=find(q(:,2)<z(k+1)); % конец q слева
    for k1=1:length(nq),
        Y2=0.5*(cosh((z0-q(nq(k1),2))/Leq).*sin((z0-q(nq(k1),2))/Leq)+...
            sinh((z0-q(nq(k1),2))/Leq).*cos((z0-q(nq(k1),2))/Leq));
        Qz1=[Qz1 '-Leq*q' num2str(nq(k1)) ...
            '*Y2((z-' num2str(q(nq(k1),2)) ')/Leq)'];
        Qz2=[Qz2 sprintf('%+d',-q(nq(k1),3)) ...
            '*Y2((z-' num2str(q(nq(k1),2)) ')/Leq)'];
        Qz0=Qz0-Leq*q(nq(k1),3)*Y2;
    end
end
fprintf('x e [%d,%d]: Q%d(z)=%s=%s\n',zk(k),zk(k+1),k,Qz1,Qz2);
Qz=[Qz Qz0]; % накапливаем функцию
end
figure
plot(z,Qz,'k')
set(get(gcf,'CurrentAxes'),...
    'FontName','Times New Roman Cyr','FontSize',12)
title('\bfЭпюра перерезывающих сил')
xlabel('\itz\rm, м')
ylabel('\itQ\rm(\itz\rm), кН')

[Mmax,iMmax]=max(abs(Mz)); % максимальный изгибающий момент
fprintf(['Максимальный по модулю изгибающий момент' ...
    '|Mmax|=%7.2f кНм\n'],abs(Mmax))
fprintf('достигается при z=%6.2f м\n',z(iMmax))
[Qmax,iQmax]=max(abs(Qz)); % максимальная перерезывающая сила
fprintf(['Максимальная по модулю перерезывающая сила' ...
    '|Qmax|=%7.2f кН\n'],abs(Qmax))
fprintf('достигается при z=%6.2f м\n',z(iQmax))
fprintf('при z=%6.2f м: |Q|=%7.2f кН\n',...
    z(iMmax),abs(Qz(iMmax)))

```

ПРИЛОЖЕНИЕ Б

Аналитическое решение на Wolfram Mathematica, балка на упругом основании

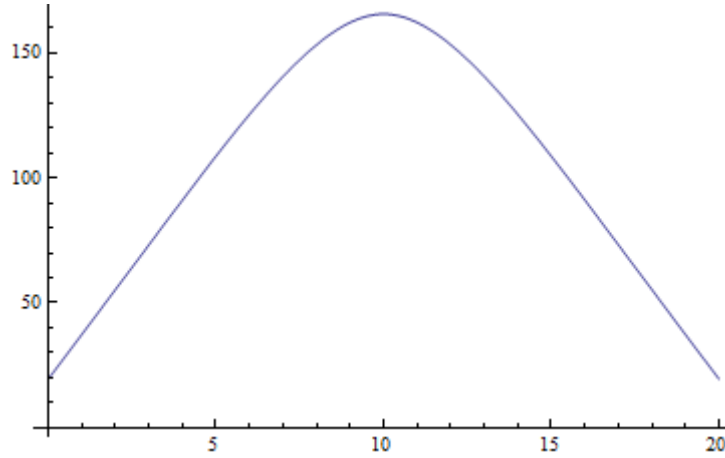
```

sol = DSolve[{1125000 * w''''[x] - 6516.22 * w'''[x] + 2401.57 * w''[x] == 500 * DiracDelta[x - 10],
-1125000 * w''[0] == 0, -1125000 * w'''[0] + 6516.22 * w'[0] == 0,
-1125000 * w''[20] == 0, -1125000 * w'''[20] + 6516.22 * w'[20] == 0}, w, x];
m = sol[[1]];

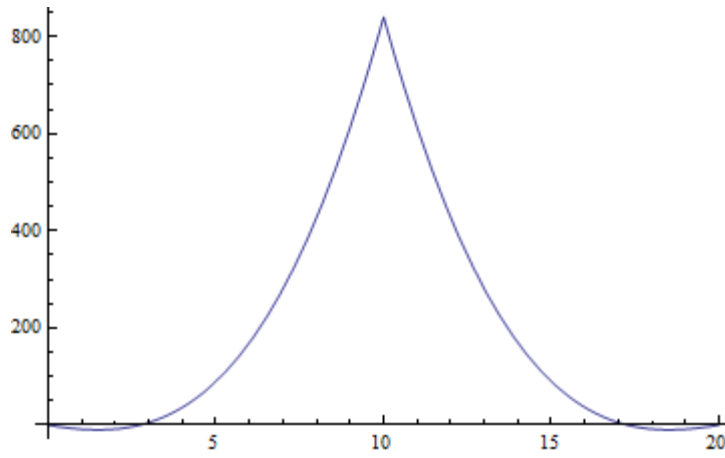
```

Графики

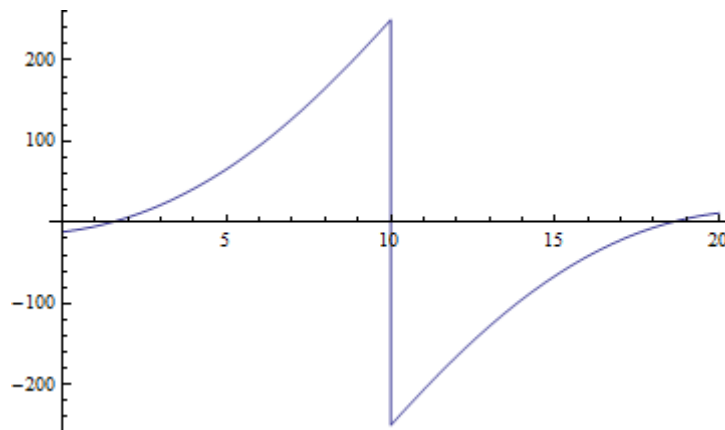
```
Plot[10000 * w[x] /. m, {x, 0, 20}]
```



```
Plot[-1125000 w''[x] /. m, {x, 0, 20}]
```



```
Plot[-1125000 w'''[x] /. m, {x, 0, 20}]
```



ПРИЛОЖЕНИЕ В

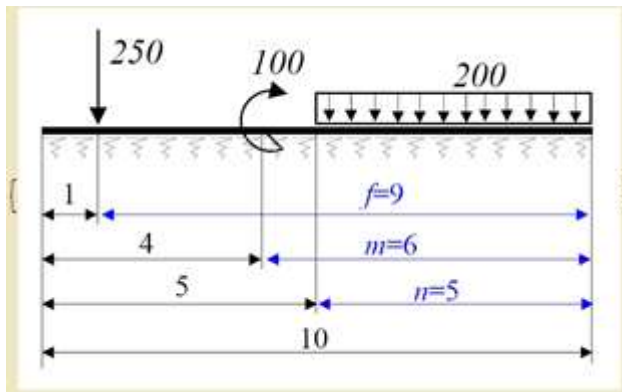
Балка на винклеровском упругом основании

Моделирование основания и балки

Телегенов К.А.

ЕНУ, ММФ, Механика

Пример взят с Structural Analysis II, Lecture notes, S. Parvanova, 2011



Очистить все переменные и определения

```
ClearAll["Global`*"]
```

Геометрия, нагрузка и жесткость

```
EI = 343 750; (* ЖЕСТКОСТЬ БАЛКИ*)
L = 10; (*ДЛИНА БАЛКИ*)
k = 55 000; (* ЖЕСТКОСТЬ ГРУНТА*)
q = 200; (*РАЗСПРЕДЕЛЕННАЯ НАГРУЗКА*)
a = 5; (*Расстояние от левого края до равномерной нагрузки*)
F = 250; (* Нагрузка силы*)
b = 1; (*Расстояние от левого конца к силе F*)
M = -100; (*Интенсивность момента*)
c = 4; (*Расстояние от левого конца до момента M*)
```

Определить непрерывную функцию нагрузки

```
Load[x_] = q * (HeavisideTheta[x - a]) + F * DiracDelta[x - b] + M * DiracDelta'[x - c]; (*функция нагрузки*)
```

Определить дифференциальное уравнение

```
ODE = EI * w''''[x] + k * w[x] == Load[x]; (*дифференциальное уравнение покоя*)
```

Определить граничные условия

```
BC1 = -EI * w''[0] == 0; (*натуральный (сила) граничное условие при x=0*)
BC2 = -EI * w''[0] == 0; (*натуральный (сила) граничное условие при x=0*)
BC3 = -EI * w''[L] == 0; (*натуральный (сила) граничное условие при x=L*)
BC4 = -EI * w''[L] == 0;
(*натуральный (сила) граничное условие при x=L*)
```

Решение дифференциального уравнения + граничными условиями

```
Solution = DSolve[{ODE, BC1, BC2, BC3, BC4}, w[x], x]; (*решение дифф. уравнения с граничными условиями*)
```

Функция перемещения

```
w[x_] = w[x] /. N[Solution][[1]]; (*определение функции поперечного смещения*)
```

Внутренние силы и реакции почвы

```

BM[x_] = -EI * w'''[x] ;
(*изгибающий момент*)
V[x_] = -EI * w''''[x] ;
(*поперечные силы*)
R[x_] = k * w[x] ; (*реакция грунта*)

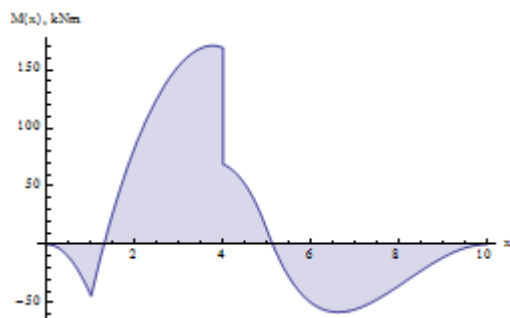
```

Эпюры

```

Plot[-BM[x], {x, 0, L}, AxesLabel -> {"x", "M(x), kNm"}, Filling -> Axis, PlotRange -> All] (*Эпюра изгибающих моментов*)

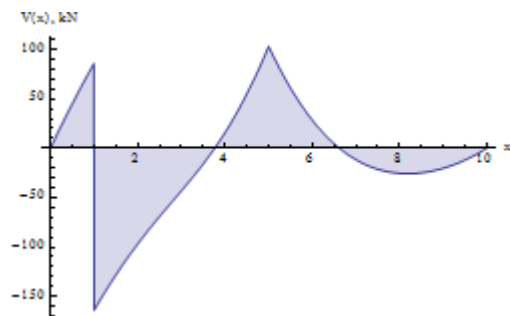
```



```

Plot[V[x], {x, 0, L}, AxesLabel -> {"x", "V(x), kN"}, Filling -> Axis, PlotRange -> All] (*Эпюра поперечных сил*)

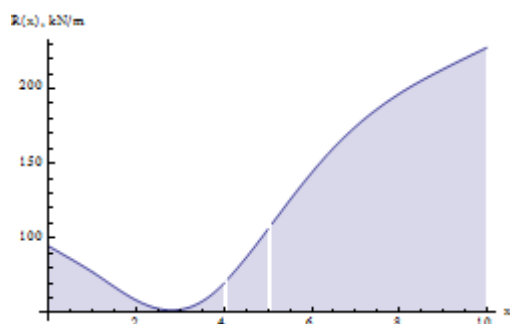
```



```

Plot[R[x], {x, 0, L}, AxesLabel -> {"x", "R(x), kN/m"}, Filling -> Axis, PlotRange -> All] (*Реакция грунта*)

```



Проверка

```

Chop[Integrate[R[x], dx] == Chop[Integrate[Load[x], dx]

```

```

True

```