

Билет 8.

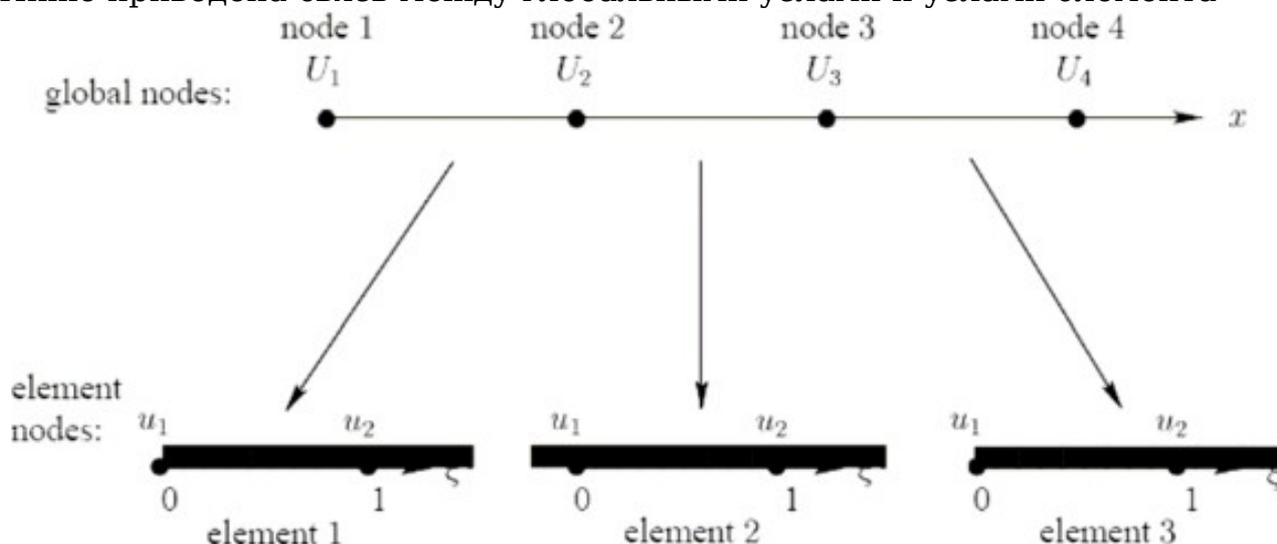
1. Базисные и весовые функции метода конечных элементов для решения уравнений в частных производных (одномерный случай).

Пусть имеется полученное экспериментально одномерное распределение температуры вдоль балки (x – координата точки, в которой измерена температура) и мы хотим найти математическую функцию, описывающую это распределение. Можно попытаться описать эту зависимость при помощи полинома и найти коэффициенты полинома при помощи метода наименьших квадратов. Полином высокой степени достаточно точно описывает набор экспериментальных данных, но при этом возникают нежелательные осцилляции, полином же низкой степени, плохо описывает кривую. Попробуем решить задачу, разбив балку на области (элементы) и попробуем описать каждую область при помощи полинома невысокой степени. Введем параметр s , который характеризует расстояние от точки до края балки, и при помощи метода наименьших квадратов найдем для каждого элемента (области) линейный полином, описывающий каждую область наилучшим образом.

Линейные полиномы хорошо описывают температурное поле в каждой области, но получившаяся аппроксимация не является непрерывной, на границах областей решения надо «сшивать». Это можно сделать следующим образом. Пусть u_1 и u_2 – значения температуры u в узлах, тогда линейную функцию между этими двумя узлами можно записать в виде: $u(\xi) = (1-\xi)u_1 + \xi u_2$, где $\xi (0 \leq \xi \leq 1)$ – нормированное расстояние вдоль кривой. Введём обозначения: $\phi_1(\xi) = 1-\xi$, $\phi_2(\xi) = \xi$ тогда

$u(\xi) = \phi_1(\xi)u_1 + \phi_2(\xi)u_2$. Удобно всегда связывать значение температуры в узле u_n с узлом элемента n и связывать значение температуры U_Δ в глобальном узле Δ с локальным узлом n элемента e при помощи матрицы связи $\Delta(n,e)$, т.е. $u_n = U_{\Delta(n,e)}$.

Ниже приведена связь между глобальными узлами и узлами элемента



Первый элемент интерполируется выражением:

$$u(\xi) = \phi_1(\xi)u_1 + \phi_2(\xi)u_2, u_1 = U_1, u_2 = U_2$$

Второй – выражением: $u(\xi) = \phi_1(\xi)u_1 + \phi_2(\xi)u_2, u_1 = U_2, u_2 = U_3$

Третий – выражением: $u(\xi) = \phi_1(\xi)u_1 + \phi_2(\xi)u_2, u_1 = U_3, u_2 = U_4$

Мы получили непрерывное температурное поле, описанное значениями температуры в узлах и линейными базисными функциями. Базисные функции можно рассматривать как весовые функции для значений в узлах. Эти весовые функции можно рассматривать как глобальные и весовая функция w_n , связанная с глобальным узлом n строится из базисных функций элементов соседних узлов.

Таким образом, мы получили непрерывную параметрическую зависимость температуры u от ξ , но чтобы получить $u(x)$, мы должны связать x и ξ . Можно написать интерполяционную формулу для определения x по значениям в узлах, так, например, для элемента 1 получим: $x(\xi) = \phi_1(\xi)x_1 + \phi_2(\xi)x_2$ и аналогично для двух других элементов. Зависимость температуры от x можно определить при помощи двух соотношений: $u(\xi) = \sum_n \phi_n(\xi)u_n$, $x(\xi) = \sum_n \phi_n(\xi)x_n$ Суммирование проводится по всем узлам элемента.

2. Поведение системы: диаграммы последовательностей.

На диаграмме последовательностей иллюстрируются события, инициированные в системе исполнителями. Создание диаграмм последовательностей является существенной частью исследования возможных способов построения системы. Поэтому данный процесс выполняется в рамках создания модели анализа. Для иллюстрации инициируемых в системе событий можно создать диаграмму последовательностей, воспользовавшись при этом входящей в состав языка UML системой обозначений.

Виды деятельности и зависимости

Диаграммы последовательностей выполняются на стадии анализа цикла разработки. Этот процесс зависит от предварительного формирования прецедентов.

Поведение системы.

Прежде чем приступить к разработке логики работы программного приложения, необходимо исследовать и определить ее поведение как "черного ящика" Поведение системы (system behavior) представляет собой описание того, какие действия выполняет система, без определенного механизма их реализации. Одной из составляющих такого описания является диаграмма последовательностей.

Прецеденты определяют, как исполнители взаимодействуют с программной системой. В процессе этого взаимодействия исполнителем генерируются события, передаваемые системе, которые представляют собой запросы на выполнение некоторой операции. Например, кассир, введя универсальный код товара, тем самым предписывает, чтобы система POST записала данные о приобретении товара. Это событие инициирует в системе выполнение некоторой операции.

Было бы неплохо отделить и проиллюстрировать операции систем, выполнение которых запрашивает исполнитель, поскольку они важны для понимания поведения системы. В качестве системы обозначений в состав языка UML входят диаграммы последовательностей (sequence diagram), С их помощью можно проиллюстрировать взаимодействие исполнителя i ; системой и операции, выполнение которых при этом инициируется.

Диаграмма последовательностей системы (system sequence diagram) является схемой, которая для определенного сценария прецедента показывает, генерируемые внешними исполнителями события, их порядок, а также события, генерируемые внутри самой системы. При этом все системы рассматриваются как черный ящик. Назначение данной диаграммы - отображение событий, передаваемых исполнителями системе через ее границы.

Сценарий прецедента — это его частный случай или реальный путь его реализации.

Диаграмма последовательностей должна быть создана для типичного хода событий прецедентов, а при необходимости и для наиболее существенных альтернативных последовательностей.

Системные события и операции.

Системное событие (system event) — это внешнее входное событие, сгенерированное для системы исполнителем. Событие инициирует выполнение определенной операции. Системная операция (system operation) является операцией, которую система выполняет в ответ на сгенерированное событие. Например, если кассир генерирует системное событие enterItem, то это приводит к выполнению системной операции enterItem. Имена, события и операции идентичны. Различие между ними заключается в том, что событие является именованным стимулирующим воздействием, а операция представляет собой реакцию на это воздействие.

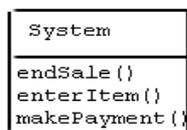
Запись системных операций.

Набор всех требуемых системных операций определяется путем идентификации системных событий. Вместе с параметрами они имеют следующий вид:

- enterItem(UPC, quantity)
- endSale()
- makePayment(amount)

Где же эти операции должны быть записаны? В языке UML операции записываются как тип.

В соответствии с приведенной системой обозначений операции могут быть сгруппированы как операции типа с именем System. При этом параметры можно не указывать.



Такую схему можно без проблем использовать для записи системных операций нескольких систем или процессов распределенного приложения. В этом случае каждой системе ставится в соответствие уникальное имя (System1, System2, ...) и назначаются собственные системные операции.

Представление типа System существенно отличается от объектов, помещаемых в концептуальную модель. Ее элементы иллюстрируют понятия реального мира, а тип System представляет собой искусственно созданную конструкцию.