

Особенности решения задачи динамики пластинки с использованием языка программирования Python

авторы: д.т.н., проф., Мондрус В.Л., к.т.н. Сизов Д.К.

Одним из современных языков программирования общего назначения является язык Python. Особенностью написания программ на данном языке является предельная ясность программного кода, максимальная ориентация на повышение производительности труда разработчика [1],[3]. Язык является свободно распространяемым и имеет реализации интерпретатора практически для всех современных компьютерных платформ.

Учитывая тот факт, что скорость выполнения программ интерпретатором является заведомо значительно более низкой, нежели исполняемых файлов программ, написанных на компилируемых языках программирования, таких как Fortran, C, C++, необходимо при реализации вычислительных процедур основное внимание уделять возможно большему использованию численных библиотек, которые могут быть беспрепятственно использованы в языке Python. В частности, свободнораспространяемая среда программирования Anaconda в стандартном дистрибутиве уже содержит множество численных библиотек, позволяющих обладать удобством манипуляции с массивами и массивоподобными объектами на уровне языка Matlab, либо Fortran, при этом скорость вычислений в значительной степени определяется алгоритмами численных библиотек. Используя язык Python в качестве соединяющего языка (языка-клея) при обращении к библиотекам, написанным на различных компилируемых языках, возможно получить программу, выполняющуюся так же быстро, как и с применением компилируемых языков программирования. Как известно, сейчас сообществом разработчиков языка Python одновременно поддерживаются две ветви развития языка: Python2x и Python3x, в качестве эталонной реализации чаще всего рассматривается версия интерпретатора CPython (www.python.org/dev/implementations/), хотя существует и ряд других интерпретаторов [1],[3], в частности, Jython, IPython, основанные либо на Java-классах, либо ориентированные для работы в Microsoft.NET Framework. Особенного внимания заслуживает экспериментальная версия интерпретатора PyPy, реализованная на языке Python. В научных вычислениях наиболее часто применяется версии Python2x, что объясняется, прежде всего несколько более простой объектной моделью языка и более высокой скоростью работы Python2x по сравнению с Python3x.

Одними из наиболее часто используемых библиотек, используемых в Python являются NumPy и SciPy [1], которые позволяют расширить стандартные объекты Python массивами, функциями для работы с массивами, матричными объектами, функциями работы с полиномами, функции линейной алгебры и др. Использование этих библиотек позволяет создавать быстрые численные программы пока большинство операций производится не над скалярами, а над массивами и матрицами--объектами данных библиотек. Для отображения научной графики--матриц, срезов массивов часто применяется пакет Matplotlib, включающий в себя основные инструменты отображения графиков функций одной и многих переменных. Возможным является использование и такое средство визуализации результатов вычислений как Gnuplot, обладающее встроенным собственным языком и своей системой команд.

В статье приводится решение задачи динамического расчета пластинки, отдельные элементы которой имеют заданное внешнее смещение, источники вибрационного воздействия находятся на контуре пластинки и в пяти внутренних областях внутри нее. Изменение координат точек,

расположенных на контуре пластинки и внутри нее происходит по гармоническому закону с различными частотами. На Рис. 1 показана схема пластинки с учетом приложенных смещений.

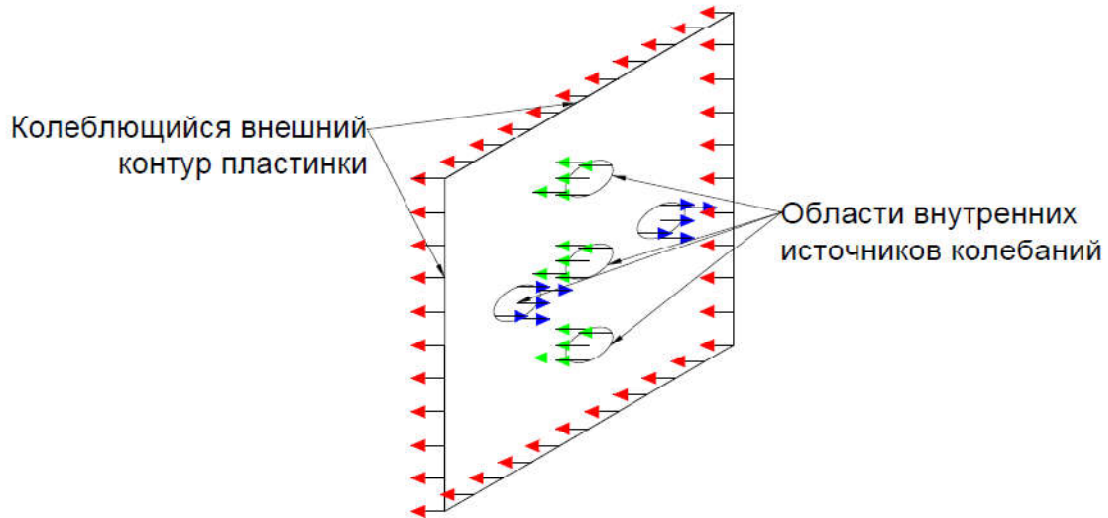


Рис. 1 Схема расположения источников колебаний на квадратной пластинке

Как следует из [2] при постоянной изгибной жесткости пластины уравнение колебаний имеет вид:

$$D\Delta\Delta w + \rho h \frac{\partial^2 w}{\partial t^2} = q(x, t), \text{ где:}$$

$$D = \frac{Eh^3}{12 \cdot (1 - \nu^2)} \text{ - изгибная жесткость пластины}$$

E, ρ, ν, h, w — соответственно модуль упругости материала пластинки, плотность, коэффициент Пуассона, толщина пластинки и прогиб срединной поверхности.

$q(x, t)$ — внешняя нагрузка на пластинку, зависит от координат точки приложения и времени t .

Приведенное в статье уравнение колебаний опирается на классическую теорию Кирхгофа-Лява. Кратко напомним её основные гипотезы:

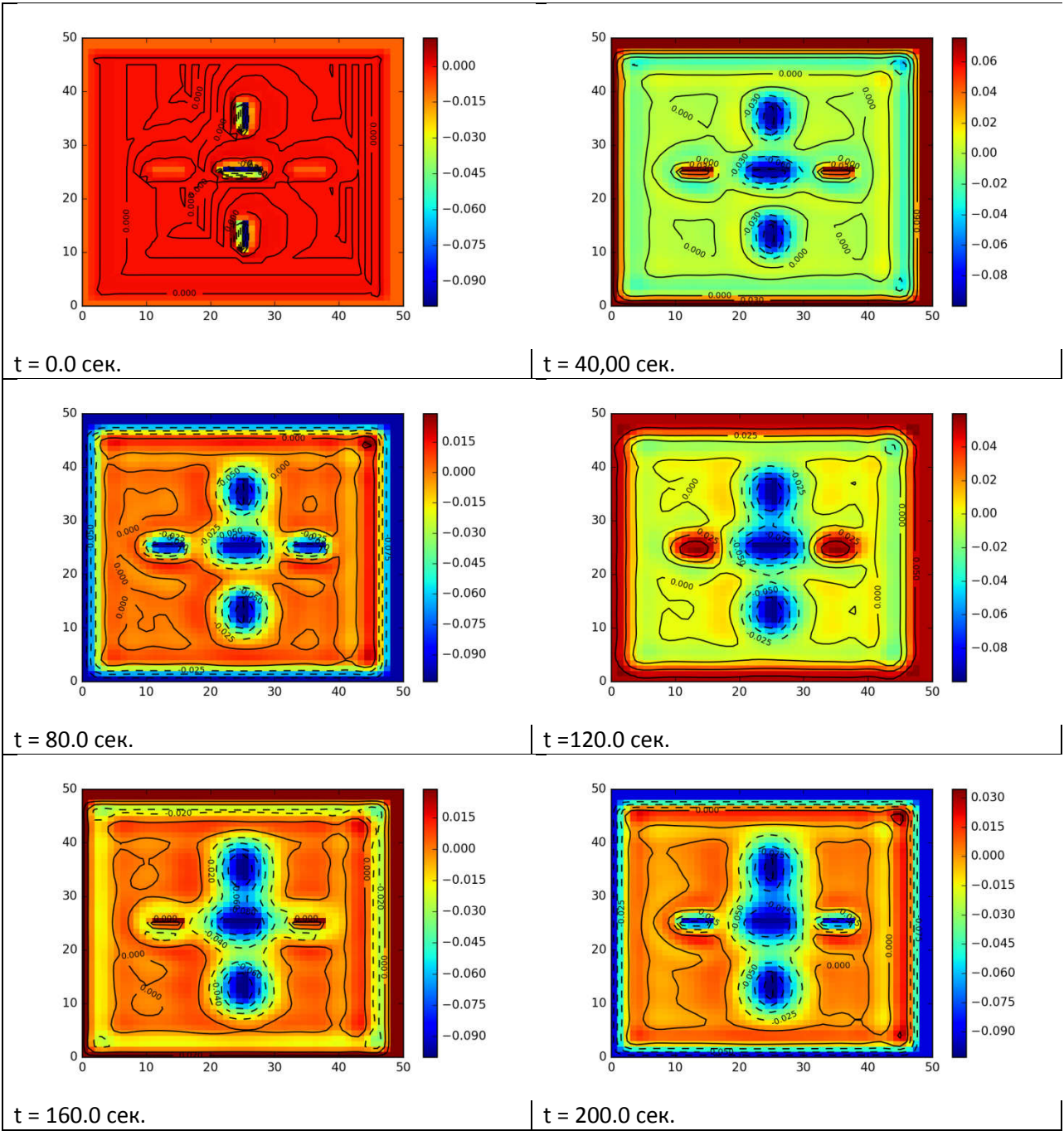
- 1) прямолинейный элемент, нормальный к срединной поверхности до деформации остается прямым и нормальным к срединной поверхности, не меняя своей длины;
- 2) нормальные напряжения между отдельными слоями пластины $\sigma_z = 0$

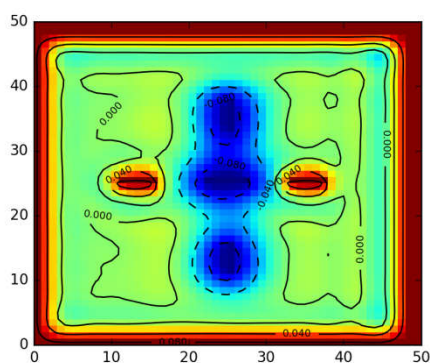
Для решения уравнения динамики удобно применять конечно-разностный метод, основанный на введении разностной сетки, дискретизации области пластины на отдельные участки и дальнейшее решение получившейся алгебраической системы уравнений.

Как известно [4],[5] решение начально-краевых задач для волнового уравнения возможно с использованием явной, явно-неявной и неявных схем. Использование явной схемы позволяет, опираясь на известное решение задачи на предыдущих временных слоях сразу определять

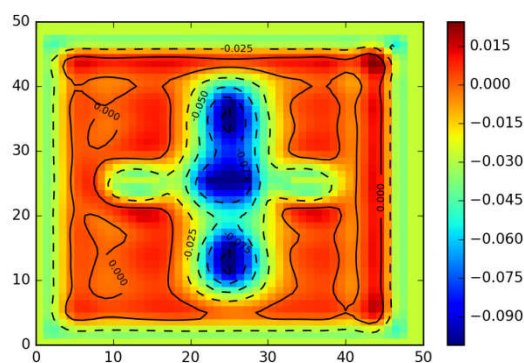
решение на текущем шаге по времени. Данный подход не требует какого-либо решения систем уравнений,но налагает жесткое ограничение на величину шага по времени, пр нарушении данного условия наблюдается постепенное уменьшение достоверности получаемого решения (условие Куранта). Использование неявной схемы требует на каждом из шагов решения получающейся системы уравнений, так как в дискретизированное уравнение и неизвестные, и искомые величины на текущем временном шаге. Явно-неявные схемы, считают в себе преимущества явного метода и гарантированную устойчивость неявной схемы. В статье для решения задачи используется неявная схема, требующая на каждом шаге решения систем уравнений, однако обладающая доказанной гарантированной устойчивостью счета. На рисунках в Таблице 1 приводятся иллюстрации, характеризующие изменение прггиба пластинки в различные моменты времени.

Таблица 1 Изополя прогибов пластинки в различные моменты времени

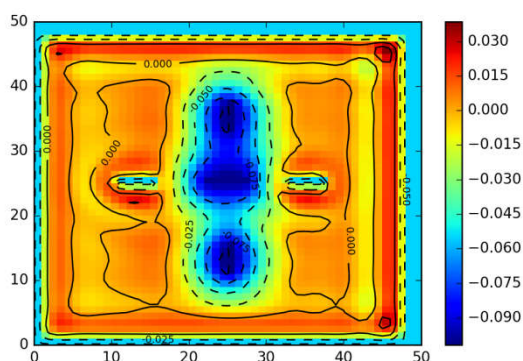




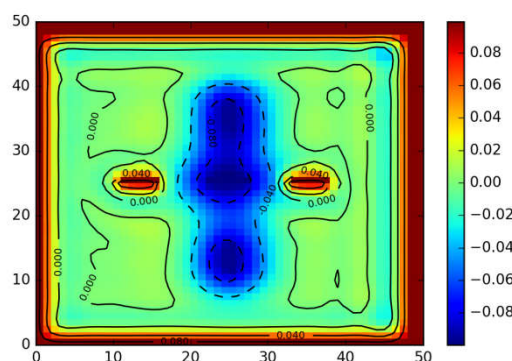
t = 240.0 сек.



t = 280.0 сек.



t = 320.0 сек.



t = 360.0 сек.

В качестве заключения хотелось бы отметить, что разработанный Python-скрипт позволяет не только за приемлемое время получить само численное решение задачи, но и в динамическом режиме наблюдать за изменением численного решения на каждом шаге расчета, что достигается путем использования подключаемой библиотеки научной графики Matplotlib.

Список литературы

1. Лучано Рамальо Python К вершинам мастерства - М: Издательство ДМК, 2016, 767 с. ISBN 978-5-97060-384-0
2. К. Васидзу Вариационные методы в теории упругости и пластичности -М: Мир, 1987, 542 с.
3. Уэс Маккинни Python и анализ данных - М: Издательство ДМК, 2015, 767 с. . ISBN 978-5-97060-315-4
4. Е.Е. Тыртышников Методы численного анализа: учебное пособие для студ. вузов --М: Издательский центр "Академия", 2007.-320 с (Университетский учебник. Сер. Прикладная математика и информатика) ISBN 978-5-7695-3925-1
5. Н.П.Савенкова, О.Г.Проворова, А.Ю.Мокин Численные методы в математическом моделировании-М: АРГАМАК-МЕДИА ИНФА-М, 2014, 176 с, ISBN 978-5-00024-019-9 (АРГАМАК-МЕДИА), ISBN 978-5-16-009705-3 (ИНФА-М, print)