

ПРИМЕНЕНИЕ ГЕТЕРОГЕННЫХ ВЫЧИСЛИТЕЛЬНЫХ СИСТЕМ ДЛЯ ЭМПИРИЧЕСКОЙ МОДОВОЙ ДЕКОМПОЗИЦИИ СИГНАЛА

Мелешеня Д. В., Бранцевич П. Ю.

Кафедра программного обеспечения информационных технологий, Белорусский государственный университет информатики и радиоэлектроники

Минск, Республика Беларусь

E-mail: meleshchenya@gmail.com, branc@bsuir.edu.by

Преобразование Гильберта-Хуанга используется для исследования нестационарных и нелинейных систем. Ключевой частью этого метода анализа данных является декомпозиция исходного сигнала на внутренние эмпирические моды. В классическом методе, предложенном Хуангом для построения мод, используется интерполяция кубическими сплайнами кривых, полученных выделением максимумов и минимумов исходного сигнала. Данная процедура является достаточно трудоемкой, поэтому в данной работе предлагается использовать неоднородную вычислительную систему для сокращения времени вычислений.

ВВЕДЕНИЕ

Анализ данных нашел свое применение как в теоретических исследованиях, так и на практике, помогая выявить ключевые характеристики сигналов, полученных в результате измерения физических величин или численного моделирования. Проблема заключается в том, что зачастую эти данные нестационарные или отражают нелинейные процессы. Для исследования сигналов такого рода отлично подходит преобразование Гильберта-Хуанга. Это возможно благодаря ключевой особенности данного метода анализа – отсутствие предварительно заданного базиса разложения. Вид базисных функций в данном случае определяется непосредственно характеристиками исходного сигнала.

1. ЭМПИРИЧЕСКАЯ МОДОВАЯ ДЕКОМПОЗИЦИЯ

Преобразование Гильберта-Хуанга производится в два основных этапа. На первом этапе производится эмпирическая модовая декомпозиция (empirical mode decomposition, EMD) исходного сигнала. Второй этап – преобразование Гильберта, выполняемое над результатами декомпозиции для вычисления мгновенных фазы и частоты (Hilbert spectral analysis, HSA).

Декомпозиция на модовые функции – итеративная процедура, ставящая в соответствие исходному сигналу набор эмпирических мод (intrinsic mode functions, IMF). В классическом определении данным Хуангом IMF представляет функцию, удовлетворяющую следующим двум условиям: на всем наборе данных количество пересечений оси абсцисс и экстремумов должно быть равно или отличаться на единицу; в любой точке среднее значение огибающих определяемых максимумами и минимумами должно быть равным нулю. Этим обеспечивается корректность, с физической точки зрения, результатов преобразования Гильберта.

Фактически алгоритм модовой декомпозиции, или просеивания, сводится к следующему.

Строятся верхняя и нижняя огибающие, для нахождения которых выделяются локальные максимумы (для верхней) и минимумы (для нижней) на которых строятся интерполяционные сплайны, представляющие собой искомую функцию. Следующим этапом является нахождение разности между средним значением огибающих и исходным сигналом. Далее, если остаток удовлетворяет критерию остановки, он считается очередной модовой функцией, в противном случае разность принимают за исходный сигнал и алгоритм повторяется. В простейшем случае в качестве критерия остановки может выступать выполнение фиксированного количества итераций. После нахождения IMF, ее вычитают из исходного сигнала и, если разность не является монотонной функцией, то алгоритм просеивания повторяется [3]. На рисунке 1 изображен исходный сигнал (Source) и результат его разложение на модовые функции (IMF1-3).

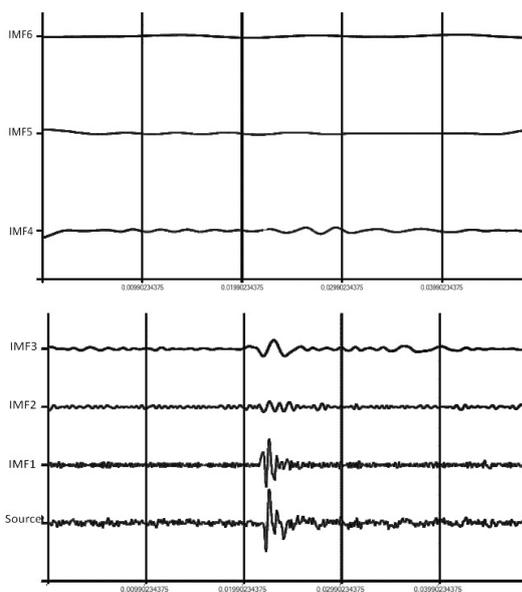


Рис. 1 – Пример разложения исходного сигнала на эмпирические модовые функции (IMF)

II. ДЕКОМПОЗИЦИЯ СИГНАЛА С ИСПОЛЬЗОВАНИЕМ ГРАФИЧЕСКОГО ПРОЦЕССОРА

Ощутимую часть времени в приведенном алгоритме занимает построение огибающих. Поэтому ускорение этого этапа обеспечивает сокращение времени выполнения модовой декомпозиции в целом. Следует отметить, что сигнал в общем случае представляет собой достаточно длинную последовательность точек, так что устранение зависимости между ними в процессе расчета и их параллельная обработка позволяет добиться прироста производительности. Для достижения высокой степени параллелизма предлагается использовать гетерогенную вычислительную систему, состоящую из процессора общего назначения (central processing unit, CPU) и графического процессора (graphics processing unit, GPU). Такой выбор обусловлен тем, что в силу своих архитектурных особенностей, GPU обеспечивает одновременный запуск большого числа потоков, выполняющих вычисления. CPU в данной связке служит для координации работы системы в целом. В качестве платформы реализации был выбран фреймворк OpenCL, предназначенный для написания и запуска вычислений на различных аппаратных платформах [1].

Для построения огибающей используется сплайн Катмула-Рома [2]. Суть метода заключается в том, что для каждого интервала исходной функции находится свой полином, описывающий кривую на данном участке. При этом значение в некоторой точке произвольного интервала определяется по формуле 1. В этой формуле

$$t = (x - x_k) / (x_{k+1} - x_k),$$

$$m_n = (p_{(n+1)} - p_{(n-1)}) / (t_{(n+1)} - t_{(n-1)}),$$

$h_{00}, h_{01}, h_{10}, h_{11}$ – базисные функции Эрмита.

Поскольку вычисление значений на каждом интервале не зависит от значений на других интервалах, расчет для каждого отрезка выполняется в отдельном потоке на отдельном вычислительном ядре GPU. Таким образом, с учетом программной модели OpenCL, итерация алгоритма модовой декомпозиции состоит из следующих этапов:

1. определение максимумов и минимумов исходного сигнала;
2. копирование исходного сигнала и массивов экстремумов в память устройства (GPU);

3. построение сплайнов огибающих;
4. вычисление среднего и разности между средним и исходным сигналом;
5. вычисление критерия остановки (в простейшем случае увеличение счетчика на единицу);
6. если критерий достигнут – эмпирическая мода найдена – скопировать моду в память хоста;
7. если критерий не достигнут – скопировать разность в участок памяти с исходным сигналом и повторить этапы 3-5.

ЗАКЛЮЧЕНИЕ

В результате разработано программное средство, позволявшее производить декомпозицию сигнала на внутренние модовые функции с использованием графического процессора. Для построения огибающих использованы сплайны Катмула-Рома. На рисунке 2 отображены результаты экспериментов на графическом адаптере AMD Radeon HD 6470M и Intel Core i7-4790. Из представленных графиков видно, что данная реализация эффективна для анализа последовательностей размер которых превышает $8 \cdot 10^4$ точек. В противном случае затраты на копирование данных из памяти доступной центральному процессору в память вычислительного устройства составляют существенную часть времени, что делает неэффективным применение GPU.

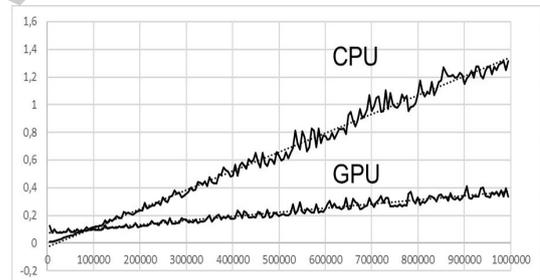


Рис. 2 – Сравнение зависимости времени декомпозиции сигнала от его длины при запуске на графическом и центральном процессорах

1. Munshi, A. The OpenCL Specification / A. Munshi // Khronos OpenCL Working Group. – March 2014.
2. Роджерс, Д. Математические основы машинной графики / Д. Роджерс, Дж. Адамс. – М.: Мир, 2001. – 604 с.
3. Huang, N. E. The empirical mode decomposition and the Hilbert spectrum for nonlinear and non-stationary time series analysis / N. E. Huang, Z. Shen [and other]. – Proc. R. Soc. Lond. A., 1998. – Т.454. – P. 903-995.

$$p(x) = h_{00}(t)p_k + h_{01}(t)p(x_{k+1} - x_k)m_k + h_{01}(t)p_{k+1} + h_{11}(t)(x_{k+1} - x_k)m_{k+1}, \quad (1)$$