

Реализация вычислительно-эффективного алгоритма оценки взаимной временной задержки сигналов

© 2012 г. Р.А. Еришов, А.А. Логинов

Нижегородский государственный университет им. Н.И. Лобачевского
romanershov1@rambler.ru

В статье предлагается алгоритм оценки взаимной временной задержки сигналов на основе модификации метода построения взаимной функции неопределённости. Модификация позволяет распараллелить алгоритм и реализовать его с использованием графического процессора общего назначения (GPU), что в сотни раз сокращает время вычислений по сравнению с процессорами традиционной архитектуры.

Ключевые слова: взаимная временная задержка, функция неопределённости.

Задача определения взаимной временной задержки сигналов возникает при решении ряда практических вопросов в различных областях прикладной физики: радиосвязь, радиолокация, гидролокация, сейсморазведка, дефектоскопия и т.д. Наиболее общим алгоритмом решения данной задачи при наличии шума и смещении спектров сигналов вследствие эффекта Доплера является алгоритм построения и анализа функции неопределённости опорного и исследуемого сигналов. Однако вследствие перебора по всем возможным значениям временной задержки и доплеровского сдвига и использования преобразования Фурье от массивов больших выборок данный метод является вычислительно неэффективным и не позволяет оценивать взаимную временную задержку между каналами в реальном масштабе времени, что наиболее важно при решении прикладных задач. Цель работы состоит в разработке алгоритма, позволяющего повысить вычислительную эффективность расчёта функции неопределённости.

В процессе разработки вычислительно-эффективных методов проведено моделирование сигналов $s_1[n]$, $s_2[n]$ и методов расчета взаимной функции неопределённости (ФН):

$$\psi(\tau, \Delta F) = \left| \sum_{n=0}^{N-1} s_1[n] s_2^*[n + \tau] \exp(-j2\pi\Delta F n T) \right|.$$

Алгоритм оценки взаимной временной задержки сигналов на основе модификации метода построения функции неопределённости состоит в следующем[1]. На вход поступают сигналы: опорный и исследуемый, организуется цикл перебора по всем возможным временным задержкам, на каждом шаге которого производится поэлементное перемножение сигналов с учётом текущего временного сдвига. Затем производится прореживание результата перемножения, выбирается шаг прореживания K , производится суммирование K отсчётов произведений сигналов, результат заносится в один элемент массива Фурье-преобразования, следующие K отсчётов суммируются, и результат заносится в один элемент

Фурье-преобразования и т.д. пока не будет заполнен весь массив. Далее выполняется быстрое преобразование Фурье. Вычисляются значения модулей преобразования, из этих значений выбирается максимальное и соответствующий ему индекс (соответствующий частоте). Максимальное значение на каждом шаге вычисления задержки заносится в соответствующий массив. В массиве определяется глобальный максимум, положение которого и определяет взаимную временную задержку сигналов.

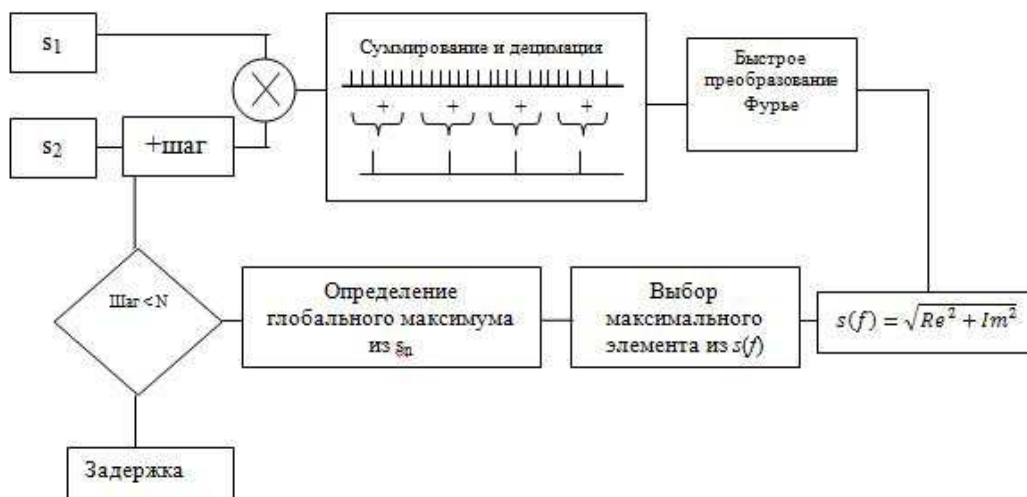


Рис.1. Алгоритм оценки взаимной временной задержки сигналов на основе модификации метода построения функции неопределённости.

В процессе реализации были смоделированы ФМ-2 сигналы, длина опорного сигнала 16384 отсчёта, исследуемого – 32768 отсчётов. Как показывает реализация данного алгоритма, модификация метода построения функции неопределённости позволяет сократить время вычислений до 4 раз по сравнению с вычислением полной функции неопределённости (без прореживания). Для большего ускорения при реализации алгоритма использовались многопоточные функции библиотек Intel Performance Libraries, которые оптимизированы под многоядерные процессоры [2], за счёт чего получено ускорение в 10.4 раза по сравнению с алгоритмом на основе полного вычисления функции неопределённости.

Однако даже такой результат не позволяет оценивать взаимную временную задержку в реальном масштабе времени. Поэтому алгоритм предполагалось реализовать с использованием графического процессора общего назначения, архитектура которого позволяет эффективно выполнять параллельные вычисления. Реализация алгоритма определения взаимной задержки сигналов на основе расчёта функции неопределённости для работы на GPU требует значительных изменений в логике его работы с целью оптимального соответствия аппаратно-программной архитектуре графического процессора. Для реализации алгоритма использовалась технология nVidia CUDA, которая строится на

концепции, что GPU выступает в роли массивно-параллельного сопроцессора к центральному процессору (CPU) [3].

Алгоритм был адаптирован с целью оптимального соответствия программно-аппаратной модели CUDA. Он состоит из четырех действий, которые последовательно выполняются на графическом процессоре (рис. 2).

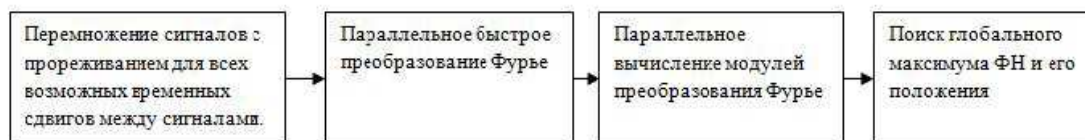


Рис. 2. Алгоритм определения взаимной временной задержки сигналов на GPU.

Сначала производится перемножение сигналов с прореживанием параллельно для всех возможных временных сдвигов. Каждый поток выполняет операцию чтения соответствующих отсчётов сигналов и их умножение, а затем проводится суммирование K элементов произведений при помощи алгоритма параллельной редукции [3]. В результате перемножения с прореживанием получается матрица произведений опорного и исследуемого сигналов, размерность которой по горизонтали равна N_1/K (N_1 – длина опорного сигнала), а по вертикали N_2-N_1 (N_2 – длина исследуемого сигнала).

Следующим этапом алгоритма является выполнение преобразования Фурье над каждой строкой получившейся матрицы при помощи функций специальной библиотеки *cufft*, входящей в состав CUDA SDK.

Далее вычисляются модули комплексных отсчётов функции неопределённости. Так как эти операции независимы друг от друга, их также можно выполнять параллельно. После вычисления модулей производится поиск глобального максимума в матрице, номер строки положения которого соответствует временной задержке между сигналами, а номер столбца – доплеровскому сдвигу. Поиск глобального максимума можно реализовать на основе алгоритма параллельной редукции массива [3].

Результатом реализации данного алгоритма стало ускорение вычислений в более чем 100 раз по сравнению алгоритмом, приведённым на рис. 1, при описанных выше параметрах моделирования. Время вычислений составляет десятки миллисекунд. Таким образом, предложенный алгоритм может быть использован при решении прикладных задач.

Список литературы

1. Algorithms for ambiguity function processing / S. Stein. IEEE Trans. Acoust., Speech, Signal Processing, vol. 29, June 1981, pp. 582-587.
2. Intel® Integrated Performance Primitives 6.0, <http://www.intel.com/cd/software/products/emea/rus/358826.htm>.
3. Боресков А.В., Харламов А.А. Основы работы с технологией CUDA. М.: ДМК Пресс, 2010, 232с.