

Министерство образования Республики Беларусь
Учреждение образования
Белорусский государственный университет
информатики и радиоэлектроники

УДК 004.67

Бессмертный
Николай Анатольевич

Методы оптимизации алгоритмов для обработки звуковых сигналов на
мобильных вычислительных платформах

Автореферат

на соискание степени магистра технических наук
по специальности 1-40 80 01 «Элементы и устройства вычислительной
техники и систем управления»

Научный руководитель
Лихачев Д.С.
канд. тех. наук, доцент

Минск 2020

КРАТКОЕ ВВЕДЕНИЕ

Дискретное Преобразование Фурье (ДПФ) является одной из наиболее важных и широко используемых вычислительных задач. Его приложения широки и включают обработку сигналов, связь и сжатие аудио / изображений / видео. Поэтому быстрые алгоритмы для ДПФ очень ценны.

В настоящее время самым быстрым таким алгоритмом является Быстрое Преобразование Фурье (БПФ). Обычно алгоритм БПФ называется алгоритмом Кули-Тьюки, так как они первыми программно осуществили алгоритм БПФ в начале 60-х годов. А в 1965 году они же опубликовали статью об алгоритме быстрого преобразования Фурье (Cooley-TukeyFFT). БПФ вычисляет преобразование Фурье сигнала размера n за $O(n \log n)$ времени.

Существование алгоритмов ДПФ, более быстрых, чем БПФ, является одним из центральных вопросов в теории алгоритмов. Целью данной работы будет представить алгоритм вычисляющий приближенное преобразование Фурье за время быстрее чем БПФ.

ОБЩАЯ ХАРАКТЕРИСТИКА РАБОТЫ

Целью диссертации является оптимизация вычисления Дискретного Преобразования Фурье в алгоритме оценщика тона IRAPT. Указанная цель определяет следующие задачи:

1. Предложить и обосновать алгоритм оптимизации;
2. Описать этот алгоритм;
3. Получить экспериментальные данные, используя алгоритм оптимизации и сравнить их с данными в алгоритме оценщика тона IRAPT.

Диссертационная работа состоит из введения, трех глав, заключения и двух приложений. В первой главе освещены описание алгоритмов ЦОС и постановка задачи. Во второй главе описывается алгоритм оптимизации - алгоритм Разреженного Преобразования Фурье. Третья глава описывает результаты проведенных экспериментов.

КРАТКОЕ СОДЕРЖАНИЕ РАБОТЫ

Разреженное преобразование Фурье начинается с хэширования частотных коэффициентов \hat{x} в сегменты, так что значение ячейки является суммой значений частотных коэффициентов, которые хэшируются в интервал. Так как \hat{x} редкий, многие ячейки будут пустыми и их можно просто выбросить. Алгоритм затем сосредотачивается на непустых сегментах и вычисляет положения и значения коэффициентов большой частоты в этих сегментах в том, что мы называем шагом оценки частоты. Сглаживание нечастотности процесса достигается с помощью фильтров. Фильтр подавляет и обнуляет частотные коэффициенты, которые хэшируются за пределы сегмента, проходя через частотные коэффициенты, которые хэшируются в блок. Простейшим примером этого является фильтр сглаживания. Напомним следующее базовое свойство преобразования Фурье: субсэмплирование во временной области вызывает сглаживание в частотной области. Формально, пусть b будет версией субсэмплирования x , то есть $b(i) = x(i \cdot p)$, где p - коэффициент субсэмплинга, который делит n . Тогда, \hat{b} , преобразование Фурье от b :

$$\hat{b}(i) = \sum_{m=0}^{p-1} \hat{x}(i + m(n/p)). \quad (1)$$

Таким образом, форма группирования, в которой частоты, равные интервалу $B = n/p$, распределены по одному и тому же сегменту, и есть такие сегменты, как показано на рисунке 2.13. Функция хэширования, полученная в результате этого группирования, может быть записана как: $h(f) = f \bmod n/p$. Кроме того, значение в каждом сегменте является суммой значений только частотных коэффициентов, которые хэшируют в сегмент, как можно видеть из уравнения 2.6. Для вышеупомянутого фильтра наложения на сегменты могут быть эффективно вычислены с использованием БПФ с точкой B , за время $O(B \log B)$. Мы устанавливаем $B = O(k)$ и, следовательно, для группирования требуется только $O(k \log k)$ времени и используется только $O(B) = O(k)$ из входных выборок x .

На этом шаге разреженное преобразование Фурье оценивает положения и значения ненулевых частотных коэффициентов, которые создали энергию в каждом из не пустых сегментов. В редких случаях многие не пустые сегменты, скорее всего, будут иметь хэшинг с одним ненулевым частотным коэффициентом, и только небольшое число будет иметь столкновение из нескольких ненулевых коэффициентов. Прежде всего сосредоточимся на сегментах с единичными ненулевыми частотными коэффициентами и оценим значение и положение этой ненулевой частоты, т.е. $\hat{x}(f)$ и соответствующей f . При от-

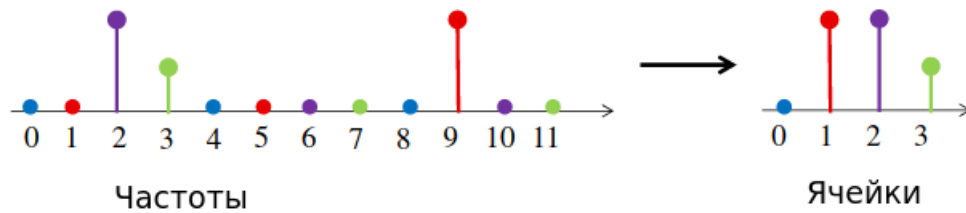


Рисунок 1. – Субсэмплирование сигнала

существовании коллизии значение ненулевой частоты коэффициент - это значение хэша, который он хэширует, поскольку все другие частоты, которые хэшируют в хэш, имеют нулевые значения. Следовательно, мы можем легко найти значение ненулевого частотного коэффициента в сегменте. Тем не менее, мы до сих пор не знаем его частотное положение f , поскольку сглаживание частоты отображает несколько частот в одну и ту же область. Простейшим способом вычисления является использование свойства фазового вращения преобразования Фурье, в котором говорится, что сдвиг во временной области приводит к повороту фазы в частотной области. В частности, мы снова выполняем процесс сглаживания после кругового сдвига x на t образцов. Поскольку сдвиг во времени приводит к повороту фазы в частотной области, значение сегмента изменяется от $\hat{b}(i) = \hat{x}(f)$ до $\hat{b}^{(t)}(i) = \hat{x}(f) \cdot e^{j\Delta\phi}$, где фазовое вращение:

$$\Delta\phi = 2\pi ft/n \quad (2)$$

Следовательно, используя изменение фазы сегмента, мы можем оценить положение коэффициента ненулевой частоты в сегменте. Обратите внимание, что фаза оборачивается вокруг каждых 2π , и поэтому сдвиг t должен быть равен 1, чтобы избежать фазового переноса при больших значениях исключения f . Поскольку существуют коэффициенты с нулевой частотой, эту оценку частоты можно сделать эффективно, используя не более $O(k)$ вычислений.

Ненулевые частотные коэффициенты, которые изолированы в их собственном сегменте, могут быть правильно оценены, как описано выше. Однако, когда ненулевые частоты сталкиваются в одном и том же сегменте, мы не можем оценивать их правильно. Следовательно, чтобы восстановить полный частотный спектр, нам нужно разрешить конфликты. Чтобы разрешить конфликт, нам нужно повторить преобразование частоты таким образом, чтобы гарантировать, что одни и те же ненулевые частоты не сталкиваются друг с другом каждый раз. То, как мы этого добиваемся, зависит от типа фильтра, используемого для наложения. Например, с помощью фильтров наложения спектров, описанных выше, мы можем многократно объединить спектр в

группы с использованием фильтров наложения спектров с одинаковой частотой дискретизации. Это меняет хеширующую функцию с $h(f) = f \bmod n/p$ на $h'(f) = f \bmod n/p'$, где p и p' взаимно просты. Фильтры гарантируют, что любые две частоты, которые сталкиваются в одной группировке, не будут сталкиваться в другой группировке. Чтобы лучше понять этот момент, рассмотрим пример на рисунке 2.14.

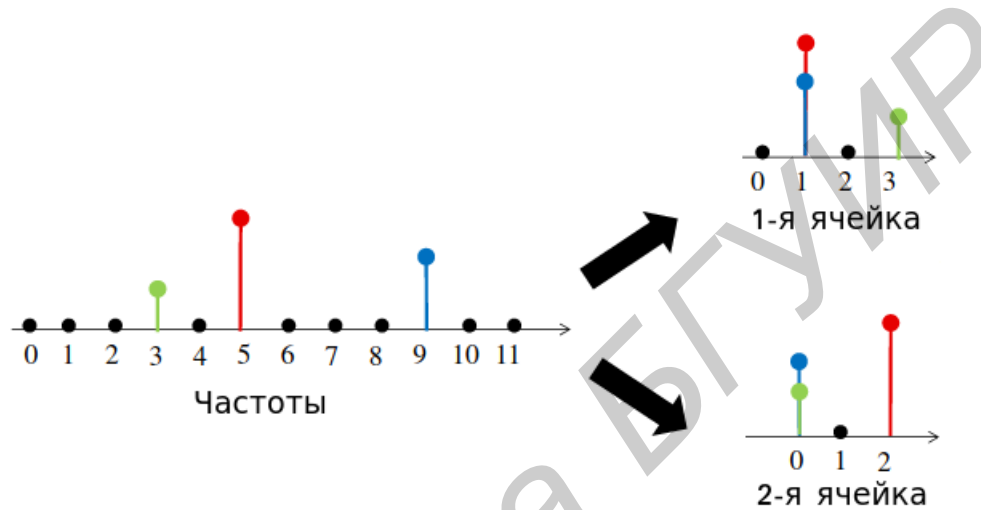


Рисунок 2. – Субсэмплирование сигнала

При первом группировании мы используем фильтр наложения имен, который субсэмплирует временной сигнал с помощью коэффициента 3. В этом случае две частоты, помеченные красным и синим, сталкиваются в сегменте, тогда как частота, помеченная зеленым, не сталкивается, как показано на фигура Во второй раз, когда мы упаковываем данные в группы, мы используем фильтр наложения имен, который под выбирает на 4. На этот раз синяя и зеленая частоты сталкиваются, тогда как красная частота не накапливается. Теперь мы можем разрешить коллизии, выполнив две итерации. Например, мы можем оценить зеленую частоту по первой группировке, где она не сталкивается. 2 Вычтем зеленую частоту из сталкивающегося ковша при второй группировке, чтобы получить голубую частоту. Затем мы возвращаемся к первому сглаживанию и вычитаем синюю частоту из ячейки, где она сталкивается, чтобы получить красную частоту. Перебираем между различными группировками путем оценки частот из сегментов, где они не сталкиваются и вычитают их из сегментов, где они действительно сталкиваются, гарантируя, что каждая ненулевая частота будет изолирована в своем собственном интервале алгоритма. Это позволяет нам правильно оценить каждую ненулевую частоту. Таким образом, в конце этапа разрешения столкновений мы

восстановили все ненулевые частоты и, следовательно, успешно вычислили преобразование Фурье сигнала.

Библиотека БГУИР

ЗАКЛЮЧЕНИЕ

В диссертационной работе был представлен алгоритм Разреженного Преобразования Фурье, позволяющий получить высокую производительность вычисления Дискретного Преобразования Фурье быстрее чем за $O(n \log n)$ времени.

Была поставлена задача оптимизации вычисления Преобразования Фурье для алгоритма IRAPT.

Алгоритм Разреженного Преобразования Фурье реализован на языке C++ и сравнен с алгоритмом БПФ пакета MATLAB, который используется в алгоритме IRAPT, а также произведено сравнение с библиотекой FFTW.

Разреженное Преобразование Фурье использовалось для оценки тона на различных входных сигналах. Исследованы характеристики и скорость работы, исходя из экспериментальных данных Разреженное Преобразование Фурье на небольших объёмах сигналов уступает Быстрому Преобразованию Фурье, но начинает выигрывать когда сигнал достаточно велик. Процент ошибки алгоритма по Евклидовой норме L1 достаточно мал и при проведении экспериментов, ошибок не было замечено.

Результаты работы свидетельствуют о выполнении цели и всех задач исследования.

Результаты работы докладывались на 56-ой научной конференции аспирантов, магистрантов и студентов БГУИР и опубликованы в сборнике тезисов докладов[1–А].

СПИСОК ОПУБЛИКОВАННЫХ РАБОТ

1–А. Бессмертный, Н. А. Методы оптимизации алгоритмов для обработки звуковых сигналов на мобильных вычислительных платформах / Н. А. Бессмертный // Компьютерные системы и сети: 55-я юбилейная научная конференция аспирантов, магистрантов и студентов, Минск, 22-26 апреля 2019 г. / Белорусский государственный университет информатики и радиоэлектроники. — Минск, — 2019. — Т. 10. — С. 247 – 248.

Библиотека БГУИР