АНАЛИЗ БЫСТРОГО АЛГОРИТМА УМНОЖЕНИЯ МАТРИЦ И ВЕКТОРОВ ДЛЯ БАНКА ЦИФРОВЫХ ФИЛЬТРОВ

DOI: 10.36724/2072-8735-2021-15-1-4-10

Manuscript received 25 October 2020;
Accepted 20 November 2020

Крейнделин Виталий Борисович,
Московский технический университет связи и информатики, Москва, Россия, **vitkrend@gmail.com**

Григорьева Елена Дмитриевна,
Московский технический университет связи и информатики, Москва, Россия, **ed.grigorieva@yandex.ru**

Ключевые слова: *многоскоростная обработка сигналов, банк (набор) цифровых фильтров, быстрый алгоритм, чувствительность к ошибкам округления, метод 3М, метод Винограда*

Представлены алгоритмы реализации векторно-матричного умножения, предназначенные для применения в банках (наборах) цифровых фильтров. Указанные алгоритмы обеспечивают существенную экономию вычислительных затрат по сравнению с традиционными алгоритмами. При этом, сокращение вычислительной сложности алгоритмов достигается без ухудшения качественных характеристик банков (наборов) цифровых фильтров. В качестве основы для построения предложенных в статье алгоритмов использованы известные ранее метод Винограда умножения действительных матриц и векторов и два варианта метода типа 3М для умножения комплексных матриц и векторов. Рассмотрены способы комбинирования этих известных методов умножения матриц и векторов для построения банков (наборов) цифровых фильтров. Проведен анализ вычислительной сложности таких способов, который показал возможность сокращения вычислительной сложности по сравнению с традиционным алгоритмом реализации банка (набора) цифровых фильтров примерно в 2,66 раза – при реализации на процессоре без аппаратного умножителя; и в 1,33 раза – при реализации на процессоре с аппаратным умножителем. Эти показатели заметно превышают показатели известных алгоритмов. Проведен анализ чувствительности предложенных в статье алгоритмов к ошибкам округления, возникающим при цифровой обработке сигналов. На основе этого анализа выбран алгоритм, который имеет вычислительную сложность, меньшую, чем сложность традиционного алгоритма, но при этом его чувствительность к ошибкам округления такая же, как и у традиционного алгоритма. Даны рекомендации по его практическому применению при разработке банка (набора) цифровых фильтров.

Информация об авторах:

Крейнделин Виталий Борисович, *заведующий кафедрой ТЭЦ, профессор, д.т.н, Московский технический университет связи и информатики, Москва, Россия*

Григорьева Елена Дмитриевна, *доцент кафедры ТЭЦ, к.т.н., Московский технический университет связи и информатики, Москва, Россия*

Для цитирования:
Крейнделин В.Б., Григорьева Е.Д.. Анализ быстрого алгоритма умножения матриц и векторов для банка цифровых фильтров // T-Comm: Телекоммуникации и транспорт. 2021. Том 15. №1. С. 4-10.

For citation:
Kreyndelin V.B., Grigorieva E.D. (2021) Analysis of fast algorithm of matrix-vector multiplication for the bank of digital filters. *T-Comm*, vol. 15, no.1, pp. 4-10. (*in Russian*)

Введение и постановка задачи

Многоскоростная адаптивная фильтрация сигналов и синтеза систем обработки сигналов на основе использования банка (набора) фильтров широко применяется в современных системах связи [0].

Поскольку верхняя частота спектра полезного сигнала обычно намного меньше частоты дискретизации, то можно использовать пониженную частоту дискретизации, что существенно снизит вычислительную сложность алгоритма обработки сигнала. В настоящее время требования к скорости передачи информации серьезно ужесточаются, что влечет за собой ужесточение требований к вычислительной сложности реализации банков (наборов) фильтров [0], [0].

Поэтому актуальным является поиск путей дальнейшего снижения вычислительной сложности их практической реализации [0], [0]. В известных работах описаны различные подходы к снижению вычислительной сложности алгоритмов обработки сигналов и банков (наборов) цифровых фильтров, в частности [0], [0], [0]. Однако, проблема снижения вычислительной сложности продолжает оставаться актуальной по мере развития техники связи.

Кроме того, при практической реализации на сигнальных процессорах банков (наборов) цифровых фильтров очень важным является оценка чувствительности используемых алгоритмов фильтрации к ошибкам округления, которые неизбежно возникают при цифровой обработке сигналов с помощью вычислительных средств с конечной разрядной сеткой. Упрощение алгоритмов обработки сигналов приводит к изменению порядка осуществления арифметических операций, что в принципе может привести к ухудшению чувствительности упрощенного алгоритма к ошибкам округления. Поэтому важным является, не только предложить упрощенные алгоритмы реализации банков (наборов) цифровых фильтров, но и оценить их чувствительность к ошибкам округления.

Как известно, реализацию банка (набора) цифровых фильтров можно представить в виде умножения некоторой комплексной матрицы на комплексный вектор входных сигналов [0], [0]. Вид матрицы определяется характеристиками используемых цифровых фильтров.

Рассмотрим сначала задачу вычисления произведения действительной квадратной матрицы $\mathbf{A}$ размерности $N \times N$ на действительный вектор-столбец $\mathbf{x}$ размерности $N \times 1$, которая часто встречается в цифровой обработке сигналов [0], [0]:

$$\mathbf{y} = \mathbf{A} \cdot \mathbf{x}, \tag{1}$$

В результате получаем действительный вектор-столбец $\mathbf{y}$ размерности $N \times 1$. Будем полагать также, что размерность N – чётное число.

Вектор $\mathbf{y}$ состоит из N компонент, которые, согласно известному традиционному методу, вычисляются следующим образом:

$$y_i = a_{i1}x_1 + a_{i2}x_2 + \ldots + a_{ij}x_j + \ldots + a_{iN}x_N, \quad i = 1, 2 \ldots N. \tag{2}$$

В методе вычислений (2) число умножений равно N^2, а число сложений равно $N(N-1)$. При больших размерностях N может оказаться, что непосредственная реализация выражения (2) потребует слишком больших вычислительных затрат. Поэтому для снижения вычислительных затрат применяют методы, называемые быстрыми алгоритмами. Рассмотрим некоторые известные быстрые алгоритмы, позволяющие снизить сложность реализации выражения (2).

1. Метод Винограда для умножения действительных матриц и векторов.

Величины y_i могут быть вычислены по известной формуле Винограда при $i = 1, 2 \ldots N$ [0]:

$$y_i = \sum_{k=1}^{N/2} \left(a_{i,2k-1} + x_{2k}\right)\left(a_{i,2k} + x_{2k-1}\right) - \sum_{k=1}^{N/2} a_{i,2k-1} \cdot a_{i,2k} - \sum_{k=1}^{N/2} x_{2k-1} \cdot x_{2k}. \tag{3}$$

Число умножений в формуле Винограда (3) равно $\frac{3}{2} \cdot N^2$, что в 1,5 раза больше, чем число умножений в традиционной формуле (2).

Заметим, что формула (3) обладает следующими особенностями:

- выражение $\sum_{k=1}^{N/2} a_{i,2k-1} \cdot a_{i,2k}$ может быть вычислено **заранее,** поскольку матрица $\mathbf{A}$ в (1) при построении банка фильтров считается постоянной и заранее известной;
- выражение $\sum_{k=1}^{N/2} x_{2k-1} \cdot x_{2k}$ является **одинаковым** для всех $i = 1, 2 \ldots N$, поэтому оно может быть вычислено только **один раз.** В связи с этим для его вычисления требуется только $\frac{N}{2}$ умножений и $\frac{N}{2} - 1$ сложений.

Таким образом, основные вычислительные затраты, необходимые для вычисления (3), сосредоточены в выражении

$$\sum_{k=1}^{N/2} \left(a_{i,2k-1} + x_{2k}\right)\left(a_{i,2k} + x_{2k-1}\right). \tag{4}$$

С учётом изложенного, число выполняемых в реальном времени умножений в формуле Винограда (3) оказывается равным $\frac{1}{2} \cdot \left(N^2 + N\right)$, что асимптотически в **два раза меньше**, чем число умножений в традиционной формуле (2). Число выполняемых в реальном времени сложений в (3) равно $N(N+2)$, что асимптотически равно числу сложений в (2).

2. Методы типа 3M для умножения матриц комплексных чисел

На практике при построении банков фильтров приходится осуществлять вычисление произведения комплексных матриц и векторов [0], [0]. Рассмотрим задачу вычисления произведения комплексной матрицы на комплексный вектор:

$$\begin{aligned}\mathbf{y}=\mathbf{A}\cdot\mathbf{x}&=\left(\mathbf{A}_C+j\mathbf{A}_S\right)\left(\mathbf{x}_C+j\mathbf{x}_S\right)=\\&=\mathbf{A}_C\mathbf{x}_C-\mathbf{A}_S\mathbf{x}_S+j\left(\mathbf{A}_S\mathbf{x}_C-\mathbf{A}_C\mathbf{x}_S\right),\end{aligned}\tag{5}$$

где $\mathbf{A}$ – комплексная квадратная матрица размерности $N\times N$; $\mathbf{A}_C$, $\mathbf{A}_S$ – действительные квадратные матрицы размерности $N\times N$; $\mathbf{x}_C$ $\mathbf{x}_S$ – д ействительные векторы-столбцы размерности $N\times 1$; $\mathbf{x}$ и $\mathbf{y}$ – комплексные векторы-столбцы размерности $N\times 1$.

Выражение (5) определяет традиционный метод перемножения комплексных матриц и векторов [0]. Этот метод требует выполнения четырех действительных векторно-матричных умножений и двух действительных векторных сложений. Существуют, однако, методы, требующие меньшего числа действительных векторно-матричных умножений.

Известен следующий метод реализации комплексного векторно-матричного умножения [0], [0], [0]:

$$\begin{aligned}\mathbf{y}=\mathbf{A}\cdot\mathbf{x}&=\mathbf{A}_C\mathbf{x}_C-\mathbf{A}_S\mathbf{x}_S+\\&+j\left(\left(\mathbf{A}_C+\mathbf{A}_S\right)\cdot\left(\mathbf{x}_C+\mathbf{x}_S\right)-\mathbf{A}_C\mathbf{x}_C-\mathbf{A}_S\mathbf{x}_S\right).\end{aligned}\tag{6}$$

Реализация метода (6) требует только трех действительных векторно-матричных умножений, но одно действительное матричное сложение и четыре действительных векторных сложения. В литературе он известен как метод 3M (3 multiplications, 3 умножения) [0], [0]. Отметим, что матричное сложение $\mathbf{A}_C+\mathbf{A}_S$ может быть вычислено **заранее**, поскольку матрица $\mathbf{A}$ определяет свойства банка фильтров и считается постоянной и заранее известной.

Векторно-матричные умножения в (6) могут быть реализованы, например, с помощью формулы Винограда (3).

Рассмотрим еще одну модификацию метода 3M [0], [0].

$$\begin{aligned}\mathbf{y}=\mathbf{A}\cdot\mathbf{x}&=\left(\mathbf{x}^T\cdot\mathbf{A}^T\right)^T=\left(\mathbf{x}_C^T-\mathbf{x}_S^T\right)\cdot\mathbf{A}_S^T+\mathbf{x}_C^T\cdot\left(\mathbf{A}_C^T-\mathbf{A}_S^T\right)+\\&+j\left(\left(\mathbf{x}_C^T-\mathbf{x}_S^T\right)\cdot\mathbf{A}_S^T+\mathbf{x}_S^T\cdot\left(\mathbf{A}_C^T+\mathbf{A}_S^T\right)\right).\end{aligned}\tag{7}$$

Выражение (7), как и выражение (6), требует выполнения трех действительных векторно-матричных умножений, двух действительных матричных сложений и трех действительных векторных сложений. Отметим также, что матричные сложения в (7) $\mathbf{A}_C^T-\mathbf{A}_S^T$ и $\mathbf{A}_C^T+\mathbf{A}_S^T$, как и в случае (6), может быть вычислено **заранее**, поскольку матрица $\mathbf{A}$, определяющая свойства банка фильтров, считается постоянной и заранее известной.

3. Анализ вычислительной сложности

Сделаем подсчёт числа арифметических операций, необходимых для реализации комплексного векторно-матричного умножения с одновременным применением формулы Винограда (3) и метода 3M (6). В этом случае реальном времени потребуется выполнить:

- $\frac{3}{2}\cdot\left(N^2+N\right)$ действительных умножений;
- $3N\left(N+2\right)+4N=3N^2+10N$ действительных сложений.

Проведем аналогичным образом подсчет числа арифметических операций, необходимых для реализации комплексного векторно-матричного умножения с одновременным применением формулы Винограда (3) и модификации (7) метода 3M. Учтем при этом, что в реальном времени для реализации (7) требуется не четыре, а только три действительных векторных сложения. В (7) требуется выполнить:

- $\frac{3}{2}\cdot\left(N^2+N\right)$ действительных умножений;
- $3N\left(N+2\right)+3N=3N^2+9N$ действительных сложений.

В таблице 1 приведены результаты сравнительного анализа вычислительной сложности

- традиционного алгоритма реализации комплексного векторно-матричного умножения (2),
- алгоритма (3), (6),
- алгоритма (3), (7).

Таблица 1

	Количество умножений	Количество сложений	Общее количество операций
Традиционный Алгоритм (2)	$4N^2$	$2N\left(N-1\right)$	$6N^2-2N$
Алгоритм (3), (6)	$1{,}5N^2+1{,}5N$	$3N^2+10N$	$4{,}5N^2+11{,}5N$
Алгоритм (3), (7)	$1{,}5N^2+1{,}5N$	$3N^2+9N$	$4{,}5N^2+10{,}5N$

Из таблицы 1 видно, что применение алгоритма (3), (6) или алгоритма (3), (7) позволяет получить существенный асимптотический (при $N\to\infty$) выигрыш в вычислительной сложности по сравнению с традиционным алгоритмом (2). Величина этого выигрыша практически одинакова у алгоритмов (3), (6) и (3), (7) и составляет:

- в 2,66 раза – при реализации на процессоре без аппаратного умножителя;
- в 1,33 раза (или на 25%) – при реализации на процессоре с аппаратным умножителем.

Нужно отметить, что при наличии в сигнальном процессоре аппаратного умножителя время выполнения операции умножения и время выполнения операции сложения совпадают [0], [0]. Если в процессоре отсутствует аппаратный умножитель, то время выполнения операции умножения в несколько десятков раз превышает время выполнения операции сложения.

4. Анализ чувствительности различных методов реализации комплексного векторно-матричного умножения к ошибкам округления

Как известно, вычисления всегда сопровождаются ошибками округления [0]. Поэтому важным является определить степень чувствительности рассмотренного метода (6) к ошибкам округления и сравнение ее с чувствительностью к ошибкам округления традиционного метода (5).

Будем полагать, что

$$\begin{aligned}&\tilde{\mathbf{A}}_C=\mathbf{A}_C+\delta\mathbf{A}_C;\tilde{\mathbf{A}}_S=\mathbf{A}_S+\delta\mathbf{A}_S;\tilde{\mathbf{x}}_C=,\\&=\mathbf{x}_C+\delta\mathbf{x}_C;\tilde{\mathbf{x}}_S=\mathbf{x}_S+\delta\mathbf{x}_S,\end{aligned}\tag{8}$$

где $\tilde{\mathbf{A}}_C,\tilde{\mathbf{A}}_S$ – приближенные значения матриц $\mathbf{A}_C$ и $\mathbf{A}_S$, соответственно; $\delta\mathbf{A}_C,\delta\mathbf{A}_S$ – ошибки округления матриц $\mathbf{A}_C$ и $\mathbf{A}_S$, соответственно; $\tilde{\mathbf{x}}_C,\tilde{\mathbf{x}}_S$ – приближенные значения векторов $\mathbf{x}_C$ и $\mathbf{x}_S$, соответственно; $\delta\mathbf{x}_C,\delta\mathbf{x}_S$ – ошибки округления векторов $\mathbf{x}_C$ и $\mathbf{x}_S$, соответственно.

Также предположим, что

$$\tilde{\mathbf{y}}_C=\mathbf{y}_C+\delta\mathbf{y}_C;\tilde{\mathbf{y}}_S=\mathbf{y}_S+\delta\mathbf{y}_S,\tag{9}$$

где $\tilde{\mathbf{y}}_C,\tilde{\mathbf{y}}_S$ – приближенные значения векторов $\mathbf{y}_C$ и $\mathbf{y}_S$, соответственно; $\delta\mathbf{y}_C,\delta\mathbf{y}_S$ – ошибки округления векторов $\mathbf{y}_C$ и $\mathbf{y}_S$, соответственно.

Задача состоит в том, чтобы найти зависимость ошибок округления $\delta\mathbf{y}_C,\delta\mathbf{y}_S$ результата от ошибок округления $\delta\mathbf{A}_C,\delta\mathbf{A}_S$, $\delta\mathbf{x}_C,\delta\mathbf{x}_S$ сомножителей в (1). Для этого подставим (8) в (5). В результате после несложных алгебраических преобразований имеем:

$$\begin{aligned}&\delta\mathbf{y}_C=\delta\mathbf{A}_C\mathbf{x}_C+\mathbf{A}_C\delta\mathbf{x}_C+\delta\mathbf{A}_C\delta\mathbf{x}_C-\\&-\delta\mathbf{A}_S\mathbf{x}_S-\mathbf{A}_S\delta\mathbf{x}_S-\delta\mathbf{A}_S\delta\mathbf{x}_S\\&\delta\mathbf{y}_S=\delta\mathbf{A}_S\mathbf{x}_C+\mathbf{A}_S\delta\mathbf{x}_C+\mathbf{A}_C\delta\mathbf{x}_S+\\&+\delta\mathbf{A}_C\mathbf{x}_S+\delta\mathbf{A}_S\delta\mathbf{x}_C+\delta\mathbf{A}_C\delta\mathbf{x}_S.\end{aligned}\tag{10}$$

Пренебрегая членами второго порядка малости, из (10) приближенно получим:

$$\begin{aligned}&\delta\mathbf{y}_C\approx\delta\mathbf{A}_C\mathbf{x}_C+\mathbf{A}_C\delta\mathbf{x}_C-\delta\mathbf{A}_S\mathbf{x}_S-\mathbf{A}_S\delta\mathbf{x}_S\\&\delta\mathbf{y}_S\approx\delta\mathbf{A}_S\mathbf{x}_C+\mathbf{A}_S\delta\mathbf{x}_C+\mathbf{A}_C\delta\mathbf{x}_S+\delta\mathbf{A}_C\mathbf{x}_S\end{aligned}.\tag{11}$$

Выражения (11) показывают зависимость ошибок округления результата от ошибок округления исходных данных для традиционного метода (5) реализации комплексного векторно-матричного умножения.

Проведем анализ метода (6) реализации комплексного векторно-матричного умножения. Для этого подставим (8) в (6). В результате после ряда несложных алгебраических преобразований и пренебрегая членами второго порядка малости, приближенно имеем:

$$\begin{aligned}&\delta\mathbf{y}_C\approx\delta\mathbf{A}_C\mathbf{x}_C+\mathbf{A}_C\delta\mathbf{x}_C-\delta\mathbf{A}_S\mathbf{x}_S-\mathbf{A}_S\delta\mathbf{x}_S\\&\delta\mathbf{y}_S\approx\delta\mathbf{A}_S\mathbf{x}_C+\mathbf{A}_S\delta\mathbf{x}_C+\mathbf{A}_C\delta\mathbf{x}_S+\delta\mathbf{A}_C\mathbf{x}_S\end{aligned}.\tag{12}$$

Выражения (12) в точности совпадают с выражениями (11). Это позволяет сделать вывод о том, что чувствительность к ошибкам округления метода (6) и чувствительность к ошибкам округления традиционного метода (5) являются одинаковыми.

Рассмотрим теперь метод (7) реализации комплексного векторно-матричного умножения. Для этого подставим (8) в (7). В результате приближенно имеем:

$$\begin{aligned}&\delta\mathbf{y}_C\approx\delta\mathbf{A}_C\mathbf{x}_C+\mathbf{A}_C\delta\mathbf{x}_C-\delta\mathbf{A}_S\mathbf{x}_S-\mathbf{A}_S\delta\mathbf{x}_S\\&\delta\mathbf{y}_S\approx\left(2\mathbf{A}_C-\mathbf{A}_S\right)\delta\mathbf{x}_S+2\delta\mathbf{A}_C\left(\mathbf{x}_S+\mathbf{x}_C\right)+\\&+\mathbf{A}_C\delta\mathbf{x}_C-\delta\mathbf{A}_S\mathbf{x}_S.\end{aligned}\tag{13}$$

Из сравнения (13) с (12) видно, что действительная часть погрешности $\delta\mathbf{y}_C$ метода (7) совпадает с действительной частью погрешности известного метода (5) и метода (6). Однако мнимая часть погрешности $\delta\mathbf{y}_S$ метода (7) заметно отличается от мнимой части погрешности методов (5) и (6).

Проведем сравнительную оценку величины этих погрешностей. В качестве критерия оценки будем использовать величину дисперсии суммарной погрешности компонент вектора $\mathbf{y}$. Корреляционная матрица $\mathbf{R}_S$ действительного случайного вектора $\delta\mathbf{y}_S$ определяется следующим образом [0]:

$$\mathbf{R}_S=E\left\{\delta\mathbf{y}_S\delta\mathbf{y}_S^T\right\},\tag{14}$$

где $E\{\dots\}$ – оператор математического ожидания; $(\dots)^T$ – операция транспонирования.

Дисперсия σ_s^2 суммарной погрешности компонент вектора $\mathbf{y}$ представляет собой сумму дисперсий компонент этого вектора. Поскольку в корреляционной матрице дисперсии компонент являются элементами ее главной диагонали, то

$$\sigma_s^2=tr\left(\mathbf{R}_S\right)=tr\left(E\left\{\delta\mathbf{y}_S\delta\mathbf{y}_S^T\right\}\right),\tag{15}$$

где $tr(\dots)$ – след матрицы.

Для традиционного метода (5) и метода (6) имеем:

$$\begin{aligned}&\sigma_s^2=tr\left(\mathbf{R}_S\right)=tr\left(E\left\{\delta\mathbf{y}_S\delta\mathbf{y}_S^T\right\}\right)\approx\\&tr\left(E\left\{\delta\mathbf{A}_S\mathbf{x}_C\mathbf{x}_C^T\delta\mathbf{A}_S^T\right\}\right)+tr\left(E\left\{\mathbf{A}_S\delta\mathbf{x}_C\delta\mathbf{x}_C^T\mathbf{A}_S^T\right\}\right)+\\&tr\left(E\left\{\mathbf{A}_C\delta\mathbf{x}_S\delta\mathbf{x}_S^T\mathbf{A}_C^T\right\}\right)+tr\left(E\left\{\delta\mathbf{A}_C\mathbf{x}_S\mathbf{x}_S^T\delta\mathbf{A}_C^T\right\}\right)=\\&tr\left(E\left\{\mathbf{x}_C^T\delta\mathbf{A}_S^T\delta\mathbf{A}_S\mathbf{x}_C\right\}\right)+tr\left(\mathbf{A}_S E\left\{\delta\mathbf{x}_C\delta\mathbf{x}_C^T\right\}\mathbf{A}_S^T\right)+\\&tr\left(\mathbf{A}_C E\left\{\delta\mathbf{x}_S\delta\mathbf{x}_S^T\right\}\mathbf{A}_C^T\right)+tr\left(E\left\{\mathbf{x}_S^T\delta\mathbf{A}_C^T\delta\mathbf{A}_C\mathbf{x}_S\right\}\right)=\\&tr\left(\mathbf{x}_C^T E\left\{\delta\mathbf{A}_S^T\delta\mathbf{A}_S\right\}\mathbf{x}_C\right)+tr\left(\mathbf{A}_S\mathbf{A}_S^T\right)+tr\left(\mathbf{A}_C\mathbf{A}_C^T\right)+\\&+tr\left(\mathbf{x}_S^T E\left\{\delta\mathbf{A}_C^T\delta\mathbf{A}_C\right\}\mathbf{x}_S\right)=\\&tr\left(\mathbf{x}_C^T\mathbf{x}_C\right)+tr\left(\mathbf{A}_S\mathbf{A}_S^T\right)+tr\left(\mathbf{A}_C\mathbf{A}_C^T\right)+tr\left(\mathbf{x}_S^T\mathbf{x}_S\right)\end{aligned}\tag{16}$$

В (16) использовано известное матричное тождество [0], [0], [0]:

$$tr\left(\mathbf{AB}\right)=tr\left(\mathbf{BA}\right) \quad (17)$$

Кроме того, при преобразованиях (16) было принято, что:

$$\begin{aligned}&E\left\{\delta\mathbf{x}_C\delta\mathbf{x}_C^T\right\}=\mathbf{1};\quad E\left\{\delta\mathbf{x}_\mathbf{S}\delta\mathbf{x}_\mathbf{S}^T\right\}=\mathbf{1};\\&E\left\{\delta\mathbf{A}_C^T\delta\mathbf{A}_C\right\}=\mathbf{1};\quad E\left\{\delta\mathbf{A}_\mathbf{S}^T\delta\mathbf{A}_\mathbf{S}\right\}=\mathbf{1}\end{aligned}, \quad (18)$$

где $\mathbf{1}$ – единичная матрица.

В выражениях (18) учтено, что

- ошибки округления различных компонент матриц и векторов некоррелированы;
- дисперсии всех ошибок округления одинаковы и нормированы.

С целью анализа величины ошибок округления для метода (7) обратимся теперь к уравнению (13). После преобразований, аналогичных преобразованиям (16), получим:

$$\begin{aligned}&tr\left(E\left\{\delta\mathbf{y}_\mathbf{S}\delta\mathbf{y}_\mathbf{S}^T\right\}\right)\approx 2\cdot tr\left(\left(\mathbf{x}_\mathbf{S}+\mathbf{x}_C\right)^T\left(\mathbf{x}_\mathbf{S}+\mathbf{x}_C\right)\right)+\\&+tr\left(\left(2\mathbf{A}_C-\mathbf{A}_\mathbf{S}\right)\left(2\mathbf{A}_C-\mathbf{A}_\mathbf{S}\right)^T\right)+\\&tr\left(\mathbf{A}_C\mathbf{A}_C^T\right)+tr\left(\mathbf{x}_\mathbf{S}^T\mathbf{x}_\mathbf{S}\right).\end{aligned} \quad (19)$$

Из сравнения выражений (18) и (19) видно, что при одинаковых величинах ошибок округления входных данных величина погрешности результата σ_s^2 для метода (7) значительно превышает погрешность результата у традиционного метода (5) и метода (6).

5. Анализ чувствительности различных методов реализации умножения действительных матриц и векторов к ошибкам округления

Обратимся снова к уравнению (2), описывающему традиционный метод вычисления произведения действительной матрицы на действительный вектор. Будем полагать, что

$$\tilde{a}_{ij}=a_{ij}+\delta a_{ij};\tilde{x}_j=x_j+\delta x_j;i=1,2...N;j=1,2...N, \quad (20)$$

где $\tilde{a}_{ij}$ – приближенные значения a_{ij}; δa_{ij} – ошибки округления элементов a_{ij}; $\tilde{x}_j$ – приближенные значения x_j; δx_j – ошибки округления элементов x_j.

Также предположим, что

$$\tilde{y}_i=y_i+\delta y_i,i=1,2...N, \quad (21$$

где $\tilde{y}_i$ – приближенные значения y_i; δy_i – ошибки округления y_i.

Найдем зависимость погрешности δy_i результата от ошибок округления δa_{ij}, δx_j сомножителей в (2). Для этого подставим (20) в (2). После несложных алгебраических преобразований, пренебрегая членами второго порядка малости, получим:

$$\tilde{y}_i=\sum_{j=1}^{N}\left(a_{ij}+\delta a_{ij}\right)\left(x_j+\delta x_j\right)\approx y_i+\sum_{j=1}^{N}\delta a_{ij}x_j+a_{ij}\delta x_j,\quad i=1,2...N. \quad (22)$$

Из (22) и (2) получаем следующую зависимость погрешности результата от ошибок округления исходных данных для традиционного метода (2) реализации действительного векторно-матричного умножения.

$$\delta y_i\approx\sum_{j=1}^{N}\delta a_{ij}x_j+a_{ij}\delta x_j, i=1,2...N\cdot \quad (23)$$

Перейдем к анализу метода Винограда (3). Для этого подставим (20) в (3). После несложных алгебраических преобразований, пренебрегая членами второго порядка малости, получим:

$$\begin{aligned}&\tilde{y}_i=\sum_{k=1}^{N/2}\left(a_{i,2k-1}+x_{2k}+\delta a_{i,2k-1}+\delta x_{2k}\right)\times\\&\times\left(a_{i,2k}+x_{2k-1}+\delta a_{i,2k}+\delta x_{2k-1}\right)-\\&\sum_{k=1}^{N/2}\left(a_{i,2k-1}+\delta a_{i,2k-1}\right)\cdot\left(a_{i,2k}+\delta a_{i,2k}\right)-\sum_{k=1}^{N/2}\left(x_{2k-1}+\delta x_{2k-1}\right)\times\\&\times\left(x_{2k}+\delta x_{2k}\right)\approx\\&y_i+\sum_{k=1}^{N/2}\left(x_{2k}\delta a_{i,2k}+a_{i,2k-1}\delta x_{2k-1}+\delta x_{2k}a_{i,2k}+\delta a_{i,2k-1}x_{2k-1}\right)\end{aligned} \quad (24)$$

Из (24) и (2) получаем следующую зависимость погрешности результата от ошибок округления исходных данных для метода Винограда (3) реализации действительного векторно-матричного умножения.

$$\delta y_i\approx\sum_{k=1}^{N/2}\left(x_{2k}\delta a_{i,2k}+a_{i,2k-1}\delta x_{2k-1}+\delta x_{2k}a_{i,2k}+\delta a_{i,2k-1}x_{2k-1}\right),\quad i=1,2...N. \quad (25)$$

Нетрудно убедиться, что выражения (23) и (25) являются идентичными, что указывает на одинаковую чувствительность к ошибкам округления традиционного метода (2) реализации действительного векторно-матричного умножения и аналогичного по назначению, но более простого с точки зрения вычислительной сложности метода Винограда (3).

Выводы

Проведенный анализ вычислительной сложности и устойчивости к ошибкам округления различных вариантов реализации банка фильтров позволяет сделать следующие выводы:

1. Предлагаемый подход позволяет существенно сократить вычислительную сложность реализации банка цифровых фильтров при сохранении его характеристик.
2. Предложенные алгоритмы реализации банка фильтров (3), (6) и (3), (7) позволяют сократить вычислительную сложность по сравнению с традиционным алгоритмом (2) примерно

- в 2,66 раза – при реализации на процессоре без аппаратного умножителя;
- в 1,33 раза – при реализации на процессоре с аппаратным умножителем.

При этом алгоритм (3), (7) требует выполнения несколько меньшего числа сложений по сравнению с алгоритмом (3), (6).

3. Предложенный алгоритм реализации банка фильтров (3), (6) обладает одинаковой чувствительностью к ошибкам округления по сравнению с традиционным алгоритмом (2).

4. Алгоритм (3), (7) обладает заметно большей чувствительностью к величине ошибок округления, чем алгоритм (3), (6), при несколько меньшей вычислительной сложности.

Литература

1. *Зайцев А.А.* Методы построения банков цифровых фильтров: тематический обзор // Цифровая обработка сигналов. 2003. № 1. С. 2-10.
2. *Витязев В.В.* Банки фильтров в системах широкополосной передачи данных // Цифровая обработка сигналов. 2016. № 2. С. 44-52.
3. *Витязев В.В., Харин А.В.* Оценка вычислительной эффективности методов построения банка фильтров-демодуляторов // Материалы II Всероссийской научно-технической конференции «Актуальные проблемы современной науки и производства», 15-17 ноября 2017 г. Рязань: ИП Коняхин А.В. (Book Jet), 2017. С. 113-123.
4. *Витязев В.В.* Многоскоростная обработка сигналов. М.: Горячая линия – Телеком, 2017. 336 с.
5. *Крейнделин В.Б., Смирнов А.Э., Бен Режеб Т.Б.К.* Эффективность методов обработки сигналов в системах MU-MIMO высоких порядков // T-Comm: Телекоммуникация и транспорт. 2016. Том 10. №12. С. 24-30.
6. *Крейнделин В.Б., Резнев А.А.* Свойства квазиоптимального кода в системах связи с пространственно-временным кодированием // T-Comm: Телекоммуникация и транспорт. 2013. Том 7. №10. С. 59-60.
7. *Крейнделин В.Б., Григорьева Е.Д.* Реализация банка цифровых фильтров с пониженной вычислительной сложностью // T-Comm: Телекоммуникация и транспорт. 2019. Том 13. №7. С. 48-53.
8. *Higham N.* Accuracy and Stability of Numerical Algorithms. Second edition. USA, Philadelphia, SIAM, 2002, 710 p.
9. *Тыртышников Е.Е.* Матричный анализ и линейная алгебра. М.: Физматлит, 2007. 480 с.
10. *Higham N.* Stability of a Method for Multiplying Complex Matrices with Three Real Matrix Multiplications. SIAM J. Matrix Anal. Appl. Vol.13. № 3, pp.681-687, July 1992.
11. *Блейхут Р.* Быстрые алгоритмы цифровой обработки сигналов. М.: Мир, 1989. 448 с.
12. *Бакулин М.Г., Крейнделин В.Б., Панкратов Д.Ю.* Технологии в системах радиосвязи на пути к 5G. М.: Горячая линия – Телеком, 2018. 280 с.
13. *Иванова В.Г. Тяжев А.И.* Цифровая обработка сигналов и сигнальные процессоры: учебное пособие – 2-е изд. Самара: ПГУТИ, 2017. 252 с.
14. *Kreyndelin V.B., Smirnov A.E.* Decreasing of computational complexity of demodulation algorithms in multi-antenna systems due to application of fast algorithms // Telecommunications and Radio Engineering. 2016. Vol. 75, issue: 19. P. 1757-1773.
15. *Вержбицкий В.М.* Основы численных методов. М.: Высшая школа, 2002. 840 с.
16. *Seber G.A.F.* Matrix Handbook for Statisticians. New Jersey, USA: John Wiley & Sons, 2008. 593 p.
17. *Fackler P.L.* Notes of Matrix Calculus. North Carolina State University, September 27, 2005. 14 p.
18. *Kreyndelin V.B. and Grigorieva E.D.* Analysis of Fast Algorithm of Matrix-Vector Multiplication for the Bank of Digital Filters // 2020 Systems of Signal Synchronization, Generating and Processing in Telecommunications (SYNCHROINFO), Svetlogorsk, Russia, 2020, pp. 1-5, doi: 10.1109/SYNCHROINFO49631.2020.9166081.

ANALYSIS OF FAST ALGORITHM OF MATRIX-VECTOR MULTIPLICATION FOR THE BANK OF DIGITAL FILTERS

Vitaly B. Kreyndelin, *Moscow Technical University of Communications and Informatics, Moscow, Russia,* **vitkrend@gmail.com**
Elena D. Grigorieva, *Moscow Technical University of Communications and Informatics, Moscow, Russia,* **ed.grigorieva@yandex.ru**

Abstract
Algorithms of implementation of vector-matrix multiplication are presented, which are intended for application in banks (sets) of digital filters. These algorithms provide significant savings in computational costs over traditional algorithms. At the same time, reduction of computational complexity of algorithms is achieved without any performance loss of banks (sets) of digital filters. As the basis for the construction of algorithms proposed in the article, the previously known Winograd method of multiplication of real matrices and vectors and two versions of the method of type 3M for multiplication of complex matrices and vectors are used. Methods of combining these known methods of multiplying matrices and vectors for building digital filter banks (sets) are considered. The analysis of computing complexity of such ways which showed a possibility of reduction of computing complexity in comparison with a traditional algorithm of realization of bank (set) of digital filters approximately in 2.66 times – at realization on the processor without hardware multiplier is carried out; and by 1.33 times – at realization on the processor with the hardware multiplier. These indicators are markedly higher than those of known algorithms. Analysis of sensitivity of algorithms proposed in this article to rounding errors arising by digital signal processing was carried out. Based on this analysis, an algorithm is selected that has a computational complexity smaller than that of a traditional algorithm, but its sensitivity to rounding errors is the same as that of a traditional algorithm. Recommendations are given on its practical application in the development of a bank (set) of digital filters.

Keywords: *multirate signal processing, Bank (set) of digital filters, fast algorithm, sensitivity to calculation errors, 3M method, Winograd method.*

References

1. Zaitsev A. A. (2003). Methods of Construction of Banks of Digital Filters: Thematic review. *Digital signal processing*. No. 1. P. 2-10. (in Russian)
2. Vityazev V.V. (2016). Filter banks in broadband data transmission systems. *Digital signal processing*. No. 2. P. 44-52. (in Russian)
3. Vityazev V.V., Harin A.V. (2017). Estimation of computational efficiency of methods of development of bank of filres-demodulators. *Proceedings of II Russian scientific-technical conference 'Actual problems of modern science and manufactoring'*, 15-17 November, 2017, Razyan. P. 113-123. (in Russian)
4. Vityazev V.V. (2017). Multirate signal processing. Moscow: Goryachaya-linia – Telecom. 336 p. (in Russian)
5. Kreyndelin V.B., Smirnov A.E., Ben Rejeb T.B.K. (2016). Efficiency of signal processing methods in high-order MU-MIMO systems. *T-Comm*. Vol.10, No.12. P. 24-30. (in Russian)
6. Kreyndelin V.B., Reznev A.A. (2013). Quasi-optimal code properties in space-time coding communication systems. *T-Comm*. Vol.7. No.10. P. 59-60. (in Russian)
7. Kreyndelin V.B., Grigorieva E.D. (2019). The Implementation of the Bank of Digital Filters with Reduced Computational Complexity. *T-Comm*. Vol.13. No.7. P.4 8-53. (in Russian)
8. Higham N. (2002). Accuracy and Stability of Numerical Algorithms. Second edition. USA, Philadelphia, SIAM. 710 p.
9. Tyrtyshnikov E.E. (2007). Matrix analysis and linear algebra. Moscow: Fizmatgiz. 480 p. (in Russian)
10. Higham N. (1992). Stability of a Method for Multiplying Complex Matrices with Three Real Matrix Multiplications. *SIAM J. Matrix Anal. Appl.* Vol.13. No.3. P. 681-687, July 1992.
11. Blahut R. (1989). Fast Algorithms for Digital Signal Processing. Moscow: Mir Publ. 448 p. (in Russian)
12. Bakulin M.G., .Kreyndelin V.B, Pankratov D.Y. (2018). Technologies in radiocommunication systems on the road to 5G. Moscow: Goryachaya linia – Telecom. 280 p. (in Russian)
13. Ivanova V.G., Tyazhev A.I. (2017). Digital Signal Processing and Signal Processors: Tutorial. 2-nd ed. Samara: PGUTI. 252 p. (in Russian)
14. Kreyndelin, V.B., Smirnov A.E. (2016). Decreasing of computational complexity of demodulation algorithms in multi-antenna systems due to application of fast algorithms. *Telecommunications and Radio Engineering*. Vol. 75, issue: 19. P. 1757-1773.
15. Verzhbitsky V.M. (2002). Basis of numeric methods. Moscow: Vyshaya shkola. 840 p. (in Russian)
16. Seber G.A.F. (2008). Matrix Handbook for Statisticians. New Jersey, USA: John Wiley & Sons. 593 p.
17. Fackler P.L. (2005). Notes of Matrix Calculus. North Carolina State University, September 27, 2005. 14 p.
18. Kreyndelin V.B. and Grigorieva E.D. (2020). Analysis of Fast Algorithm of Matrix-Vector Multiplication for the Bank of Digital Filters. *2020 Systems of Signal Synchronization, Generating and Processing in Telecommunications (SYNCHROINFO)*, Svetlogorsk, Russia, 2020, pp. 1-5, doi: 10.1109/SYNCHROINFO49631.2020.9166081.

Information about authors
Vitaly B. Kreyndelin, *Moscow Technical University of Communications and Informatics, chief of the Department of electric circuit theory, Moscow, Russia*
Elena D. Grigorieva, *Moscow Technical University of Communications and Informatics, member of the Department of electric circuit theory, Moscow, Russia*